ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO

TRABALHO DE FORMATURA

"SIMULAÇÃO COMO UMA FERRAMENTA DE PLANEJAMENTO EM AMBIENTE DE PROCESSAMENTO DE DADOS"

Dario Ikuo Miyake

Prof.Orientadora: Heloísa Helena O.Martins Shih

1984

Departamento de Engenharia de Produção
Escola Politécnica da USP
AVALIAÇÃO DO TRABALHO DE FORMATURA

Prof HELOISA	A
Prof. SHIMIZU	C
Prof. P. BUENO	A
Nota Final	A

ERRATAS

Onde se lê	Leia-se	Página
1. ..seleção da alternativa a ser simulada	..seleção da alternativa a ser implantada	14
2. ..(figura 4.13 e Anexo I)..	..(figura 4.12 e Anexo I)..	106

Obs. A figura 5.16 que consta na página 205 só é mencionada na página 206.

Aos

meus pais.

AGRADECIMENTOS

Reservo este espaço para expressar os meus sinceros agradecimentos àqueles que me deram apoio e colaboração na realização deste trabalho:

- À Professora Heloísa Helena O.Martins Shih pela orientação segura, dedicada e sempre atenciosa que me foi prestada;

- Ao Engº Renato Arata, pelo apoio e orientação que me foram dados;

- Ao Engº Raul Armando Gennari Filho, pela oportunidade de estágio e recursos oferecidos;

- Aos Engºs. Mário Miyake e Shih Li Chung do Agrupamento de Pesquisa Operacional e Estatística do IPT pelos livros emprestados e experiências transmitidas;

- Ao Engº José Agostinho Baitello pela orientação quanto ao uso do pacote GPSS;

- À Solange Andreoni pela orientação quanto ao uso do pacote SPSS;

- À Eliana Toyoshima pela presteza e qualidade dos serviços de datilografia executados;

- À todos da minha família que contribuíram para a concretização deste trabalho.

SUMÁRIO

A questão do aumento de produtividade, hoje já não se limita somente ao âmbito das fábricas e está sendo perseguida nos mais diversos setores da economia.

Este trabalho tem como objetivo, verificar a viabilidade de se desenvolver uma ferramenta de "software", que facilite aos programadores no trabalho de desenvolvimento de programas novos. É mais um exemplo da possibilidade de se substituir grande parte do trabalho manual ou mental realizado pelo ser humano, mediante uso do computador.

Esse estudo de avaliação foi baseado sobretudo na técnica de Simulação, e portanto, exigiu esforços no sentido de se desenvolver um modelo adequado, além de envolver conceitos teóricos advindos de outras áreas do conhecimento relacionadas com a Engenharia de Produção.

INDICE

1. INTRODUÇÃO 1
   1.1. A Empresa 2
   1.2. O Tema do Trabalho 7

2. ALTERNATIVAS LEVANTADAS E PRÉ-AVALIAÇÃO 11
   2.1. Alternativas e Linhas de Ação 13
      2.1.1. Desenvolvimento de uma ferramenta de "software" própria 16
      2.1.2. Introdução de ferramentas desenvolvidas por terceiros 21
   2.2. Avaliação das Alternativas 25
      2.2.1. Critérios de avaliação 26
      2.2.2. Qualificação das alternativas 28
   2.3. Seleção da Melhor Alternativa 28

3. O META-COBOL 36
   3.1. Descrição 37
   3.2. Exemplo de Aplicação do Meta-Cobol 46

4. A SITUAÇÃO ATUAL 56
   4.1. Apresentação da Situação Atual 67
      4.1.1. O sistema de trabalho do pessoal de desenvolvimento de Sistemas de Informação 67
      4.1.2. O uso de ferramentas de "software" no ambiente 75
      4.1.3. Trabalhos de manutenção e outras atividades 78
      4.1.4. Dados do atual nível de atividade 78
      4.1.5. Dados dos atuais tempos de atendimento 90
   4.2. Anãlise da Situação Atual 98
      4.2.1. Com relação ao nível de atividade da Programação 98

4.2.2. Com relação à distribuição de atividade da Programação 103
4.2.3. Com relação aos tempos efetivos de atendimento 104

5. AVALIAÇÃO DO IMPACTO NO AMBIENTE 108
5.1. Definição do Sistema 112
5.2. Formulação do Modelo 116
5.2.1. Linguagem escolhida para simulação 116
5.2.2. Geração de transações 118
5.2.3. Estabelecimento de condições iniciais 120
5.2.4. Modelos de atendimento 121
5.2.5. Filas e critérios de prioridade 125
5.3. Tratamento dos Dados 134
5.3.1. Dados para o estabelecimento das condições iniciais 135
5.3.2. Dados para a geração de transações 136
5.3.3. Dados de tempos de atendimento 138
5.4. Desenvolvimento de um Programa para o Modelo 147
5.5. Validação do Modelo 150
5.6. Planejamento das Simulações do Uso do Meta-Cobol no Ambiente da Programação 154
5.6.1. Definição do horizonte de simulação 156
5.6.2. O cenário de simulação 158
5.6.3. Preparação dos dados para a simulação do futuro 162
5.6.4. Ajustes no programa de simulação 181
5.7. Processamento das Simulações e Interpretação dos Resultados 183
5.7.1. Processamento das simulações para a avaliação do impacto da introdução do Meta-Cobol 183
5.7.2. Resultados da avaliação do impacto da introdução do Meta-Cobol 189
5.7.3. Processamento das simulações para Análise de sensibilidade 210
5.7.4. Resultados da análise de sensibilidade 211

6. VIABILIDADE ECONÔMICA DO META-COBOL 221
6.1. Levantamento de Custos 222
6.2. Levantamento de Benefícios 228
6.3. Confronto dos Benefícios com os Custos 231

7. CONCLUSÕES 235
7.1. Quanto ao Estudo de Avaliação do Meta-Cobol 236
7.2. Quanto ao Modelo Desenvolvido 240

APÊNDICE A 243

ANEXO I 245
ANEXO II 259
ANEXO III 275
ANEXO IV 284
ANEXO V 292
ANEXO VI 310
ANEXO VII 326
ANEXO VIII 342
ANEXO IX 349

BIBLIOGRAFIA 356

GLOSSÁRIO

ACF - Advanced Communication Function

"batch" - Tipo de processamento em que o tempo de resposta não é a principal exigência e que envolve a manipulação de um grande volume de dados de entrada (lotes) e de relatórios gerados

COBOL - Common Business Oriented Language

Compilador - Programa que traduz um programa em linguagem de alto nível para um programa em código de máquina

comandos macros - correspondem a rotinas de operações que podem ser subdivididas em seqüências relativamente longas de instruções de máquina

CPD - Centro de Processamento de Dados

e.p.a. - elaborado pelo autor

FORTRAN - Formula Translation

Gerador - Programa capaz de construir outros programas (total ou parcialmente) utilizando parâmetros que variam de caso para caso

GPSS - General Purpose Simulation System

"hardware" - Conjunto formado pelas unidades físicas (equipamentos) que constituem um sistema de computação

IBM - International Business Machines

"link-edição" - Processo adotado em alguns computadores (ex. IBM) para complementar a etapa de compilação e tornar o programa fonte executável em máquina

"on-line" - Em comunicação direta com o sistema

OS/VS - Operating System/Virtual Storage

"patch" - Emenda realizada em programas desenvolvidos por meio de geradores, com o fim de complementar a sua função

PL/I - Program Language Number I

Programa fonte - Programa codificado em linguagem de alto nível

Programa objeto - Programa em código de máquina

ref. - Referência

"software" - Conjunto de programas (componente lógico) que determinando o funcionamento do "hardware", permite que o computador realize seu trabalho.

SPSS - Statistical Package for the Social Sciences

Tradução - Processo de conversão das instruções - fonte de uma linguagem em instruções - fonte equivalentes de uma segunda linguagem

VSAM - Virtual Storage Acess Method

VTAM - Virtual Telecommunication Acess Method

CONVENÇÕES SIMBÓLICAS

1. (n) número da nota de rodapé chamada.

2. (ref.n) número da referência bibliográfica chamada.

ÍNDICE DE TABELAS

Tabelas

1.1 - Configuração básica de "software" do ambiente 4

2.1 - Alternativas de linguagem da quarta geração e banco de dados para a alternativa 3 24

2.2 - Alguns bancos de dados já lançados no Brasil 25

2.3 - Resultados da avaliação qualitativa das alternativas 29

2.4 - Avaliação da alternativa 1 30

2.5 - Avaliação da alternativa 2 31

2.6 - Avaliação da alternativa 3 32

2.7 - Avaliação da alternativa 4 33

2.8 - Avaliação da alternativa 5 34

4.1 - Codificação das atividades da Programação 80

4.2 - Número de programas novos recebidos por mês para serem desenvolvidos 83

4.3 - Número de programas para manutenção recebidos por mês 83

4.4 - Número de programas novos liberados por mês 84

4.5 - Número de programas para manutenção liberados por mês 84

4.6 - Número de programas novos para desenvolvimento em mãos ao fim de cada mês 85

Tabelas

4.7 - Número de programas para manutenção em mãos ao fim de cada mês 85

4.8 - Aproveitamento do tempo útil da Programação para atividades com programas e outras atividades 86

4.9 - Dados globais médios referentes ao fluxo de programas pela Programação 90

5.1 - Código para interpretação do parâmetro PH 1 119

5.2 - Critérios de prioridade 134

5.3 - Tempo médio entre chegadas por tipo de atividade para o ano de 1984 137

5.4 - Estimativas dos coeficientes de assimetria e de curtose obtidas de cada amostra 141

5.5 - Função descontínua para a determinação do tempo de duração das férias 144

5.6 - Estimativas para os outros tempos envolvidos no atendimento de serviços produtivos 145

5.7 - Resumo dos resultados obtidos nas simulações para calibração do modelo 152

5.8 - Projeção do número de programas novos a ser produzidos por ano e por tipo (para a Programação) 165

5.9 - Projeção da taxa média mensal de chegadas de programas por tipo para cada ano (para o grupo médio) 165

Tabelas

5.10 - Tempo médio entre chegadas em horas úteis para cada tipo de programa e por ano (para o grupo médio) 166

5.11 - Tabela utilizada para se estimar o número de programas em produção no início de cada ano 171

5.12 - Projeção do número de manutenção a serem executadas 172

5.13 - Projeção dos serviços de manutenção a serem executados por tipo e por ano 173

5.14 - Projeção da taxa média mensal de chegadas de manutenções por tipo para cada ano 174

5.15 - Tempo médio entre chegadas de manutenções em horas úteis por tipo de programa para cada ano 174

5.16 - Dados para as gerações de outras atividades e de férias conforme o tamanho da equipe 175

5.17 - Média de programas novos e manutenções em mãos para cada ano 176

5.18 - Distribuição relativa (em %) dos programas novos e para manutenção em mãos na situação inicial 177

5.19 - Estimativas para a redução nos tempos efetivos de desenvolvimento para programas "batch" de atualização 179

5.20 - Média de programas em mãos estimada 185

5.21 - Combinações de número de programador, situação e ano experimentados 188

Tabelas

5.22 - Dados globais obtidos das simulações do futuro com o Meta-Cobol 193

5.23 - Dados globais obtidos das simulações do futuro sem o Meta-Cobol 194

5.24 - Média dos valores obtidos nas simulações em comparação com os valores médios reais esperados 196

5.25 - Valores utilizados para a determinação do efetivo adequado à Programação 198

5.26 - Produção média individual (programas liberados/mês) 201

5.27 - Prazos médios em dias úteis com a introdução do Meta-Cobol 203

5.28 - Prazos médios em dias úteis sem a introdução do Meta-Cobol 204

5.29 - Distribuição do tempo útil corrido, excluindo férias e ociosidade, entre desenvolvimento e manutenção de programas 208

5.30 - Dados globais das simulações processadas para o ano de 1988 considerando o aproveitamento do Meta-Cobol por um grupo médio de 13 programadores 212

5.31 - Dados globais obtidos das simulações para a análise da sensibilidade do desempenho do sistema em relação à variação do seu tamanho 217

6.1 - Valores dos custos unitários de recursos 222

Tabelas

6.2 - Custos dos recursos necessários ao desenvolvimento e implantação do Meta-Cobol 225

6.3 - Custo de utilização do computador para geração de programas fonte com o uso do pré-compilador do Meta-Cobol 227

6.4 - Custos anuais de manutenção para o período entre 1986 a 1989 228

6.5 - Economia anuais com redução do efetivo de programadores 230

6.6 - Economia anuais com a redução das requisições de serviços de digitação 231

7.1 - Distribuição do custo e do prazo de desenvolvimento de Sistemas de Informação 238

ÍNDICE DE FIGURAS

Figuras

1.1 - Configuração básica de "hardware" 5

2.1 - Esquema do processo em curso com vistas à seleção da alternativa a ser implantada 14

2.2 - Esquema de utilização de um gerador de programas fonte para aplicações "batch" 17

2.3 - Estimativa da participação das ferramentas convencionais no total de "software" a ser desenvolvido nos próximos anos 19

2.4 - Integração das ferramentas de "software" propostas pela alternativa 3 22

3.1 - Processo de pré-compilação do programa original para geração de programa fonte em COBOL 44

3.2 - Relatório da Situação das Necessidades de Matéria-Prima 47

3.3 - Ilustração da função do programa exemplo 48

3.4 - Programa exemplo em Meta-Cobol 49

3.5 - Programa fonte em COBOL para a geração do Relatório das Necessidades de Matéria-Prima 50

4.1 - Estrutura organizacional do pessoal de desenvolvimento de sistemas 68

4.2 - Etapas do desenvolvimento de programas novos 72

4.3 - Relacionamento do grupo de programadores com a respectiva Equipe de Analistas 74

Figuras

4.4 - Distribuição dos programas "on line" conforme a aplicabilidade do SCOA para sua geração 76

4.5 - Gráfico do ganho de produtividade proporcionável por uma ferramenta de "software" conforme a sua aplicabilidade 77

4.6 - Exemplo da Folha de Acompanhamento da Programação 81

4.7 - Representação gráfica do fluxo de programas novos que passa pela Programação 87

4.8 - Representação gráfica do fluxo de programas para manutenção que passa pela Programação 88

4.9 - Representação gráfica da distribuição do tempo útil entre atividades produtivas e não produtivas 89

4.10 - Exemplo de Folha de Acompanhamento do Programador 93

4.11 - Distribuição de freqüências do tempo de desenvolvimento de programas "batch" de atualização 99

4.12 - Histograma e estatísticas da distribuição do tempo efetivo para o desenvolvimento de programas "batch" de atualização 100

4.13 - Formato aproximado da curva da distribuição da densidade de probabilidade dos tempos efetivos de atendimento 106

5.1 - Recursos necessários para a elaboração do modelo em função de sua fidelidade 114

Figuras

5.2 - Esquema do sistema a ser considerado na formulação do modelo 115

5.3 - Exemplo de concatenação de atividades por um programador 122

5.4 - Modelo de atendimento das atividades de desenvolvimento de programa novo 123

5.5 - Modelo global de atendimento de atividades 126

5.6 - Filas a serem tratadas pelo grupo médio de programação 129

5.7 - Carta de E.S.Pearson, preenchida com os pontos obtidos das amostras 142

5.8 - Função de distribuição acumulada para o tempo de atendimento dos serviços não concluíveis 146

5.9 - Integração dos módulos no programa de simulação 149

5.10 - Gráfico do fator q em função de p 153

5.11 - Gráfico do número médio de programas em mãos em função de p 153

5.12 - Gráfico das variáveis crescentes do sistema 163

5.13 - Evolução do total de programas em produção ao longo dos anos 168

5.14 - Ilustração da técnica utilizada para a estimação do número de programadores adequado 187

5.15 - Projeção do número de programadores necessário para o período de 1985 a 1989 a se continuar sem o Meta-Cobol 199

Figuras

5.16 - Distribuição do número de etapas de depuração por programa com e sem o uso do Meta-Cobol 205

5.17 - Gráfico da média de programas em mãos em função do ganho de produtividade proporcionável pelo Meta-Cobol para um grupo médio de 13 programadores em 1988 213

5.18 - Gráfico do prazo médio de entrega de programas "batch" em função do ganho de produtividade proporcionável pelo Meta-Cobol para um grupo médio de 13 programadores em 1988 214

5.19 - Gráfico do prazo médio para a entrega de programas "batch" em função do tamanho do grupo 218

5.20 - Gráfico comparativo dos prazos realizados contra os prazos estipulados 219

6.1 - Fluxo dos dispêndios anuais em ORTN 232

6.2 - Fluxo das economias anuais em ORTN 232

1. INTRODUÇÃO

1.1. A EMPRESA

Este trabalho foi realizado durante o estágio em um dos maiores grupos industriais do Brasil. Este grupo é formado por várias empresas, atuando em diversas áreas da economia. A primeira empresa do grupo foi fundada em 1918 para montagem e manutenção de elevadores importados. Em 1944, em consequência das dificuldades de importação impostas pela II Grande Guerra, constituiu-se uma outra empresa para a comercialização de barras e chapas de aço e peças fundidas importadas e nacionais. Em 1954, criou-se uma empresa para a fabricação de equipamentos industriais de médio e grande porte.

A partir daí, o grupo experimentou um rápido crescimento com a criação de várias outras empresas, principalmente durante o processo de rápida industrialização do Brasil na década de 70, e hoje conta com subsidiárias e sucursais em diversos países da América do Sul. Graças a uma política de manter alto padrão de qualidade em seus produtos e serviços, à ênfase em utilizar a mais adequada e atual tecnologia, e à política de permanente reinversão de recursos, esse grupo conquistou participação em consideráveis segmentos do mercado nacional e até internacional, estando hoje consolidado como dos maiores grupos industriais de capital privado.

As diversas empresas do grupo conforme as suas características estão agrupadas em 3 setores quais sejam:

- Setor Industrial.
- Setor Siderúrgico.
- Setor de Bens de Capital.

O estágio em si, foi desenvolvido num outro setor, o de Informática e Serviços de Teleprocessamento. Esse setor originou-se do antigo Departamento de Análise e Programação, tendo sido recentemente criado pela crescente importância e expansão dos seus serviços. Todos estes setores estão subordinados à Administração Geral do grupo.

Por orientação estratégica, as empresas do grupo têm procurado privilegiar e aproveitar os mais recentes avanços na área da tecnologia eletrônica digital e de processamento de dados, concentrando as atenções nas seguintes áreas de atuação:

- Aplicação da eletrônica digital (tecnologia do microprocessador) em seus produtos e utilização de equipamentos de automação industrial.

- Aproveitamento da tecnologia CAD (Computer Assisted Design) para a automação de trabalhos de projeto e desenho de equipamentos.

- Desenvolvimento de Sistemas de Informação para automação administrativa.

O âmbito do presente trabalho, desenvolvido no Setor de Informática e Serviços de Teleprocessamento, está localizado neste último tópico.

Com o advento da tecnologia de processamento de dados "on-line"[1] (conscientização de cada vez maior necessidade de se ter informações mais atualizadas e confiáveis), foram desenvolvidos e implantados dezenas de novos e modernos sistemas, além de terem sido reformulados e atualizados sistemas mais antigos que já estavam em operação e utilizavam somente processamento "batch". Com isto, o uso de terminais de vídeo difundiu-se tanto nos setores produtivos como administrativos, de todas as empresas do grupo.

Como todas essas empresas compartilham do Centro de

---

[1] Os leitores que não estiverem familiarizados com esta e outras expressões comuns no jargão da área de processamento de dados, e que serão daqui em diante, constantemente referidas no texto, poderão consultar o seu significado no Glossário do Trabalho.

Processamento de Dados (CPD) central, este cresceu vertiginosamente e hoje já pode ser considerado como um dos maiores do Brasil, no setor industrial. A figura 1.1 e a tabela 1.1 apresentam respectivamente, a configuração básica de "hardware" e de "software" do CPD em meados de 1984. Este crescimento, evidentemente, foi acompanhado também por um aumento das instalações, da estrutura organizacional e do quadro de funcionários (o efetivo de programadores e analistas, quase quadruplicou entre 1980 e 1984).

Entretanto, essa evolução não se deu somente em termos físicos. Houve também uma evolução do ponto de vista qualitativo, no trabalho de desenvolvimento de Sistemas de Informação. O pessoal dessa área, seguindo orientação de ordem tática (gerencial), tem procurado aumentar a produtividade e a qualidade do seu trabalho, mediante adoção e uso de "ferramentas adequadas para cada situação". Assim, ao longo dos últimos anos, foram introduzidas diversas novas ferramentas tais como terminais de vídeo, pacotes para aplicações científicas, programas utilitários da IBM, recursos de apoio (transações) "on-line" para uso através de terminais e também, ferramentas para auxílio ao desenvolvimento de sistemas criadas interna

Componente	Denominação
. Sistema Operacional	OS/VS1
. Gerenciador da organização de dados nas memórias de disco	VSAM
. Monitor de Teleprocessamento	CICS
. Gerenciador de recursos de comunicação de sistema	ACF - VTAM
. Gerenciador de linhas de comunicação	ACF - NCP

Tabela 1.1 - Configuração básica de "software" do ambiente - e.p.a. -

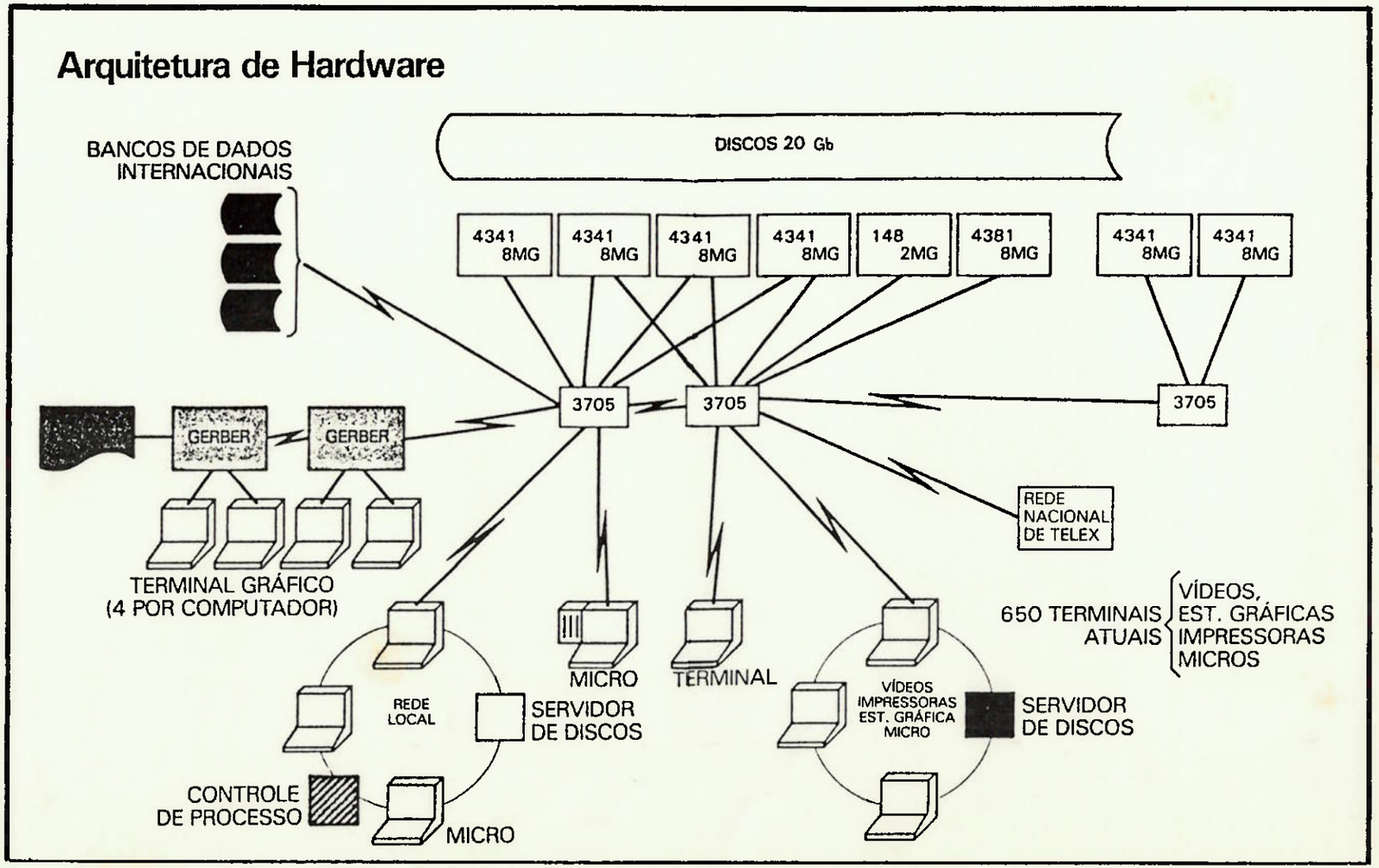


Figura 1.1 - Configuração básica de "hardware"

mente. Dentre estas ferramentas[1] desenvolvidas destacam-se as seguintes:

a) Gerador de programas fonte em COBOL destinados a aplicações "on-line" (SCOA).

b) Estrutura de armazenamento de dados (DB VSAM).

c) Dicionário de Dados.

d) Sistema para gerenciamento a controle das informações sobre os dados armazenados e outros recursos utilizados no ambiente de desenvolvimento de Sistemas de Informação (Sistema Global).

e) Gerador de programas "batch" para emissão de relatórios.

No Setor de Informática e Serviços de Teleprocessamento, existe ainda, uma grande expectativa de que o seu crescimento deverá continuar firme nos próximos anos. Mas se por um lado o crescimento é um aspecto positivo e salutar, é necessário também, estar atento às suas possíveis implicações negativas. Conscientes disso, membros da direção, gerência e supervisão desse setor, têm se reunido periodicamente para discutir questões de ordem estratégica e tática. E dentre essas questões, sem dúvida alguma, uma das mais importantes e que tem despertado especial atenção, é a que se refere ao problema da produtividade do trabalho de desenvolvimento de sistemas. O desenvolvimento de um Sistema de Informação pode ser dividido basicamente em 4 etapas:

- Levantamento de requisitos e análise do sistema.
- Projeto do sistema.
- Codificação dos programas.
- Implantação e testes do sistema.

---

[1] Como no decorrer dos próximos Capítulos, voltaremos a nos referir a estas ferramentas próprias do ambiente, preparamos no Apêndice A uma descrição mais apurada de cada uma delas.

Depois de desenvolvido, o sistema entra em fase operacional por um tempo que varia de caso para caso (vida do sistema). Durante a operação, normalmente são necessárias intervenções para manutenção.

Assim, cientes de que a questão da produtividade deve ser examinada tendo-se em mente todas essas etapas, têm sido levantadas várias sugestões no sentido de se propor uma nova metodologia de desenvolvimento de sistemas, que resulte efetivamente num melhor desempenho da equipe de desenvolvimento. Evidentemente, essa nova metodologia deverá ser constituída da adoção de um conjunto de novos métodos e da manutenção de outros, que se ajuste da melhor maneira possível ao problema e às características do ambiente.

Dentre as novas medidas que têm sido sugeridas, uma das mais importantes é a de se melhorar a produtividade dos programadores no desenvolvimento de programas "batch". Recentemente, problema análogo quanto ao desenvolvimento de programas "on-line", foi solucionado com a criação de uma ferramenta de "software" própria, que é o SCOA. Cogita-se agora, tomar uma medida semelhante para o caso dos programas "batch". É a respeito da escolha e da viabilidade de desenvolvimento desta nova ferramenta, que trataremos no presente trabalho.

1.2. O TEMA DO TRABALHO

Na época do início do estágio, estava em fase de concepção o estudo que tratava da viabilidade de se criar uma nova ferramenta de "software", como alternativa para se melhorar o produtividade dos programadores no desenvolvimento de programas "batch". Antes de escolherem a alternativa a ser estudada, o pessoal mais diretamente envolvido no problema (gerente da Área de Desenvolvimento, seus supervisores e alguns analistas), levantou e avaliou diversas alternativas de ferramentas de "software", dentre as quais algumas já desenvolvidas por terceiros e existentes no mercado nacional. Todo esse processo de levantamento, avaliação e seleção de alternativas é tratado com maior detalhe no Capítulo 2, embora

o autor não tenha tido participação direta.

Partindo-se dessa situação, ou seja, com a ferramenta de "software" a ser estudada já definida, foi estabelecida com a gerência da Área de Desenvolvimento a seguinte divisão dos estudos a serem realizados:

a) Desenvolvimento, a cargo do autor deste trabalho, de um estudo de avaliação dos impactos que seriam causados por essa ferramenta de "software" no ambiente da Programação, e com base nos resultados a serem obtidos, verificar a sua viabilidade econômica.

b) Desenvolvimento por parte de uma equipe interna formada pelo próprio gerente, mais alguns de seus supervisores e outros analistas, de um ante-projeto da ferramenta para se avaliar a viabilidade técnica desta alternativa, com base no "know-how" acumulado com a criação de outras ferramentas de "software" próprias. (vide item 1.1.).

O tema deste trabalho é pois, o estudo mencionado em a . Antes de prosseguirmos, traçaremos brevemente os limites do estudo e descreveremos o escopo do resto do trabalho.

Vimos que abordaremos em particular, a questão da viabilidade de se adotar uma ferramenta de "software" para facilitar o desenvolvimento de programas "batch". Embora nada impeça que tal ferramenta caso adotada, venha a ser aproveitada também pelos analistas, restringiremos o estudo ao seu uso básico principal: no ambiente da Programação. Além disso , embora o objetivo principal de se criar tal ferramenta, seja o de se auxiliar os trabalhos de desenvolvimento, é possível que, na prática, pudesse ser aproveitada também em serviços de manutenção. No entanto, não consideraremos essa hipótese, pois o interesse maior é o de se aproveitá-la no seu objetivo principal e as perspectivas quanto ao seu uso com outras finalidades ainda são muito incertas.

Por fim, ressaltamos que a rigor, essa ferramenta

não pretende ser universal. Os programas "batch", conforme a sua aplicação, se dividem em programas administrativo-comerciais e científicos. No ambiente do Setor de Informática e Serviços de Teleprocessamento,os primeiros são codificados na linguagem COBOL e os científicos em FORTRAN ou PL/I. Existem ainda alguns raros programas codificados em ASSEMBLER. A ferramenta proposta em si, viria a abranger a parcela composta pelos programas administrativo-comerciais, que constituem a grande maioria dos programas "batch". Os demais tipos de programas, que convencionaremos chamar de "programas "batch" em outras linguagens", deverão continuar a ser desenvolvidos pelo método atual.

O estudo de avaliação dos impactos a que nos propomos, mostrou-se ser desde o início, uma tarefa das mais difíceis, principalmente pela dificuldade em se encontrar pesquisas semelhantes já realizadas, pelas características muito particulares do ambiente da Programação, pela enorme variedade de fatores e variáveis envolvidas e pela escassez de dados importantes ao trabalho.

Optamos por basear o desenvolvimento do trabalho sobretudo em técnicas estatísticas e de simulação.

No Capítulo 2, como mencionado anteriormente, apresentaremos as alternativas levantadas, a sua avaliação e a seleção da melhor das mesmas. No Capítulo 3, apresentaremos com maior detalhe a ferramenta de "software" sugerida pela melhor alternativa. No Capítulo 4, apresentaremos a descrição da situação atual (válida para todo o ano de 1984) com dados representativos da realidade. No Capítulo 5, que é o corpo do trabalho, apresentaremos o processo de avaliação dos impactos no ambiente partindo desde a definição do modelo construído para o estudo, até o processamento de simulação para se investigar o desempenho da Programação com a introdução de tal ferramenta, além de interpretarmos os resultados obtidos. Incluímos também neste Capítulo, um estudo de Análise de Sensibilidade.

No Capítulo 6, aproveitando os resultados obtidos nas simulações, apresentaremos um estudo de viabilidade econômica dessa ferramenta. Finalmente, o Capítulo 7, foi reservado para a apresentação das conclusões gerais do trabalho.

2. ALTERNATIVAS LEVANTADAS E PRÉ-AVALIAÇÃO

Como vimos no Capítulo 1, quando se iniciou o presente trabalho, o pessoal engajado nos estudos visando o aumento de produtividade na Programação, já havia levantado e avaliado (sem a participação do autor) as alternativas mais viáveis para se "atacar" o problema do desenvolvimento de programas "batch". Todas estas alternativas envolvem o uso de ferramentas de "software". Procuraremos ilustrar sucintamente neste capítulo, como se desenrolou tal processo até a decisão de se selecionar uma alternativa que será objeto do presente estudo. Isso contribuirá para que o leitor tenha uma melhor idéia da ferramenta que foi selecionada e dos motivos que levaram a isso.

As alternativas que foram examinadas, têm em comum o fato de sugerirem o uso de alguma(s) ferramenta(s)de "software". Isto aconteceu pelo fato de as pessoas atualmente envolvidas nos estudos terem formação e experiência essencialmente técnicas, estando razoavelmente a par das ferramentas propostas e em condições de avaliá-las. Evidentemente, existem algumas outras alternativas, dentre as quais, aquelas que encaram o problema segundo o ponto de vista de Recursos Humanos(como maior ênfase em treinamento, política de reciclagem e atualização, etc.) e que, sem dúvida, também merecem consideração. Porém, vamos nos ater às alternativas consideradas, desenvolvendo o estudo de pré-avaliação para a alternativa escolhida. Os resultados alcançados poderão ser, então, comparados ou acrescentados a outros que adviriam de medidas baseadas em outros ramos do conhecimento. Somente, então, decidir-se-á pelas medidas que deverão ser, efetivamente, escolhidas e implantadas. Há inclusive, a possibilidade de que no final, a decisão acabe sendo a de se combinar as melhores alternativas (alguns pontos fracos da melhor alternativa podem ser compensados pelos pontos fortes de outra(s)) e não a de se implantar somente a melhor, de maneira isolada.

Por tudo isso, é que o título deste capítulo usa o termo "pré-avaliação", afinal, as alternativas de "software" foram apenas pré-avaliadas e somente a melhor delas é que está sendo, presentemente, detalhada para, posteriormente, passar pelo processo de avaliação e seleção final.

O esquema da figura 2.1 lustra todo esse processo de maneira global, localizando também as fronteiras do presente trabalho.

2.1. ALTERNATIVAS E LINHAS DE AÇÃO

As alternativas foram sendo veiculadas e levantadas aos poucos, baseando-se na experiência das pessoas envolvidas no processo, em informações colhidas em outras fontes (publicações especializadas, empresas do ramo, estudo de casos reais, etc.) e nas constatações que foram feitas há cerca de 4 anos, quando houve uma grande reconfiguração dos recursos de "hardware" e de "software" do ambiente de processamento de dados (troca dos antigos equipamentos Burroughs por modelos mais recentes da IBM), ocasião na qual, várias ferramentas de "software", inclusive algumas que serão aqui abordadas, foram testadas e avaliadas "in loco".

À medida que as alternativas foram sendo levantadas, ficou bem evidente a existência de dois conjuntos distintos que englobavam alternativas com características semelhantes entre si, ou seja, constatou-se a existência de duas linhas de ação diferentes para se abordar o problema em questão, quais sejam:

Linha 1 - Desenvolvimento de uma ferramenta de "software" própria.

Linha 2 - Introdução de novas ferramentas de "software", já desenvolvidas por terceiros e lançadas no mercado.

A diferença fundamental entre as duas linhas de ação[1] reside na origem da ferramenta: ou o Setor de Informática e

---

[1] Doravante referenciadas somente pela sua numeração.

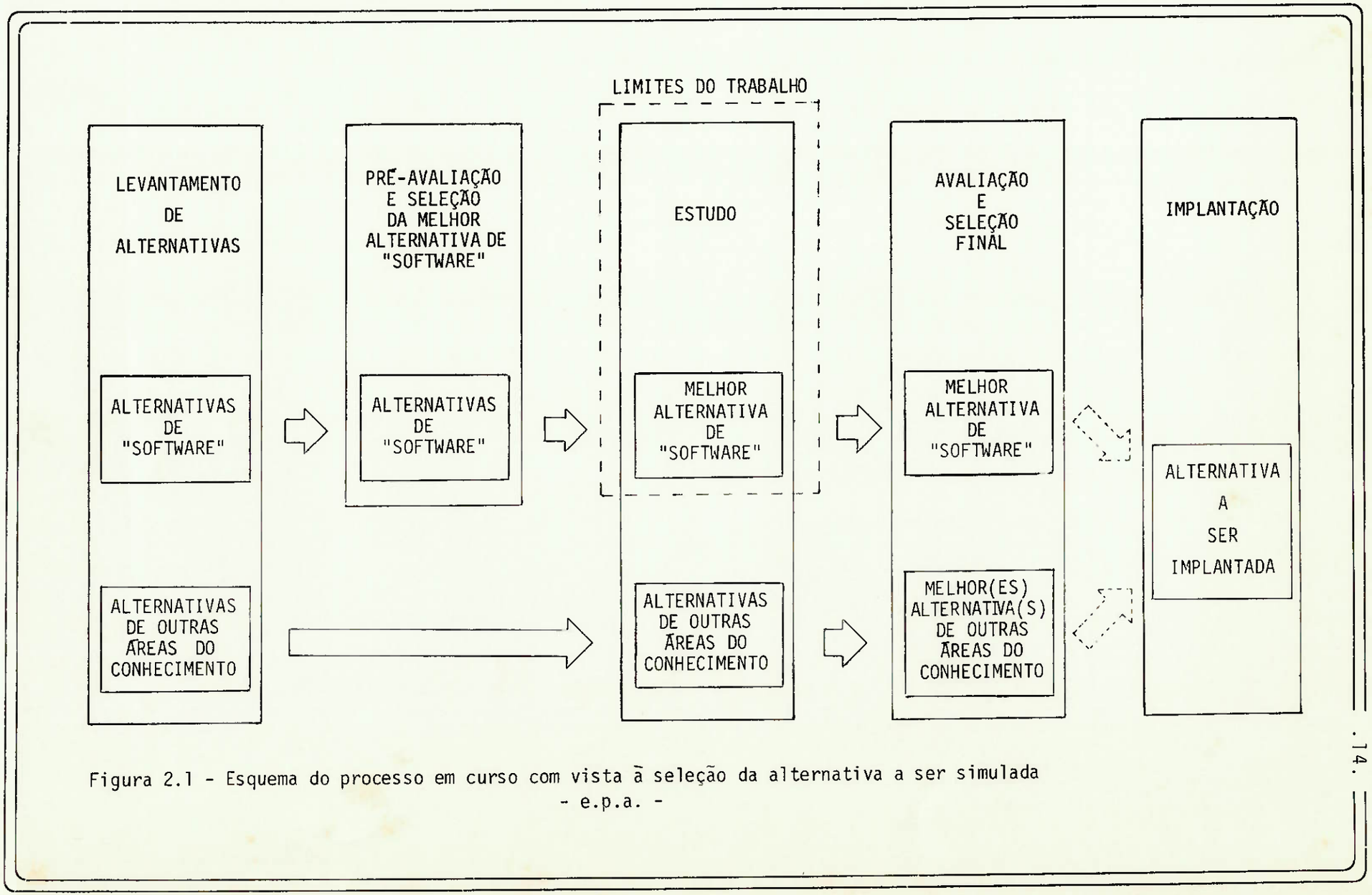


Figura 2.1 - Esquema do processo em curso com vista à seleção da alternativa a ser simulada - e.p.a. -

Serviços de Teleprocessamento adota uma ferramenta própria arcando com os esforços necessários para o seu desenvolvimento, ou opta por comprar uma que já foi desenvolvida por outra empresa. Elas têm a característica em comum de se visar maior produtividade no desenvolvimento de programas "batch", aliviando o trabalho dos programadores via introdução de alguma(s) nova(s) ferramenta(s) de "software" que viria(m) a assumir grande parte dos esforços humanos hoje necessários. É mais um exemplo de possibilidade de mecanização[1] do trablaho utilizando-se o próprio computador.

Como as alternativas são eminentemente técnicas, serão apresentados alguns conceitos importantes para um melhor entendimento das mesmas:

BANCO DE DADOS - Coleção de dados inter-relacionados, armazenados juntos e com redundância controlada para servir a uma ou mais aplicações; os dados são armazenados (em arquivos) de forma a serem independentes dos programas que os usam. Um método comum é utilizado na adição de novos dados, na modificação e na recuperação de dados existentes no Banco de Dados (vide ref. 1).

DICIONÁRIO DE DADOS - "Depósito" de dados que descreve a natureza de cada dado usado no sistema, muitas vezes incluindo descrições de processos, entradas de glossário, relacionamentos entre os dados armazenados e outros itens (vide ref. 2).

DIRETÓRIO DE DADOS - "Depósito" de dados, geralmente processado por computador, que indica onde cada dado do sistema está armazenado (vide ref. 2).

SISTEMA GERENCIADOR DE BANCO DE DADOS (SGBD) - ferramenta que viabiliza o uso de banco de dados permitindo a independência

---

[1] No caso, apenas parcial, pois nenhuma das ferramentas consideradas é capaz de substituir por completo a participação do programador.

entre os seus dados e os programas. Um bom SGBD também oferece estruturas de dados compatíveis com as características dos diversos usuários, permite e mantém relacionamento entre os dados, controla redundância, garante a integridade e segurança dos dados, permite o acesso concorrente, oferece interfaces para manipulação de dados voltados ao programador e usuário final ("end-user"), e, oferece um conjunto de funções de apoio ao administrador do Banco de Dados (vide ref. 1).

Apresentaremos agora, nos subitens seguintes, as duas linhas de ação, cada qual com a descrição do seu conjunto de alternativas. A avaliação das mesmas é apresentado no item 2.2. e a seleção da melhor alternativa no item 2.3..

Para que o leitor tenha melhor entendimento do texto, os nomes e expressões técnicas introduzidos neste Capítulo, foram incluidos no Glossário do trabalho.

2.1.1. DESENVOLVIMENTO DE UMA FERRAMENTA DE "SOFTWARE" PRÓPRIA

As alternativas dessa linha corresponderiam a uma continuidade da tendência que se tem verificado nos últimos anos no ambiente de desenvolvimento de sistemas que é a de se procurar desenvolver ferramentas próprias para o atendimento de necessidades específicas. São frutos deste esforço as ferramentas apresentadas no Capítulo 1, a estrutura de dados DB VSAM, além de outras menos representativas.

As ferramentas propostas nesta linha são funcionalmente muito semelhantes à algumas que já foram desenvolvidas como o SCOA e o Gerador de Relatórios, pois a sua função será a de gerar ou auxiliar regeração de programa fonte[1] de programas "batch". Nesse caso também, o programa fonte seria codificado na linguagem COBOL de modo que se pudesse aproveitar

---

[1] Talvez não completamente, mas pelo menos uma grande parte.

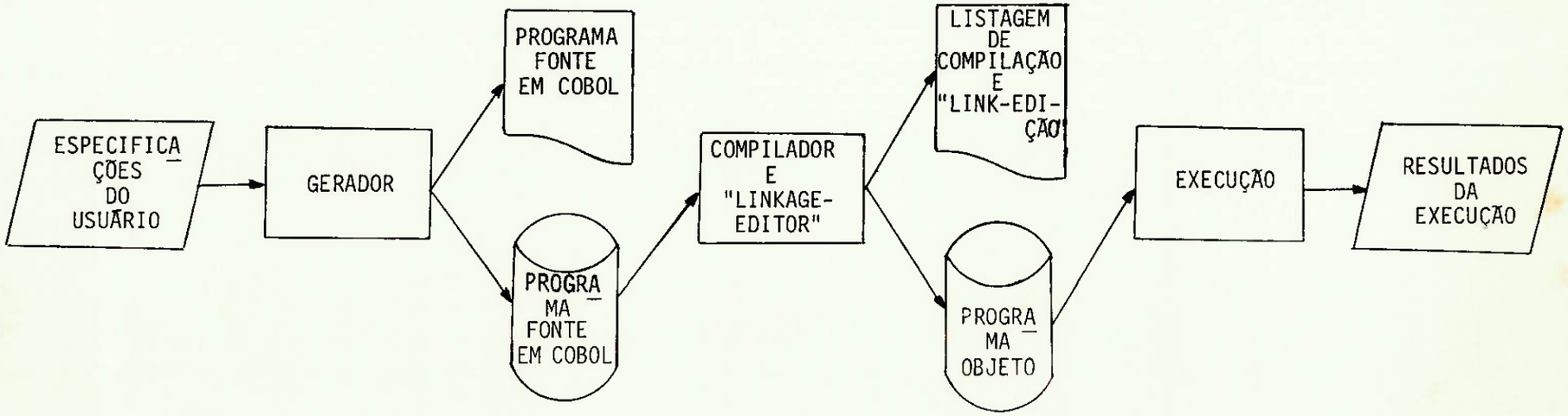


Figura 2.2 - Esquema de utilização de um gerador de programas fonte para aplicações "batch"
- e.p.a. -

o compilador e o "linkage-editor" dessa linguagem para se obter o corresponde programa objeto executável em máquina. Esse processo é ilustrado na figura 2.2. A sua diferença básica com o da atual situação (processo convencional) consiste na inclusão do processamento referente à geração do programa fonte.

Dentre as alternativas consideradas nessa linha, nenhuma diz respeito ao desenvolvimento completo de uma nova linguagem de mais alto nível, voltada à aplicações "batch", incluindo até a construção de um compilador e "linkage-editor" ou interpretador para a sua conversão em linguagem de máquina. Um projeto desta envergadura é algo muito mais complexo, ambicioso, caro e demorado, provavelmente não coberto pela capacidade técnica ("know-how") acumulada pela equipe de desenvolvimento de sistemas.

As alternativas[1] que se enquadram nessa linha são apresentadas a seguir.

A. Alternativa 1 - Desenvolvimento de uma ferramenta de "software" com características procedurais (codificação manual instrução por instrução)

Os programas "batch" passariam a ser codificados numa versão própria da linguagem COBOL mais sofisticada e de mais alto nível. Assim, os programadores e analistas teriam em mãos uma nova ferramenta que além de incorporar a linguagem COBOL, ofereceria poderosos recursos adicionais como comandos mais potentes. Na prática, conforme a adaptabilidade da ferramenta ao programa "batch" que se pretender desenvolver, os usuários teriam duas alternativas de se usar o gerador: codificá-lo por completo usando-se somente os recursos adicionais da ferramenta (melhor caso) ou, codificá-lo parcialmente, completando o resto com os recursos convencionais da linguagem

---

[1] Doravante referenciadas apenas pela sua numeração.

COBOL. O projeto dessa ferramenta já levaria em consideração uma série de fatores ambientais tais como, acesso a arquivos do DB VSAM, acesso ao Dicionário de Dados, padrões adotados, etc.

Convêm salientar que há bons motivos para que a ferramenta proposta baseie-se na linguagem COBOL (vide ref. 3).

- Atualmente no Brasil cerca de 85% dos programas para fins administrativos e/ou comerciais são codificados em(COBOL) grande domínio e popularidade) e apesar da penetração de inúmeros produtos lançados ultimamente no mercado, pelo menos a médio prazo, a participação das ferramentas convencionais (das quais o COBOL é o exemplo mais representativo) continuará a ser significativa como mostra a figura 2.3.

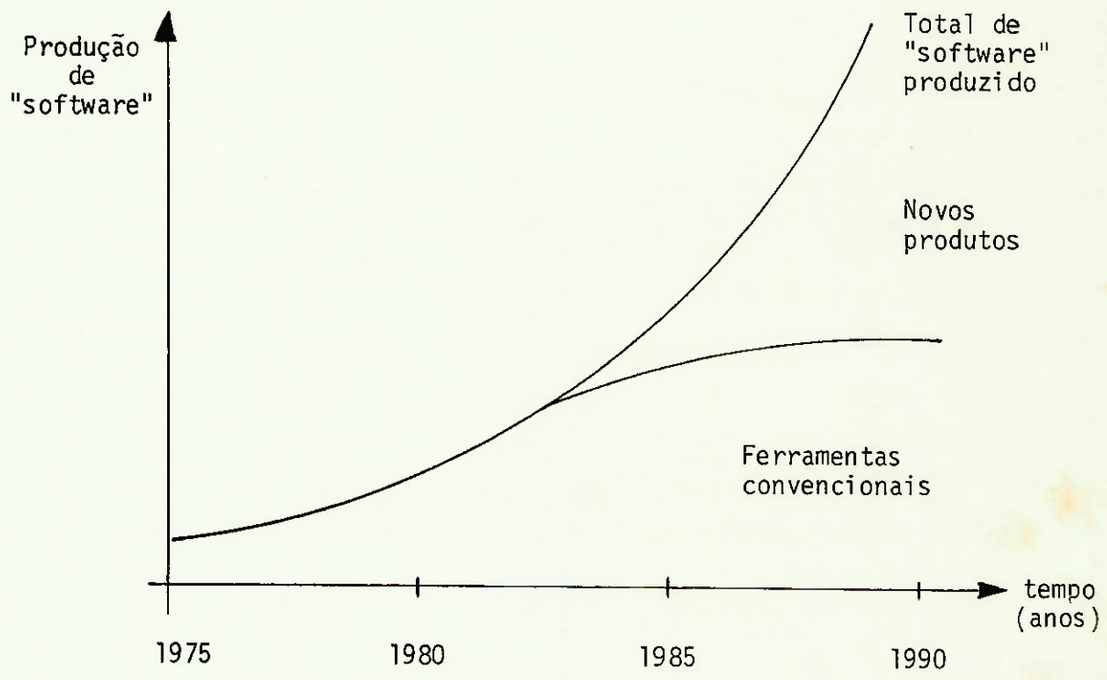


Figura 2.3 - Estimativa da participação das ferramentas convencionais no total de "software" a ser desenvolvido nos próximos anos
- transcrito da ref. 3 -

- As linguagens convencionais como o COBOL e PL/1, foram criadas antes do advento da tecnologia "on-line" e são, portanto, até denominadas de linguagens "batch" por se voltarem, principalmente, às aplicações "batch".

- As linguagens convencionais terão uma participação menor na produção de "software" mas serão empregadas em aplicações específicas como programas "batch" ou outras que não sejam atendidas satisfatoriamente pelas novas ferramentas.

Então, com base nessas perspectivas, o que esta alternativa sugere seria a agilização da utilização da linguagem COBOL em aplicações do tipo "batch".

B. Alternativa 2 - Desenvolvimento de uma ferramenta de "software" própria com características não-procedurais

Esta alternativa é muito semelhante à anterior e praticamente tudo que se apresentou a seu respeito continua válido. A diferença fundamental está na técnica de se fornecer as especificações necessárias para a geração do programa fonte. Na alternativa 1, isso é feito via uma linguagem procedural como se o usuário estivesse codificando normalmente o seu programa, e a alternativa 2, propõe que isso seja feito não-proceduralmente, via telas específicas em transações "on-line" (ex.tela para apresentação de tabelas de decisão, fluxogramas, relacionamentos com arquivos,etc.). Se os novos recursos oferecidos pela ferramenta não forem suficientes para se gerar completamente um programa, o usuário deverá examinar primeiro o programa fonte a ser gerado para localizar os pontos onde deveriam ser inseridos códigos complementares ("patches") e em seguida digitá-los para que o gerador receba as complementações necessárias e crie a versão completa. Todo esse processo seria equivalente ao uso atual do SCOA (vide Apêndice A) para a geração de programas "on-line". Vimos que no caso anterior (alternativa 1), a complementação poderia ser feito proceduralmente (com códigos em COBOL) quando da codificação do próprio programa original.

2.1.2. INTRODUÇÃO DE FERRAMENTAS DESENVOLVIDAS POR TERCEIROS

Nesta linha encontram-se alternativas que propõem a compra de uma ferramenta de "software" já desenvolvida por outras empresas que atuam nessa área, acompanhada ou não de outras ferramentas de apoio. Evidentemente, o vínculo com o fornecedor não se limitaria somente à compra do(s) produto(s) propriamente dito(s). Seria necessário também, contar com os seus serviços de suporte e de manutenção, fornecimento de outras versões do(s) produto(s), treinamento, etc.

As alternativas que se enquadram nessa linha são apresentadas na seqüência:

Alternativa 3[1] - Introdução de uma linguagem de quarta geração apoiada em algum banco de dados munido de um Sistema Gerenciador de Banco de Dados (SGBD)

Com o advento da tecnologia "on-line", foram desenvolvidas as chamadas linguagens de quarta geração, que são linguagens de altíssimo nível, traduzidas para código de máquina por interpretadores (não por compiladores), dotadas de interessantes recursos não convencionais e, em geral, são do tipo não procedural. São comumente chamadas de ferramentas para o desenvolvimento de aplicações "on-line" (vide ref. 3) pois vieram a preencher, notadamente, essa parcela do mercado de produtos de "software", que estava sendo coberta, insatisfatoriamente, por linguagens "batch" como o COBOL, apoiados em comandos especiais de teleprocessamento como os do monitor de teleprocessamento CICS.

O que esta alternativa propõe é apenas o aproveitamento dos recursos para programação "batch" oferecidos por uma dessas ferramentas e implantar um Banco de Dados (com todos

---

[1] Daremos seqüência a numeração das alternativas iniciada no subitem 2.1.1.

os seus demais elementos de apoio, tais como o SGBD, Dicionário de Dados, Diretório de Dados e fitas para LOG[1] de atualização) que pudesse ser acessado e manipulado pela mesma. Convém lembrar que, todo esse conjunto de ferramentas de "software" seria direcionado apenas para se tentar facilitar o desenvolvimento de aplicações "batch", já que não há interesse em aproveitá-lo também no desenvolvimento de aplicações "on-line", o que, como vimos, já está sendo muito bem atendido pelas ferramentas "próprias" do ambiente. A figura 2.4 ilustra como seria a integração de todo esse conjunto, onde a linguagem de quarta geração proposta não aparece explicitamente, mas apenas o seu produto imediato, no caso, os programas de aplicação "batch". Observe que nessa figura aparece a participação do Administrador de Banco de Dados. Este é o elemento (ou uma

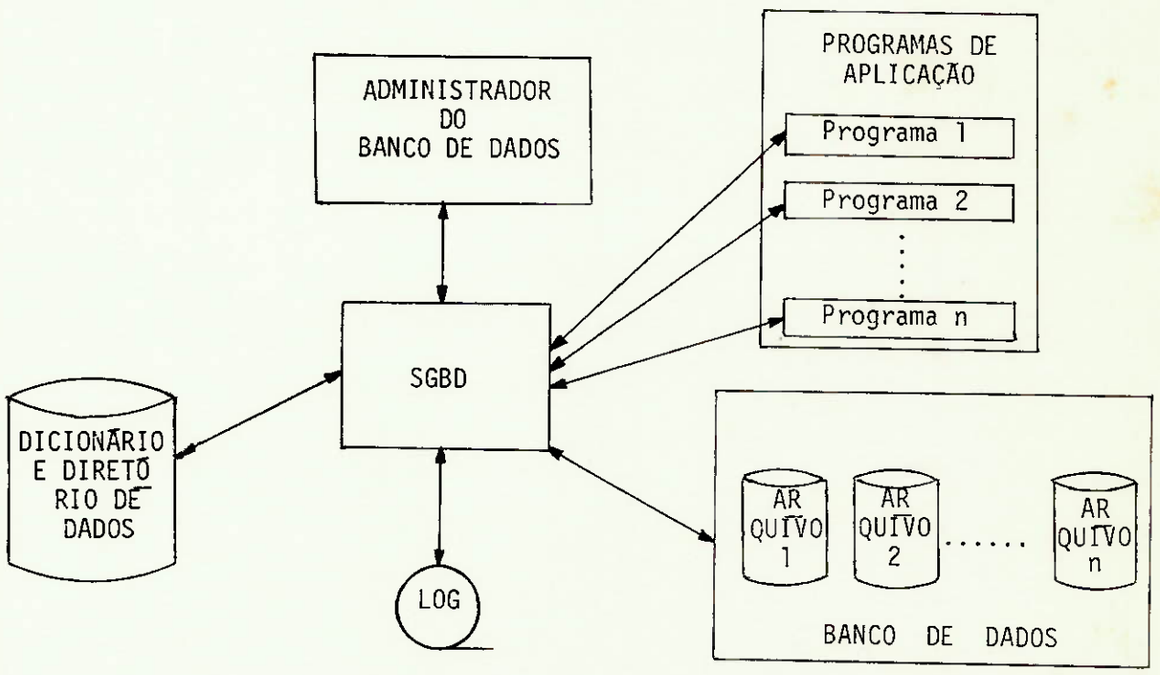


Figura 2.4 - Integração das ferramentas de "software" propostas pela alternativa 3
- adaptado da ref. 3 -

---

[1] Gravação das imagens anterior e posterior a uma modificação para a garantia da integridade dos dados (vide ref. 1).

equipe) que acompanha, controla e apoia a utilização do SGBD, dividindo com este as tarefas de gerência do Banco de Dados. Como a empresa não conta com tal profissional (ou equipe), este teria de ser contratado. A tabela 2.1 apresenta alguns produtos já lançados no Brasil que poderiam compor esta alternativa. Observe-se que todos são compatíveis com o sistema operacional OS/VS1, monitor de teleprocessamento CICS e método de acesso VSAM (que fazem parte da configuração de "software" básico do ambiente).

Alternativa 4 - Introdução de um Banco de Dados munido de SGBD, e de uma interface para acessá-lo ("host language interface") conservando a linguagem COBOL

Esta alternativa é muito semelhante à anterior, sendo que a maioria dos aspectos lá apresentados permanece válido para este caso. A única diferença está na maneira de se acessar e manipular os dados. Esta alternativa, propõe que isso seja feito mantendo-se o COBOL como a linguagem básica de programação "batch", complementando-a com uma interface para acesso. Existem dois tipos de interface, sendo que qualquer uma delas poderia ser adotada. O primeiro (mais comum) consiste na utilização do comando macro CALL, para chamada de rotina-padrão. Esta recebe os parâmetros de especificação de operação e os dados necessários à execução da operação. O segundo consiste na utilização de comandos de uma linguagem de manipulação de dados específica do sistema misturados com os comandos normais da linguagem COBOL. O programa assim formado deve passar por um pré-processador que transforma os comandos da linguagem de manipulação em chamadas de rotinas-padrão pela macro CALL; a saída deste pré-processador será o programa fonte a dar entrada no compilador da linguagem COBOL. Desta forma, a figura 2.4 continua válida para esta alternativa. O leitor deve apenas estar ciente de que pela alternativa anterior, os programas de aplicação "batch" seriam desenvolvidos por meio de uma linguagem de quarta geração, e que nesta não. A tabela 2.2 fornece a interface para acesso (todas por meio da macro CALL) a ser utilizado com a linguagem COBOL pa

LINGUAGEM	FORNECEDOR	BANCO(S) DE DADOS ACESSADO(S)	COMPATIBILIDADE COM		
			OS/VS1	CICS	VSAM
Gener/OL	Pansophic	DL-1, IMS	sim	sim	sim
Mantis	Cincom	Total, TIS/DB, CMS, Mantis	sim	sim	sim
Ideal	SCI	Datacom/DB, DL-1	sim	sim	sim
Natural	Consist	Adabas	sim	sim	sim
Ramis II	Princeton	Adabas, DL-1, IDMS, IMS, Total	sim	sim	sim

Tabela 2.1 - Alternativas de linguagens de quarta geração e Banco de Dados para a Alternativa 3 - adaptado da ref. 4 -

ra 3 exemplos de Banco de Dados (vide ref. 1) já lançados no Brasil:

BANCO DE DADOS	FORNECEDOR	ROTINA-PADRÃO DE INTERFACE CHAMADA
Total	Cincom	DATBAS
Adabas	Software Ag	ADABAS
IMS	IBM	CBLTDLI

Tabela 2.2 - Alguns Bancos de Dados já lançados no Brasil - e.p.a. -

Alternativa 5 - Introdução de uma linguagem de quarta geração

Essa alternativa, em comparação com as demais da linha de ação 2, difere substanciamente, pelo fato de dispensar o uso de Banco de Dados e consequentemente, de todos os outros elementos de apoio que o acompanham. Assim continuar-se-ia a armazenar os dados acessados por programas de aplicação "batch" no DB VSAM do ambiente. Esses programas, no entanto, passariam a ser desenvolvidos com os recursos de uma linguagem de quarta geração, apesar destas serem mais indicadas para o desenvolvimento de aplicações do tipo "on-line".

2.2. AVALIAÇÃO DAS ALTERNATIVAS

Procuraremos representar aqui como se deu o processo de avaliação das alternativas de "software". Todo esse processo se deu de forma qualitativa, frente à enorme dificuldade de avaliar as alternativas levantadas mediante métodos e critérios mais fundamentados e rigorosos, baseados em conceitos de

engenharia econômica, custos, mensuração dos possíveis ganhos (benefícios) e perdas, etc. Além da complexidade envolvida num processo dessa natureza, outros fatores contribuiram para torná-lo inviável na prática, quais sejam:

- o tempo necessário para a sua execução;

- o volume de recursos que seria necessário;

- as alternativas da linha de ação 1 propõem uma ferramenta ainda não existente, o que dificulta ainda mais a obtenção de dados quantitativos a seu respeito;

- as alternativas da linha de ação 2, apenas sugerem vários produtos semelhantes sem, no entanto, apresentar qualquer tipo de definição. Num processo de avaliação mais elaborado, seria necessário chegar a uma melhor definição quanto ao produto ou combinação de produtos (caso das alternativas 3 e 4) que melhor atenderia às necessidades do ambiente. Tal tarefa também é das mais complexas, já que, embora as opções sugeridas apresentem a mesma função (a nível global) elas diferem muito entre si quanto as características operacionais, projeto, custos, versatilidade, serviços de suporte, desempenho, etc. (vide ref. 3).

- há muitos fatores intangíveis que devem ser considerados.

2.2.1. CRITÉRIOS DE AVALIAÇÃO

Antes de se proceder a avaliação qualitativa (julgamento) das alternativas, foram estabelecidos os quesitos a serem observados, os quais são apresentados a seguir:

a) Ganho de produtividade proporcionável

Corresponde ao grau de atendimento ao fim a que se destina, ou seja, proporcionar ganho de produtividade no desenvolvimento de programas para aplicação "batch". Esse ganho é o benefício direto mais esperado, pois, a grosso modo, po

deríamos dizer que quanto maior for o ganho de produtividade, melhor o problema será solucionado.

b) Custos envolvidos

Corresponde aos custos mais relevantes, com os quais o Setor de Informática e Serviços de Teleprocessamento teria de arcar para alcançar os benefícios vislumbrados.

c) Adequabilidade ao ambiente e ao problema

Corresponde a capacidade da alternativa de se adequar às características físicas (configuração de "hardware" e de "software", recursos humanos, etc.) e conceituais (filosofia, tática, cultura, padrões, etc.) do ambiente, e portanto, envolve a consideração de uma série de fatores intangíveis. Este quesito servirá também para avaliar a racionalidade e objetividade da alternativa.

Como a avaliação foi qualitativa, estabeleceu-se a seguinte escala de notas conceituais conforme o grau de satisfação de uma alternativa em relação a cada quesito:

A - satisfação ótima

B - satisfação boa

C - satisfação razoável

D - satisfação abaixo do mínimo desejado

E - satisfação nula.

Convém observar que o ganho de produtividade proporcionável e a adequabilidade ao ambiente e ao problema são quesitos que se correlacionam de maneira direta com o grau de satisfação, ou seja, quanto maiores, melhores são as respectivas notas conceituais. Com relação ao quesito custos envolvidos, a correlação é inversa, ou seja, quanto menores, a nota conceitual é melhor.

2.2.2. QUALIFICAÇÃO DAS ALTERNATIVAS

O processo de avaliação envolveu praticamente todo o pessoal envolvido no estudo do problema (vide Capítulo 1) . Os seus resultados (nota conceitual que cada alternativa recebeu em relação aos 3 quesitos de avaliação) estão sumarizados na tabela 2.3. As principais considerações (justificativas)que levaram às qualificações apresentadas nessa tabela, são apresentadas sucintamente na seqüência, também em forma de tabelas (tabelas 2.4 a 2.8).

2.3. SELEÇÃO DA MELHOR ALTERNATIVA

Ainda durante o processo de avaliação, constatou-se claramente que as alternativas da linha 1 eram superiores às da linha 2. Isso foi decorrente, fundamentalmente, dos seus custos envolvidos serem relativamente menores e do fato de as ferramentas por elas propostas poderem ter o seu projeto voltado, diretamente, ao atendimento das necessidades específicas do ambiente, levando ainda em consideração, outras características ambientais e a possibilidade de integração com as outras ferramentas próprias já existentes.

A grande desvantagem das alternativas da linha 2 reside no fato de as mesmas sugerirem produtos com características muito mais genéricas, voltadas a um campo de aplicação mais vasto, do qual parcela considerável já é atendida (e muito bem) por um conjunto de ferramentas próprias, tais como o Sistema "Global", DB VSAM e o Gerador de Programas para Aplicações "on-line" (SCOA), ou seja, carecem de maior objetividade e racionalidade, para poderem constituir uma solução mais pertinente ao problema. Justamente por esta maior satisfação e pelo fato de os produtos sugeridos terem sido desenvolvidos por terceiros (margem de lucro), as alternativas dessa linha envolvem custos maiores.

QUESITO / ALTERNATIVAS	GANHO DE PRODUTIVIDADE PROPORCIONÁVEL	CUSTOS ENVOLVIDOS	ADEQUABILIDADE AO AMBIENTE E AO PROBLEMA
1. Desenvolvimento de uma ferramenta própria com características procedurais	A	A	A
2. Desenvolvimento de uma ferramenta própria com características não-procedurais	B	A	B
3. Introdução de uma linguagem de quarta geração apoiada num Banco de Dados	A	D	C
4. Introdução de um Banco de Dados e de uma interface para acessá-lo por meio do COBOL	C	C	C
5. Introdução de uma linguagem de quarta geração	B	B	B

Tabela 2.3 - Resultados da avaliação qualitativa das alternativas
- e.p.a. -

QUESITO	NOTA CONCEITUAL	PRINCIPAIS CONSIDERAÇÕES
Ganho de produtividade	A	. o gerador aliviaria grande parte do trabalho de programação . o gerador facilitaria o trabalho de depuração dos programas . técnica procedural é mais indicada para aplicações "batch" . com a mecanização do trabalho de codificação, diminuir-se-ia a possibilidade de erros de sintaxe
Custos envolvidos	A	. uma equipe pequena (cerca de 3 pessoas) teria condições de desenvolver a ferramenta num prazo de 10 a 12 meses . custo de consumo de computador para desenvolvimento . custos de implantação e treinamento . custo de consumo de computador para geração de programas(gerador) . uma pessoa em regime de tempo parcial bastaria para os serviços de manutenção
Adequabilidade ao ambiente e ao problema	A	. integração com as demais ferramentas do ambiente . os programas fonte continuam em COBOL . não requer nenhuma alteração da atual configuração de "software" . respeita a orientação: "uma ferramenta para cada situação" . aumenta "know-how" e garante auto-suficiência em manutenção . projeto em cima das necessidades específicas do ambiente

Tabela 2.4 - Avaliação da alternativa 1 (ferramenta própria com características procedurais)
- e.p.a. -

QUESITO	NOTA CONCEITUAL	PRINCIPAIS CONSIDERAÇÕES
Ganho de produtividade	B	. o gerador aliviaria grande parte do trabalho de programação . a medida que aumenta a necessidade de inserção de "patches", a ferramenta perde aplicabilidade (os ganhos diminuem) . técnica não-procedural é mais indicada para aplicações "on-line" . a possibilidade de erros de sintaxe diminui com a mecanização do trabalho de codificação
Custos envolvidos	A	. idem alternativa 1, mais custos operacionais necessários (consumo de computador, terminal de video, outros) para especificação de programas por técnica não-procedural
Adequabilidade ao ambiente e ao problema	B	. integração com as demais ferramentas do ambiente . os programas fonte continuam em COBOL . não requer nenhuma alteração da atual configuração de "software" . respeita a orientação: "uma ferramenta para cada situação" . aumenta "know-how" e garante auto-suficiência em manutenção . projeto em cima das necessidades específicas do ambiente . técnica não-procedural é mais indicada para aplicações "on-line"

Tabela 2.5 - Avaliação da alternativa 2 (ferramenta própria com características não-procedurais) - e.p.a. -

QUESITO	NOTA CONCEITUAL	PRINCIPAIS CONSIDERAÇÕES
Ganho de produtividade	A	. o potencial de uma linguagem de quarta geração apoiada num Banco de Dados é muito grande . linguagem de quarta geração é mais voltada para aplicações "on-line" . em geral envolve técnica não-procedural (melhor para aplicações "on-line") . linguagem interpretada facilita testes para depuração de programas
Custos envolvidos	D	. custos de aquisição a implantação de uma linguagem de quarta geração, um Banco de Dados e todos os elementos de apoio necessários . linguagem interpretada acarreta custos maiores de consumo de computador (problemática na fase de operação) . custos para gerenciamento do Banco de Dados (Administrador de B.D.) . custos de consumo de computador para manipulação do Banco de Dados . custos de treinamento
Adequabilidade ao ambiente e ao problema	C	. o ambiente já conta com ferramentas próprias mais adequadas que o Bando de Dados e todos os seus elementos de apoio . não integração com as demais ferramentas do ambiente . problema de sintonia entre o SGBD e o monitor de TP CICS (experiência) . não respeita a orientação: "uma ferramenta para cada situação" . dependência de serviços de suporte de terceiros . os produtos sugeridos ainda estão em fase de penetração (vulnerabilidade) . despadronização mediante introdução de ferramentas redundantes

Tabela 2.6 - Avaliação da alternativa 3 (linguagem de quarta geração apoiada num Banco de Dados) - e.p.a. -

QUESITO	NOTA CONCEITUAL	PRINCIPAIS CONSIDERAÇÕES
Ganho de produtividade	C	. os ganhos de produtividade limitar-se-iam à lógica da manipulação de dados armazenados no Banco de Dados . técnica procedural é mais indicada para aplicações "batch"
Custos envolvidos	C	. custos de aquisição e implantação de um Banco de Dados e de uma interface para acesso por meio da linguagem COBOL . custos de consumo de computador para pré-processamento do programa . custos de gerenciamento de Banco de Dados (Administrador de B.D.) . custos de consumo de computador para manipulação do Banco de Dados . custos de treinamento
Adequabilidade ao ambiente e ao problema	C	. o ambiente já conta com ferramentas próprias mais adequadas que o Banco de Dados e todos os seus elementos de apoio . não integração com as demais ferramentas do ambiente . problema de sintonia entre o SGBD e o monitor de TP CICS (experiência) . a lógica de manipulação de dados é apenas parte do problema . despadronização mediante introdução de ferramentas redundantes . dependência de serviços de suporte de terceiros

Tabela 2.7 - Avaliação da alternativa 4 (linguagem COBOL associada a uma interface para se acessar um Banco de Dados)

- e.p.a. -

QUESITO	NOTA CONCEITUAL	PRINCIPAIS CONSIDERAÇÕES
Ganho de produtividade	B	. linguagem de quarta geração perde muito de sua potencialidade quando não apoiada em Banco de Dados . linguagem de quarta geração é mais voltada para aplicações "on-line" . em geral envolve técnica não procedural (melhor para aplicações "on-line") . linguagem interpretada facilita testes para depuração de programas
Custos envolvidos	B	. custos de aquisição e implantação de uma linguagem de quarta geração . custos de treinamento . custos operacionais (consumo de computador, terminal de video, outros) para desenvolvimento de programas por técnica não procedural . linguagem interpretada acarreta custos maiores de consumo de computador (problemática na fase de operação)
Adequabilidade ao ambiente e ao problema	B	. não integração com as demais ferramentas do ambiente . não respeita a orientação: "uma ferramenta para cada situação" . dependência de serviços de suporte de terceiros . os produtos sugeridos ainda estão em fase de penetração (vulnerabilidade) . sub-aproveitamento da ferramenta já que é mais voltada para aplicações "on-line"

Tabela 2.8 - Avaliação da alternativa 5 (linguagem de quarta geração)
- e.p.a. -

Convencido de que a melhor linha de ação a ser seguida era realmente a de se desenvolver uma ferramenta própria, o pessoal envolvido no processo de avaliação acabou selecionando a alternativa 1. A alternativa 1, em comparação com as demais foi superior em todos os quesitos e foi selecionada, em detrimento da alternativa 2 (outra alternativa da linha de ação 1), pelo fato de sugerir uma técnica de desenvolvimento de programas procedural, mais adequada a programas do tipo "batch".

A alternativa 1, selecionada como a melhor dentre as alternativas de "software", é a que será analisada no decorrer dos próximos Capítulos.

3. O META-COBOL

A alternativa escolhida, já foi superficialmente descrita no subitem 2.2.1., entretanto, procuraremos complementar a sua apresentação de modo que o leitor possa ter uma idéia melhor do que virá a ser a ferramenta de "software" selecionada. Não entraremos em detalhes de como a ferramenta seria viabilizada tecnicamente (características e "arquitetura" do "software" que teria de ser desenvolvido) pois isso é problema de natureza mais técnica e será resolvido pela equipe encarregada do detalhamento do projeto. De qualquer forma, foi necessário consultar estas pessoas de modo que pudéssemos coletar as informações necessárias para a elaboração do presente Capítulo.

Apresentaremos no item 3.1. uma breve complementação a respeito dessa ferramenta e no item 3.2. um exemplo de aplicação. As pessoas não familiarizadas com programação, e em particular com a linguagem COBOL, talvez encontrem dificuldades para entenderem as considerações aqui apresentadas (1). Mas isso não prejudicará o entendimento dos próximos Capítulos onde trataremos especificamente dos impactos que se verificaram no ambiente da Programação com a eventual implantação dessa ferramenta. O único aspecto que merecerá maior atenção é a potencialidade dessa ferramenta em proporcionar grandes ganhos de produtividade em desenvolvimento de programas "batch" o que poderá ser visualizado, observando o exemplo de aplicação que apresentaremos.

3.1. DESCRIÇÃO

A ferramenta que foi selecionada, por pretender ser uma espécie de agilizador do uso da linguagem COBOL, foi

---

(1) Aos leitores que tiverem maior interesse por uma descrição mais apurada, aconselhamos consultar livros especializados a respeito como "Programação COBOL" de Alex C.Bastos.

batizada de Meta-Cobol[1]. O prefixo "Meta" vem do idioma grego, e significa "além de", ou seja, o Meta-Cobol seria algo "além do COBOL".

Assim como o COBOL é uma ferramenta constituída de uma linguagem e de um compilador e "link-editor" para transformar o programa fonte codificado nessa linguagem em programa executável pelo computador; o Meta-Cobol seria constituído de uma "meta-linguagem" (linguagem COBOL mais "meta-comandos", ou seja, comandos adicionais de grande potencialidade) e de um pré-compilador (ou pré-processador) para processar o programa originalmente codificado nessa meta-linguagem, e a partir do mesmo, gerar o correspondente programa fonte na linguagem COBOL. Uma vez obtido esse programa fonte, bastaria dar seqüência ao procedimento convencional de compilação e "link-edição" (vide figura 2.2).

Como vimos no Capítulo 2, um dos pontos fortes do Meta-Cobol seria a sua adequabilidade ao ambiente. No ambiente do Setor de Informática e Serviços de Teleprocessamento, a maioria dos Sistemas de Informações que têm sido desenvolvidos e os planejados para um futuro próximo, são notadamente "on-line", ou seja, a entrada de dados é feita através de transações "on-line". Aos programas "batch" tem sido reservadas funções como listagens (relatório) e atualização de determinados campos dos registros de determinados arquivos. Funções como consistências de dados e emissão de relatórios de inconsistências, são muito raras para os programas "batch" . Então, o desenvolvimento do Meta-Cobol se daria a partir dessas características do ambiente. Não sendo possível incorporar todos os recursos idealizados para o Meta-Cobol, logo na sua primeira versão, tratar-se-ia de dar prioridade, àqueles meta-comandos que atenderiam as necessidades mais básicas do ambiente.

---

[1] Doravante, sempre referiremo-nos à ferramenta selecionada pelo seu nome "Meta-Cobol".

Apresentamos a seguir as principais características que foram idealizadas para o Meta-Cobol.

A. Codificação da Meta-Comandos Juntamente com os Comandos Normais do COBOL

Como vimos, o Meta-Cobol não viria a constituir por si só uma linguagem de programação propriamente dita. Ele seria "apenas" um conjunto de poderosos recursos adicionais. Ocorrerão casos em que se poderá especificar toda a lógica do programa a ser gerado, apenas com tais recursos adicionais do Meta-Cobol. Mas, como não se trata de uma ferramenta universal, em certos casos haverá necessidade de o usuário complementar a especificação do programa, codificando normalmente em comandos do COBOL, as partes que o Meta-Cobol não estaria preparado a gerar.

B. Meta-Comandos

Os meta-comandos que serão oferecidos pelo Meta-Cobol, têm os seguintes objetivos principais:

a) proporcionar comandos de altíssimo nível (maior potencialidade que os comandos normais do COBOL).

b) ser de fácil e rápida assimilação (necessidade de pouco treinamento).

c) facilitar a codificação da lógica do programa.

d) eliminar ou diminuir (dependendo do caso) a necessidade de o usuário se preocupar com partes do programa fonte que pouco representam com relação à resolução do problema em si[1], ou seja, tornar o trabalho de programação mais objetivo.

---

[1] Para se programar em COBOL, é necessário dispender considerável parte do tempo total de desenvolvimento, em documentação, definição de arquivos e especificação das estruturas de dados, preparando a codificação da "Identification Division", "Environment Division" e "Data Division". Além disso, até algumas partes da "Procedure Division", são constituídas de códigos que pouco representam à resolução do problema em si.

e) propiciar maior independência entre o programa e a estrutura de dados do DB VSAM (livrar os programadores das particularidades envolvidas no acesso ao DB VSAM).

f) agilizar o aproveitamento do Dicionário de Dados implantado no ambiente.

g) melhorar a documentação do programa fonte (o programa original que seria codificado em meta-linguagem, pelo seu enfoque objetivo, sintético, abrangente e de fácil entendimento da lógica de processamento, já seria um documento de descrição do programa fonte em COBOL).

h) facilitar a comunicação entre os analistas (que formulam a função do programa) e os programadores (que recebem a incumbência de desenvolver um programa com a função formulada pelo analista responsável) através de uma ferramenta comum[1].

C. Geração do Programa Fonte

Para que o pré-compilador tenha condições de "interpretar" o programa fonte e detectar entre os diversos comandos, aqueles que seriam meta-comandos e os que seriam comandos normais do COBOL, é necessário adotar alguns padrões. Os principais padrões de codificação, que foram elaborados observando-se a estrutura de programação com a linguagem COBOL, seriam os seguintes:

a) Os comandos normais do COBOL devem ser codificados normalmente como em qualquer programa fonte.

b) Os meta-comandos poderão ser codificados a partir de qualquer coluna entre as posições 12 e 72 (a estética é por conta do usuário), e no caso de se exceder uma linha, (passar da coluna 72), a continuação poderá se dar na próxima linha (normalmente como na sintaxe do COBOL).

---

[1] Os analistas poderiam se basear na meta-linguagem do COBOL para expressar a lógica do processo que se deseja incurtir no programa a ser desenvolvido. Hoje, na falta de qualquer tipo de padronização, cada um adota o seu próprio estilo de comunicação, o que tem dificultado o trabalho dos programadores.

c) O fim da especificação de um meta-comando deve ser assinalado com um caracter ";" (os comandos normais do COBOL são encerrados por um ".").

d) Os meta-comandos serão diferenciados dos comandos normais do COBOL, por serem precedidos por um caracter de identificação, por exemplo, o " * " (asterisco).

Com base nesses padrões, o pré-compilador do Meta-Cobol seria desenvolvido para processar linha por linha o programa original em meta-linguagem, e sempre que detectasse um meta-comando pelo seu caracter especial, trataria de buscar em arquivos próprios, rotinas padrões em comandos normais do COBOL que teriam função equivalente, e então, providenciaria automaticamente a sua expansão (copiar e inserir) no programa fonte, em substituição ao meta-comando originalmente codificado. Evidentemente, não seria uma expansão imediata. Antes, o pré-compilador teria de preencher a rotina padrão pré-pronta, com os nomes particulares de arquivos, nomes de campos de dados, chaves de acesso, valores de comparação e outras especificações fornecidas pelos meta-comandos, particularizando-a para uma determinada aplicação.

O programa fonte em COBOL a ser gerado seria então um conjunto formado pelos comandos normais que já faziam parte do programa original (que podem existir ou não) e pelas rotinas que seriam expandidas pelo pré-compilador.

Na listagem de saída do programa, viriam ambos os programas quais sejam: o original em meta-linguagem e o fonte gerado em COBOL. O programa original serviria então como um "resumo" do gerado, melhorando em muito a sua documentação (atualmente isso é um problema crônico).

D. As Funções da Meta-Linguagem

Já vimos que seriam oferecidos diversos meta-comandos para exercer funções de processamento de altíssimo nível. Segundo os idealizadores do Meta-Cobol, essas seriam apenas as chamadas funções "explícitas", ou sejam, aquelas que se

riam referenciadas diretamente (explicitamente) no programa original. O Meta-Cobol, no entanto, oferecerá também funções (recursos) que embora não codificadas claramente no programa, poderiam ser deduzidas através dos comandos normais em COBOL ou dos meta-comandos.

Essas funções implícitas, estão muito relacionadas com a preocupação dos idealizadores do Meta-Cobol em destinar aos programadores, somente a tarefa de se preocupar exclusivamente com os códigos "produtíveis" do programa, ou seja, as partes da "Procedure Division" (onde se especifica os passos a serem tomados pelo programa) que representam a lógica de resolução do problema em si.

Desta forma, embora não requisitado explicitamente pelos programadores, o pré-compilador se encarregaria de gerar automaticamente, a "Identification Division", a "Environment Division" e a "Data Division". Essas divisões de um programa em COBOL, por sua vez são constituídas de seções. Dentre as mesmas, aquelas que seriam tratadas com maior importância são as seguintes:

a) File Control

Seção da Environment Division onde é feita a especificação dos meios de entradas e saídas a serem requisitados pelos diversos arquivos do programa (fitas, discos, impressora, etc.).

b) File Section e Working-Storage Section

Seções da Data Division onde se especificam o tipo e o formato dos dados de arquivos (entrada e saída) e dos dados de trabalho utilizados. Vulgarmente, costuma-se dizer que são seções onde se especifica o "lay-out" das estruturas de dados.

c) Report Section

Seção da Data Division, onde se codifica as especificações do relatório a ser impresso pelo programa (somente

para programas "batch" de emissão de relatórios).

Para se gerar as seções dadas em a e em b o pré-compilador se baseará nos próprios nomes de arquivos ou de campos (dados) de trabalho que seriam mencionados no decorrer da Procedure Division. Com base então nesses nomes, o pré-compilador trataria de consultar os arquivos do Dicionário de Dados do ambiente ("depósito" onde os analistas, durante a etapa de projeto dos Sistemas de Informação, armazenam a descrição da natureza dos dados envolvidos) e retirar de lá todas as informações necessárias para gerar tais seções. Alguns campos para dados de trabalho poderão não estar cadastrados nos arquivos, pois lidam com dados temporários que são usados apenas durante o processamento. Isso deverá ser informado pelo programa ao pré-compilador pelo uso do prefixo "XWS" no nome de tais campos. Assim a sua especificação (declaração na Working-Storage Section) seria gerada automaticamente, sem que o pré-compilador consultasse o Dicionário de Dados.

Somente o "lay-out" das estruturas de dados não cadastradas pelos analistas, teriam de ser codificados normalmente pelo programador. O pré-compilador teria então, a capacidade de distinguir dados cadastrados dos não cadastrados (especificação junto com o programa original) e dar o tratamento adequado para cada caso. Observe-se que desta forma, dependendo do caso, a geração dessas seções seria total (a maioria dos casos) ou parcialmente feita pelo pré-compilador.

Outro interessante recurso que o Meta-Cobol irá oferecer, é a inicialização automática de registros com zeros ou espaços em branco, conforme o tipo de seus campos. Isso tem como finalidade, evitar que ocorram "abortos" de programas em execução, devido a erros de lógica que poderiam ser evitados se os registros já tivessem sido devidamente inicializados, antes de se partir para qualquer tipo de operação. São erros fáceis de se evitar e que são bem frequentes no ambiente de desenvolvimento de Sistemas de Informação.

A figura 3.1 mostra de maneira bem simplificada como se daria todo esse processo de pré-compilação e geração do programa fonte em COBOL.

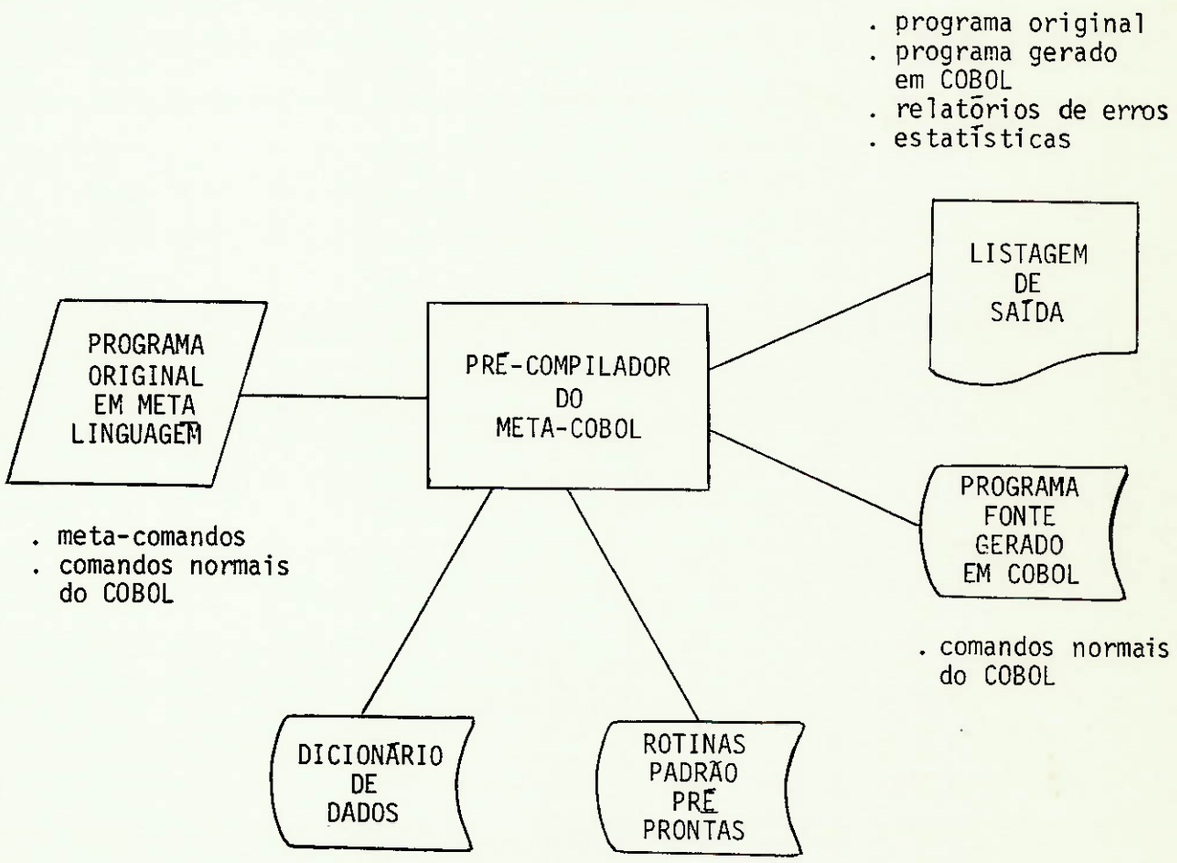


Figura 3.1 - Processo de pré-compilação do programa original para geração de programa fonte em COBOL

- e.p.a. -

Observe-se que, dispensando o trabalho de digitação, o programa fonte gerado, já seria automaticamente gravado num arquivo em disco (caso o pré-processamento transcorra normalmente

sem se verificar qualquer condição de erro) para poder ser submetido à compilação e "link-edição".

E. Integração com as Demais Ferramentas Próprias do Ambiente

Este aspecto também pesou muito na fase de avaliação das alternativas. O Meta-Cobol, sendo uma ferramenta a se adequar às características do ambiente deverá ser desenvolvido prevendo-se a sua integração com outras ferramentas já implantadas. A figura 3.1 mostra apenas uma relação mais direta com o Dicionário de Dados do ambiente. No entanto, o Meta-Cobol estará indiretamente ligado ao DB VSAM, já que ofereceria meta-comandos com a função de tornar mais fácil o acesso aos seus dados (independência entre a estrutura dos dados e o programa) e também, acabaria incorporando o Gerador de Relatórios (caso particular de programas "batch"), já que teria também a função de gerar as especificações de relatório na seção da Report-Section, tarefa esta, por sinal, muito meticulosa e desgastante para o programador. Como podemos observar, com esta integração, cada ferramenta pode passar a complementar uma outra, e ao mesmo tempo, consegue-se uma diluição dos investimentos.

F. Outras Funções do Meta-Cobol

A princípio, o aproveitamento do Meta-Cobol seria destinado ao desenvolvimento de programas "batch". Para tanto, o pré-compilador incurtiria no programa fonte a ser gerado, alguns passos adicionais de modo a se facilitar a depuração do programa em caso de erros de lógica (detecção de erros verificando listagens do conteúdo da memória no instante do aborto da execução do programa), os quais não comentaremos aqui pois envolvem conceitos muito sofisticados da linguagem COBOL.

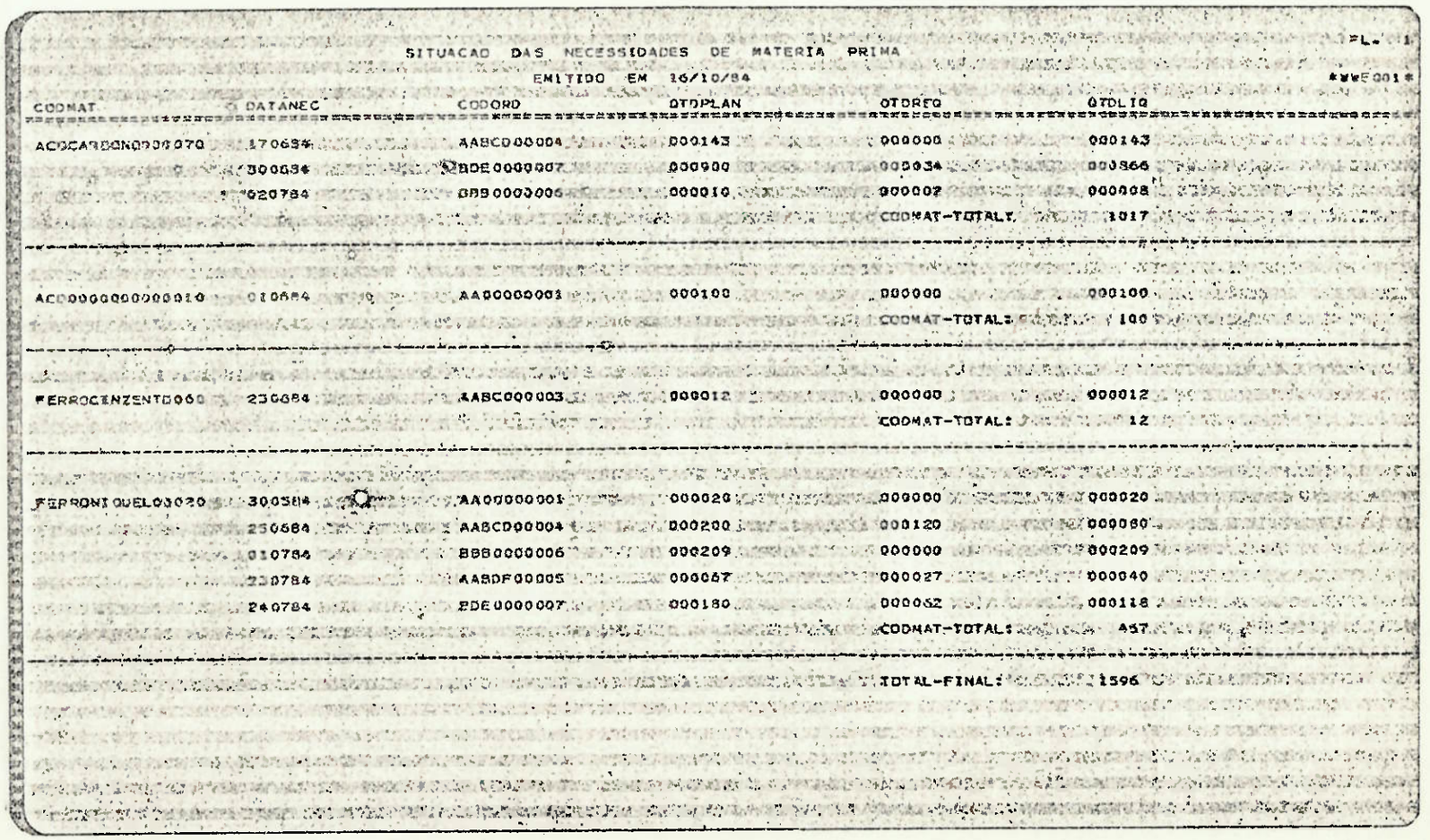
Além disso, o Meta-Cobol poderia também acabar sendo aproveitado nos trabalhos de manutenção de programas (ou seja, ao invés de aproveitar os códigos COBOL já existentes e

"remendá-los", refazer em Meta-Cobol). Esta possibilidade existe, mas nem mesmo os idealizadores do Meta-Cobol têm formada alguma opinião a respeito de como e em que intensidade, isso se daria na prática, e se viria a ser realmente vantajoso para a maioria dos casos de manutenção (geralmente as alterações são de simples a média complexidade, facilmente executáveis sobre o próprio programa fonte). Portanto , diante dessas incertezas, não consideraremos neste trabalho esta hipótese e assumiremos que o Meta-Cobol, seria usado exclusivamente para o fim a que foi idealizado, ou seja, agilizar o desenvolvimento de programas "batch" em COBOL. Isso não deverá prejudicar os resultados de uma eventual avaliação favorável do Meta-Cobol, já que estamos admitindo o pior caso (subaproveitamento da ferramenta).

3.2. EXEMPLO DE APLICAÇÃO DO META-COBOL

Para que o leitor tenha uma melhor idéia do que viria a ser o Meta-Cobol, a meta-linguagem e os respectivos meta-comandos, apresentamos um exemplo de aplicação dos mesmos na resolução de um problema muito comum no setor industrial.

O problema que adotamos como exemplo, é o de se emitir um relatório como o que é mostrado na figura 3.2 . Este relatório entitulado de "Relatório da Situação das Necessidades de Matéria-Prima", seria destinado ao levantamento do total de cada tipo de matéria-prima que seria necessário para a execução das ordens de produção já emitidas. Para uma dada ordem, a sua necessidade líquida de uma dada matéria-prima, é a diferença entre a quantidade de consumo total da mesma planejada para a sua execução, e a quantidade da mesma que já foi requisitada. Esse problema pode ser resolvido conforme mostra a figura 3.3 desenvolvendo-se um programa "batch"(chamado de WWEØØ1) para processar em lotes, os dados armazenados no Arquivo de Ordens de Fabricação (chamado de WW1ORØ) e no Arquivo de Requisições de Suprimentos (chamado de WW1RQØ) e criar o relatório desejado (chamado de WWEQØ1). A função básica do programa (sem entrar em detalhes) seria a de ler o Arquivo de Ordens de Fabricação e para cada registro de or



SITUACAO DAS NECESSIDADES DE MATERIA PRIMA

FL. 1

EMITIDO EM 16/10/84

WWE001

CODMAT	DATANEC	CODORD	QTDPLAN	QTDREQ	QTDLIQ
ACOCARBON0000070	170684	AABCD00004	000143	000000	000143
	300684	BDE0000007	000900	000034	000866
	020784	BBB0000006	000010	000002	000008
				CODMAT-TOTAL:	1017
ACO0000000000010	010684	AA00000001	000100	000000	000100
				CODMAT-TOTAL:	100
FERROCENZENTO060	230684	AABC000003	000012	000000	000012
				CODMAT-TOTAL:	12
FERRONIQUEL03020	300584	AA00000001	000020	000000	000020
	250684	AABCD00004	000200	000120	000080
	010784	BBB0000006	000209	000000	000209
	230784	AABDF00005	000067	000027	000040
	240784	BDE0000007	000180	000062	000118
				CODMAT-TOTAL:	457
				TOTAL-FINAL:	1596

Figura 3.2 - Relatório da Situação das Necessidades de Matéria-Prima - e.p.a. -

dem encontrado, ele teria de localizar as respectivas requisições de suprimentos no Arquivo de Requisições, e dentre as encontradas separaria apenas as que se referem aos suprimentos de matéria-prima. Com base então, nesses registros ele calcularia as necessidades líquidas para cada ordem, reclassificaria os dados obtidos como se deseja no relatório e, somente então, emitiria os resultados obtidos realizando as totalizações e quebras (saltos) desejados. (vide figura 3.2).

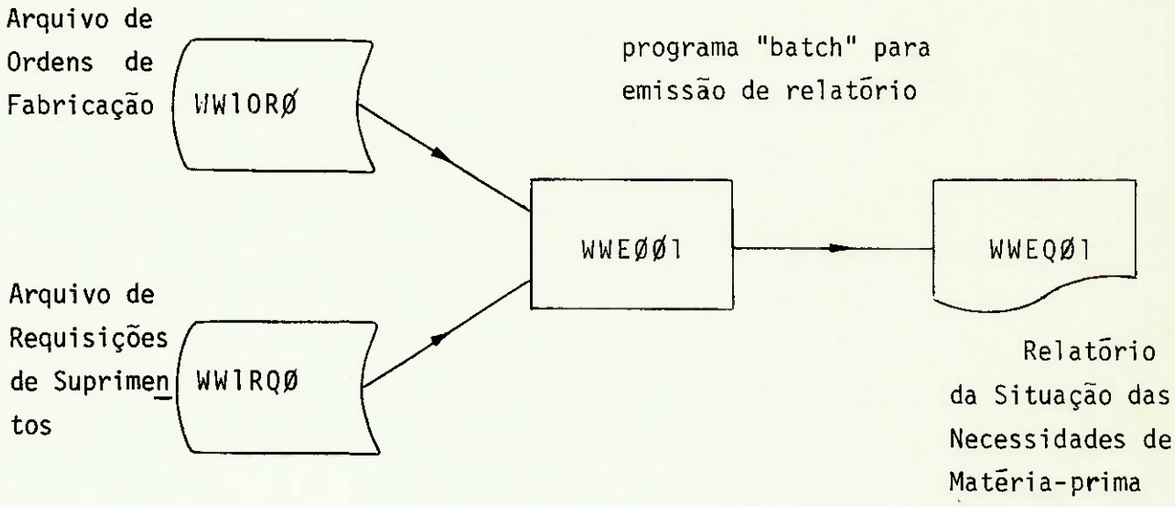


Figura 3.3 - Ilustração da função do programa-exemplo - e.p.a. -

Apresentamos na figura 3.4 como seria o programa original em meta-linguagem[1] que seria fornecido ao pré-compilador do Meta-Cobol, e na figura 3.5 caso seria, aproximadamente, o programa fonte em COBOL que seria gerado. Observe se que a diferença de proporção é realmente muito grande.

---

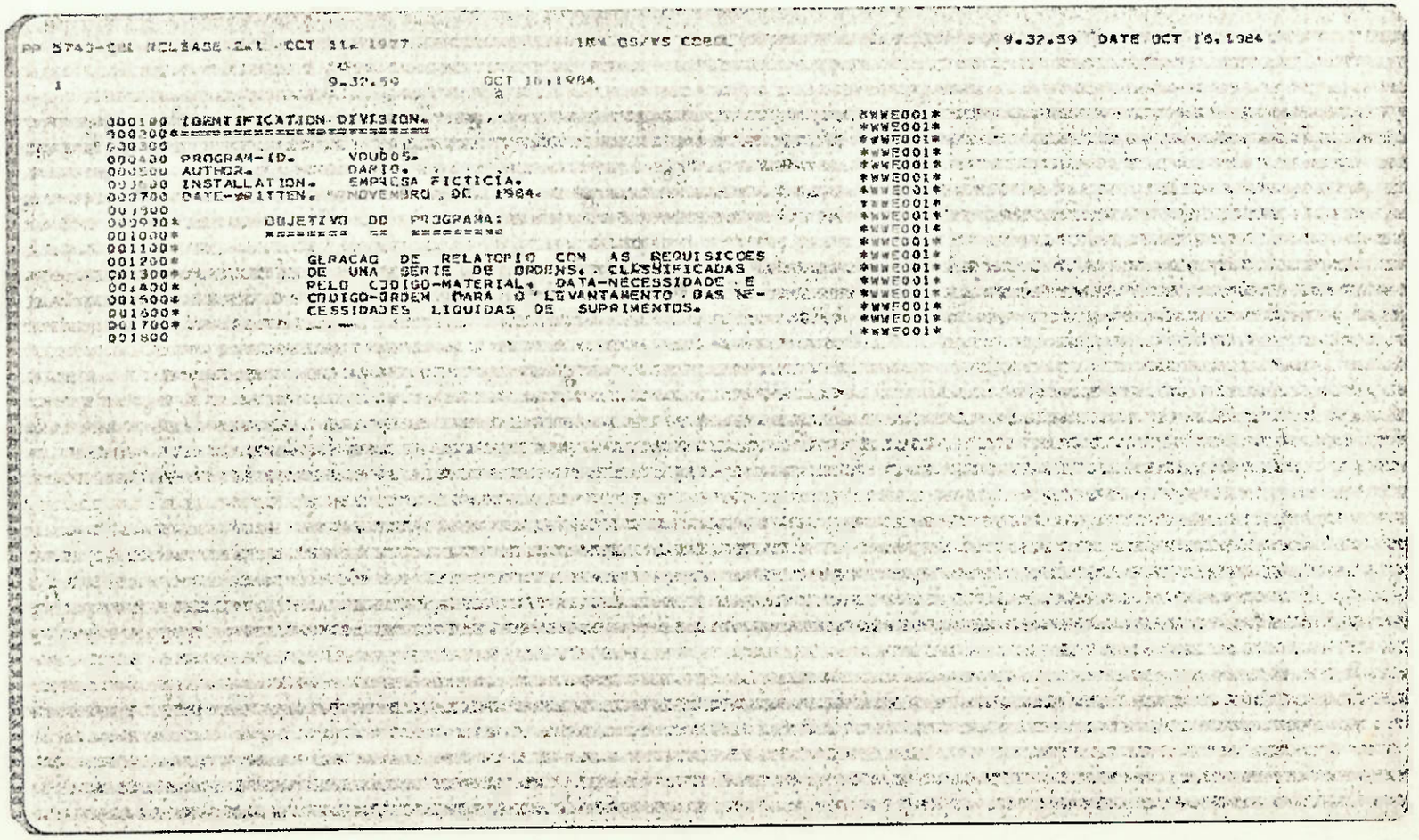
[1] Está claro que ainda não existe o Meta-Cobol, e portanto os meta-comandos aqui apresentados estão apenas idealizados pela equipe de desenvolvimento.

Esse programa que já poderia ser considerado de média complexidade, e que pelos tempos médios que se têm verificado demoraria para ser desenvolvido entre 3 e 6 dias úteis (em concatenação com outras atividades).

```
*START WW1ORØ KEY (FIRST) RETURN (OK);
*EXEC UNTIL ENDFILE WW1ORØ;
   *GET WW1ORØ NEXT
        ENDFILE (EXIT);
   *EXEC UNTIL LOGICAL-END WW1RQØ;
        *GET WW1RQØ NEXT
             VIA (WW1ORØ)
             SELECTION (WW1ZRQØ-TIPO ='MP')
             ENDFILE (EXIT);
        *WRITE WW1EØØ FROM WW1ZRQØ;
   *END-EXEC.
*END-EXEC.

*REPORT GENERATE FROM WW1EØØ
        FIELDS (WW1ZRQØ-CODMAT,
                WW1ZRQØ-DATANEC,
                WW1ZRQØ-CODORD,
                WW1ZRQØ-QTDPLAN,
                WW1ZRQØ-QTDREQ,
                XWS-QTDLIQ)
        TITLE (SITUACAO DAS NECESSIDADES DE MATERIA
PRIMA)
        ORDER BY (WW1ZRQØ-CODMAT,
                  WW1ZRQØ-DATANEC,
                  WW1ZRQØ-CODORD)
        WHERE (XWS-QTDLIQ=WW1ZRQØ-QTDPLAN - WW1ZRQØ
-QTDLIQ)
        ON (WW1ZRQØ-CODMAT) SUMMARIZE
        ON END SUMMARIZE;
```

Figura 3.4 - Programa exemplo codificado em Meta-Cobol - e.p.a. -

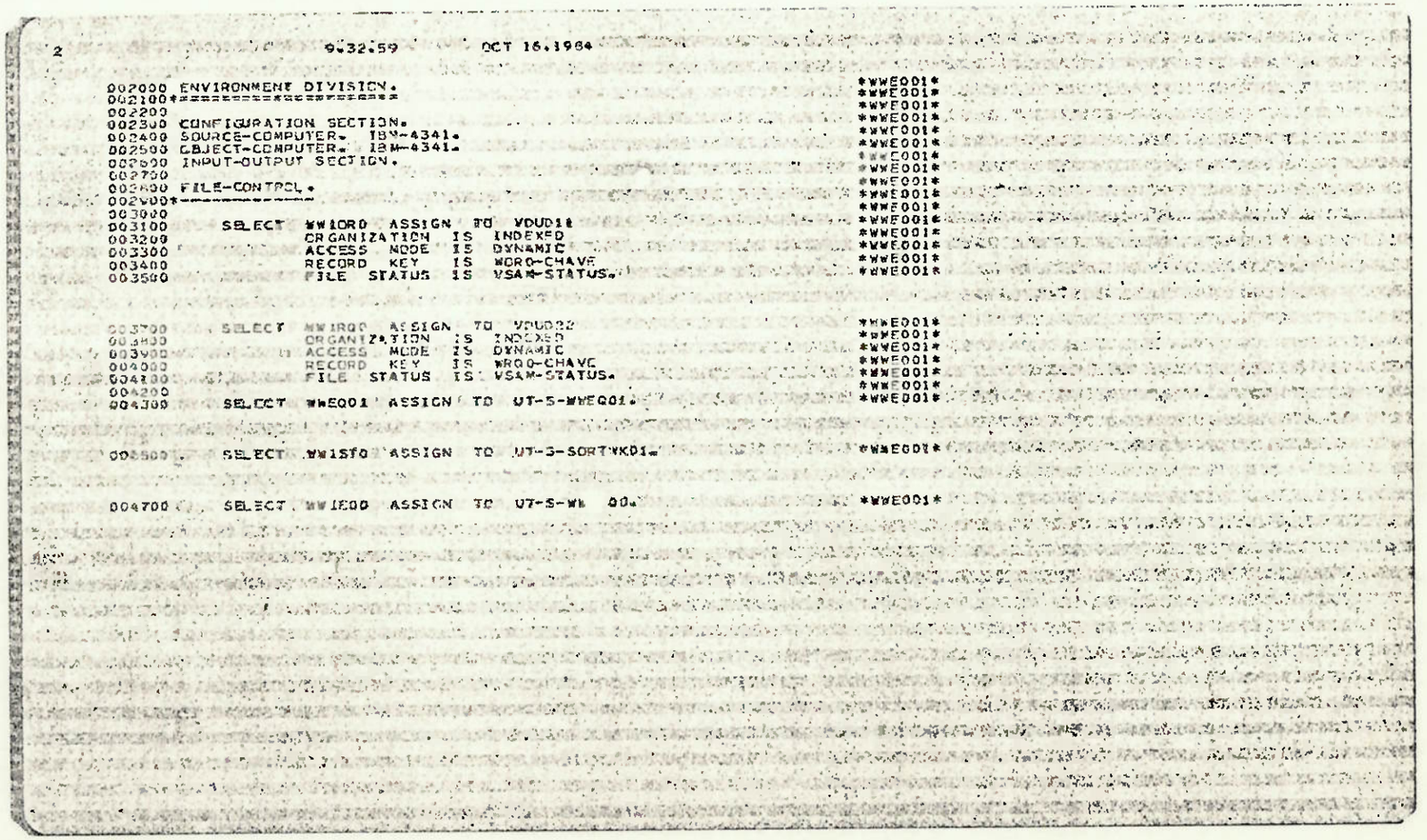


```
5740-CB1 RELEASE 2.1 OCT 31, 1977          IBM OS/VS COBOL                    9.32.59  DATE OCT 16,1984

 1                          9.32.59          OCT 16,1984

000100 IDENTIFICATION DIVISION.                                          *WWE001*
000200*========================                                          *WWE001*
000300                                                                   *WWE001*
000400 PROGRAM-ID.      VOUD05.                                          *WWE001*
000500 AUTHOR.          DARIO.                                           *WWE001*
000600 INSTALLATION.    EMPRESA FICTICIA.                                *WWE001*
000700 DATE-WRITTEN.    NOVEMBRO DE 1984.                                *WWE001*
000800                                                                   *WWE001*
000900*          OBJETIVO DO PROGRAMA:                                   *WWE001*
001000*          ********* == *********                                  *WWE001*
001100*                                                                  *WWE001*
001200*          GERACAO DE RELATORIO CON AS REQUISICOES                 *WWE001*
001300*          DE UMA SERIE DE ORDENS, CLASSIFICADAS                   *WWE001*
001400*          PELO CODIGO-MATERIAL, DATA-NECESSIDADE E                *WWE001*
001500*          CODIGO-ORDEM PARA O LEVANTAMENTO DAS NE-                *WWE001*
001600*          CESSIDADES LIQUIDAS DE SUPRIMENTOS.                     *WWE001*
001700*                                                                  *WWE001*
001800                                                                   *WWE001*
```

Figura 3.5 - Programa fonte em COBOL para a geração do Relatório das Necessidades de Matéria-Prima - e.p.a. -



```
2                        9:32:59     OCT 16,1984

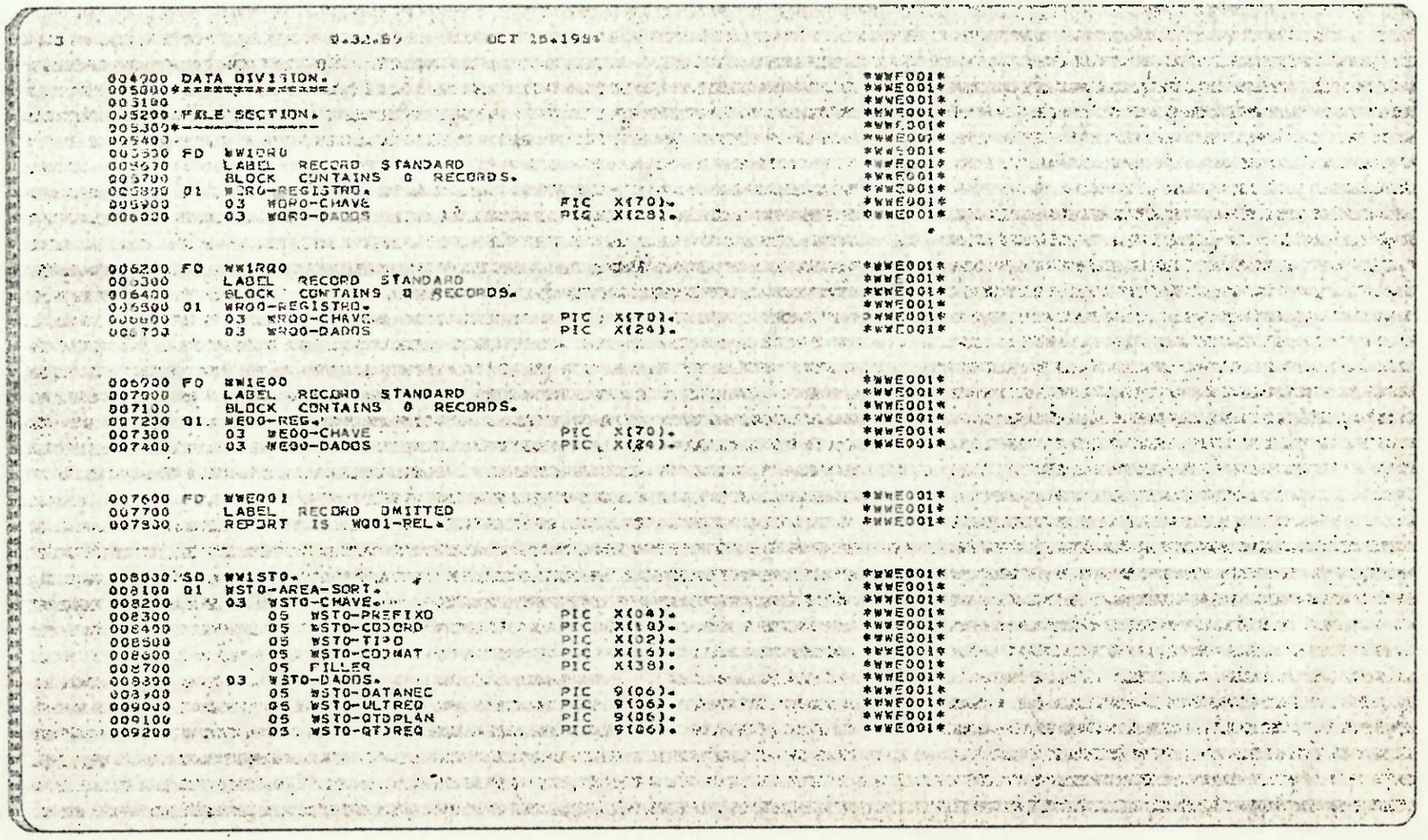
002000 ENVIRONMENT DIVISION.                                          *WWE001*
002100*====================*                                          *WWE001*
002200                                                                *WWE001*
002300 CONFIGURATION SECTION.                                         *WWE001*
002400 SOURCE-COMPUTER.  IBM-4341.                                     *WWE001*
002500 OBJECT-COMPUTER.  IBM-4341.                                     *WWE001*
002600 INPUT-OUTPUT SECTION.                                          *WWE001*
002700                                                                *WWE001*
002800 FILE-CONTROL.                                                  *WWE001*
002900*------------*                                                  *WWE001*
003000                                                                *WWE001*
003100     SELECT  WW1ORD  ASSIGN  TO  VDUD18                         *WWE001*
003200             ORGANIZATION  IS  INDEXED                          *WWE001*
003300             ACCESS  MODE  IS  DYNAMIC                          *WWE001*
003400             RECORD  KEY   IS  WORD-CHAVE                       *WWE001*
003500             FILE  STATUS  IS  VSAM-STATUS.                     *WWE001*


003700     SELECT  WW1RQD  ASSIGN  TO  VDUD22                         *WWE001*
003800             ORGANIZATION  IS  INDEXED                          *WWE001*
003900             ACCESS  MODE  IS  DYNAMIC                          *WWE001*
004000             RECORD  KEY   IS  WRQD-CHAVE                       *WWE001*
004100             FILE  STATUS  IS  VSAM-STATUS.                     *WWE001*
004200                                                                *WWE001*
004300     SELECT  WWEQ01  ASSIGN  TO  UT-S-WWEQ01.                   *WWE001*


004500     SELECT  WW1SFD  ASSIGN  TO  UT-S-SORTWKD1.                 *WWE001*


004700     SELECT  WW1E00  ASSIGN  TO  UT-S-WW  00.                   *WWE001*
```

Continuação da figura 3.5



```
3                    9.32.59     OCT 15,1994

004000 DATA DIVISION.                                          *WWE001*
005000*=============                                           *WWE001*
005100                                                         *WWE001*
005200 FILE SECTION.                                           *WWE001*
005300*------------                                            *WWE001*
005400                                                         *WWE001*
005500 FD  WWIRQ0                                              *WWE001*
005600     LABEL  RECORD  STANDARD                             *WWE001*
005700     BLOCK  CONTAINS  0  RECORDS.                        *WWE001*
005800 01  WIRQ-REGISTRO.                                      *WWE001*
005900     03  WIRQ-CHAVE              PIC  X(70).            *WWE001*
006000     03  WIRQ-DADOS              PIC  X(28).            *WWE001*

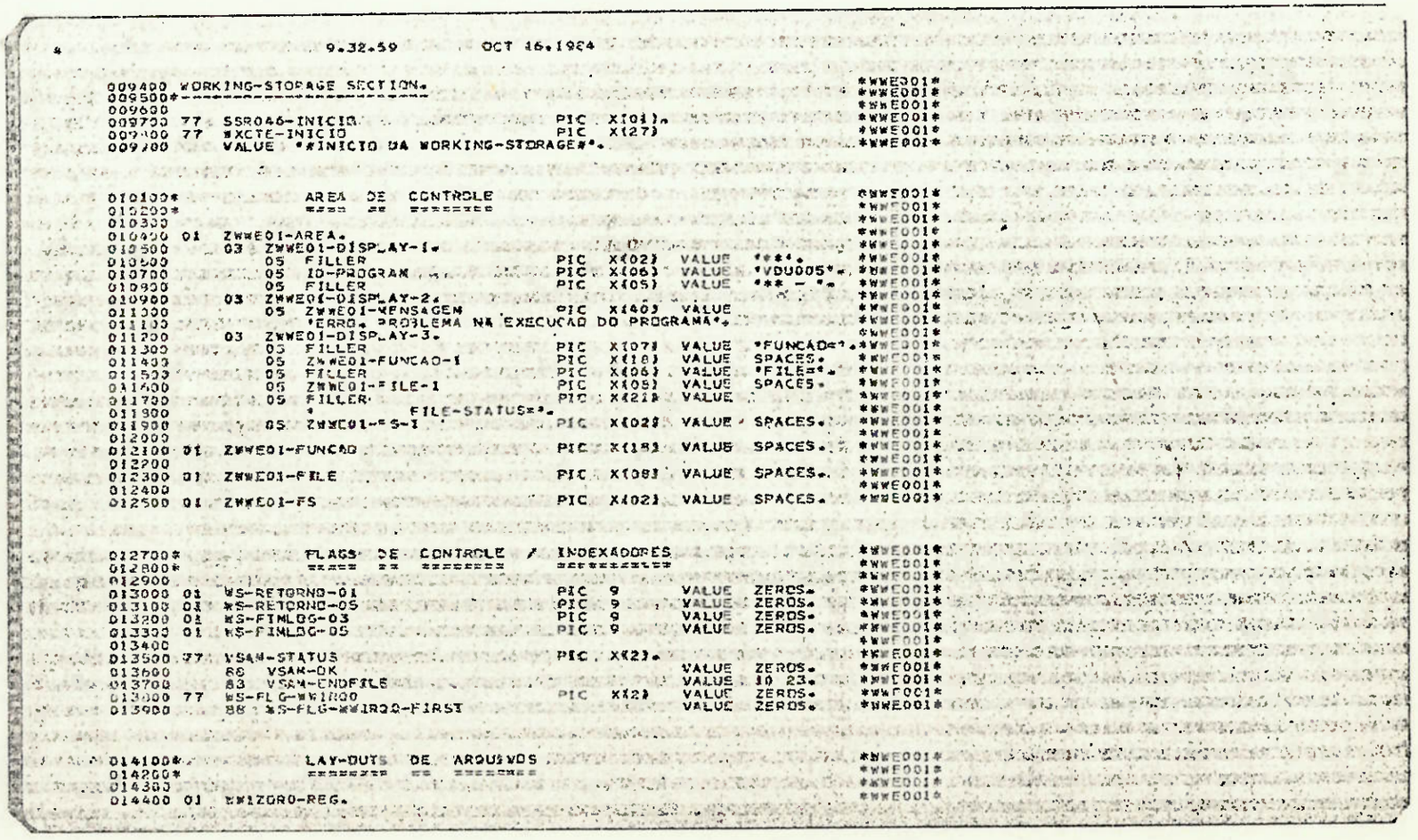

006200 FD  WW1RQ0                                              *WWE001*
006300     LABEL  RECORD  STANDARD                             *WWE001*
006400     BLOCK  CONTAINS  0  RECORDS.                        *WWE001*
006500 01  WRQ0-REGISTRO.                                      *WWE001*
006600     03  WRQ0-CHAVE              PIC  X(70).            *WWE001*
006700     03  WRQ0-DADOS              PIC  X(24).            *WWE001*


006900 FD  WWIE00                                             *WWE001*
007000     LABEL  RECORD  STANDARD                             *WWE001*
007100     BLOCK  CONTAINS  0  RECORDS.                        *WWE001*
007200 01  WE00-REG.                                           *WWE001*
007300     03  WE00-CHAVE              PIC  X(70).            *WWE001*
007400     03  WE00-DADOS              PIC  X(24).            *WWE001*


007600 FD  WWEQ01                                             *WWE001*
007700     LABEL  RECORD  OMITTED                              *WWE001*
007800     REPORT  IS  WQ01-REL.                               *WWE001*


008000 SD  WW1ST0.                                            *WWE001*
008100 01  WST0-AREA-SORT.                                     *WWE001*
008200     03  WST0-CHAVE.                                     *WWE001*
008300         05  WST0-PREFIXO        PIC  X(04).            *WWE001*
008400         05  WST0-CODORD         PIC  X(10).            *WWE001*
008500         05  WST0-TIPO           PIC  X(02).            *WWE001*
008600         05  WST0-CODMAT         PIC  X(16).            *WWE001*
008700         05  FILLER              PIC  X(38).            *WWE001*
008800     03  WST0-DADOS.                                     *WWE001*
008900         05  WST0-DATANEC        PIC  9(06).            *WWE001*
009000         05  WST0-ULTREQ         PIC  9(06).            *WWE001*
009100         05  WST0-QTDPLAN        PIC  9(06).            *WWE001*
009200     03  WST0-QTDREQ             PIC  9(06).            *WWE001*
```

Continuação da figura 3.5



```
4                          9.32.59        OCT 16,1984

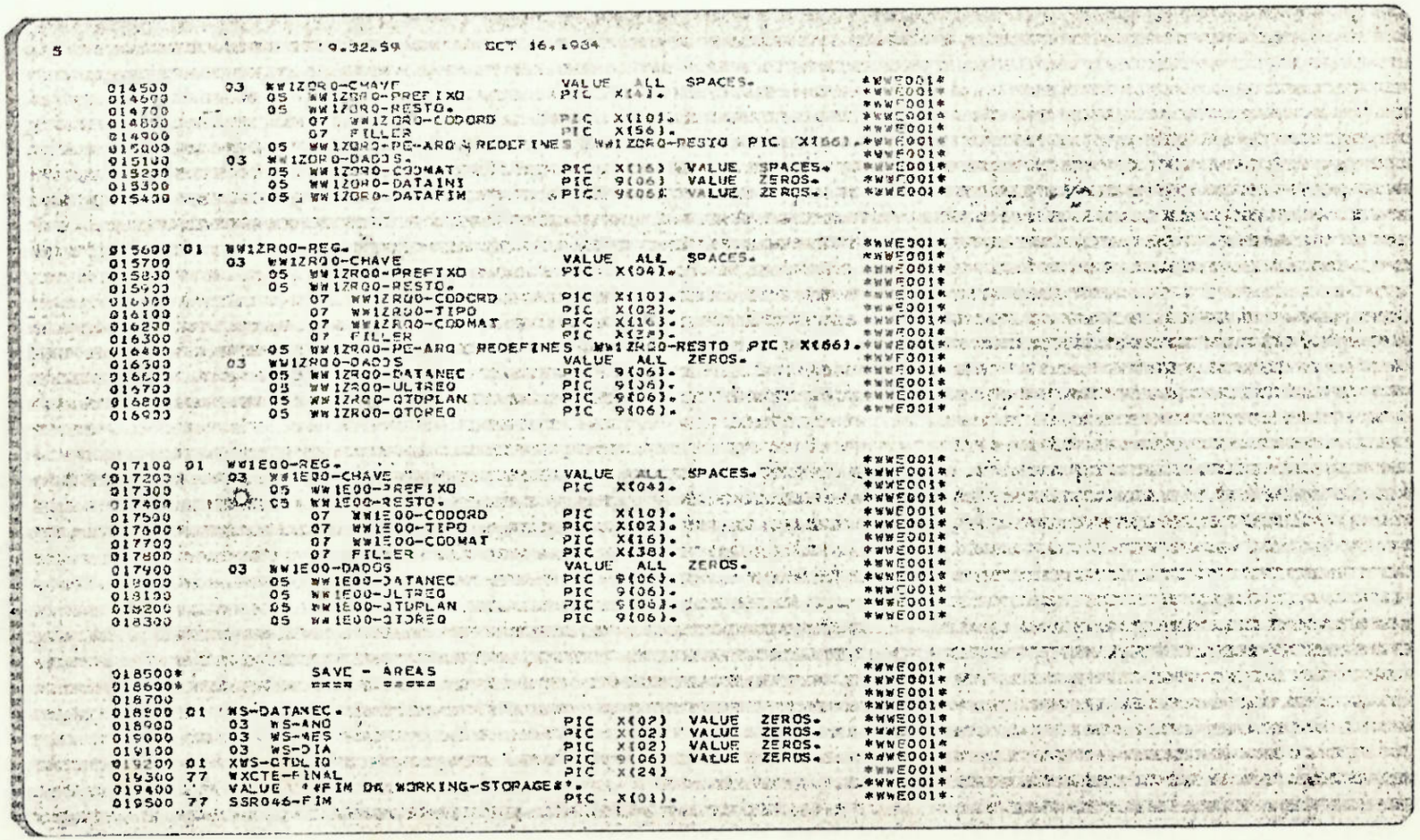
009400 WORKING-STORAGE SECTION.                                          *WWE001*
009500*----------------------                                           *WWE001*
009600                                                                   *WWE001*
009700 77  SSR046-INICIO                  PIC  X(01).                   *WWE001*
009800 77  WXCTE-INICIO                   PIC  X(27)                    *WWE001*
009900     VALUE  '*INICIO DA WORKING-STORAGE*'.                        *WWE001*

010000*                                                                 *WWE001*
010100*            AREA  DE  CONTROLE                                   *WWE001*
010200*            ====  ==  ========                                   *WWE001*
010300                                                                   *WWE001*
010400 01  ZWWE01-AREA.                                                 *WWE001*
010500     03  ZWWE01-DISPLAY-1.                                        *WWE001*
010600         05  FILLER                 PIC  X(02)  VALUE  '**'.      *WWE001*
010700         05  ID-PROGRAM             PIC  X(06)  VALUE  'VDU005'.  *WWE001*
010800         05  FILLER                 PIC  X(05)  VALUE  ' - '.     *WWE001*
010900     03  ZWWE01-DISPLAY-2.                                        *WWE001*
011000         05  ZWWE01-MENSAGEM        PIC  X(40)  VALUE             *WWE001*
011100             'ERRO. PROBLEMA NA EXECUCAO DO PROGRAMA'.            *WWE001*
011200     03  ZWWE01-DISPLAY-3.                                        *WWE001*
011300         05  FILLER                 PIC  X(07)  VALUE  'FUNCAO='. *WWE001*
011400         05  ZWWE01-FUNCAO-1        PIC  X(18)  VALUE  SPACES.    *WWE001*
011500         05  FILLER                 PIC  X(06)  VALUE  'FILE='.   *WWE001*
011600         05  ZWWE01-FILE-1          PIC  X(08)  VALUE  SPACES.    *WWE001*
011700         05  FILLER                 PIC  X(21)  VALUE             *WWE001*
011800             '  FILE-STATUS='.                                    *WWE001*
011900         05  ZWWE01-FS-1            PIC  X(02)  VALUE  SPACES.    *WWE001*
012000                                                                   *WWE001*
012100 01  ZWWE01-FUNCAO                  PIC  X(18)  VALUE  SPACES.    *WWE001*
012200                                                                   *WWE001*
012300 01  ZWWE01-FILE                    PIC  X(08)  VALUE  SPACES.    *WWE001*
012400                                                                   *WWE001*
012500 01  ZWWE01-FS                      PIC  X(02)  VALUE  SPACES.    *WWE001*

012700*            FLAGS  DE  CONTROLE  /  INDEXADORES                  *WWE001*
012800*            =====  ==  ========     ==========                   *WWE001*
012900                                                                   *WWE001*
013000 01  WS-RETORNO-01                  PIC  9      VALUE  ZEROS.     *WWE001*
013100 01  WS-RETORNO-05                  PIC  9      VALUE  ZEROS.     *WWE001*
013200 01  WS-FIMLOG-03                   PIC  9      VALUE  ZEROS.     *WWE001*
013300 01  WS-FIMLOG-05                   PIC  9      VALUE  ZEROS.     *WWE001*
013400                                                                   *WWE001*
013500 77  VSAM-STATUS                    PIC  X(2).                    *WWE001*
013600     88  VSAM-OK                                VALUE  ZEROS.     *WWE001*
013700     88  VSAM-ENDFILE                           VALUE  10 23.     *WWE001*
013800 77  WS-FLG-WWIR00                  PIC  X(1)   VALUE  ZEROS.     *WWE001*
013900     88  WS-FLG-WWIR00-FIRST                    VALUE  ZEROS.     *WWE001*

014100*            LAY-OUTS  DE  ARQUIVOS                               *WWE001*
014200*            ========  ==  =======                                *WWE001*
014300                                                                   *WWE001*
014400 01  WWIZERO-REG.                                                 *WWE001*
```

Continuação da figura 3.5



```
5                          9.32.59      OCT 16,1984

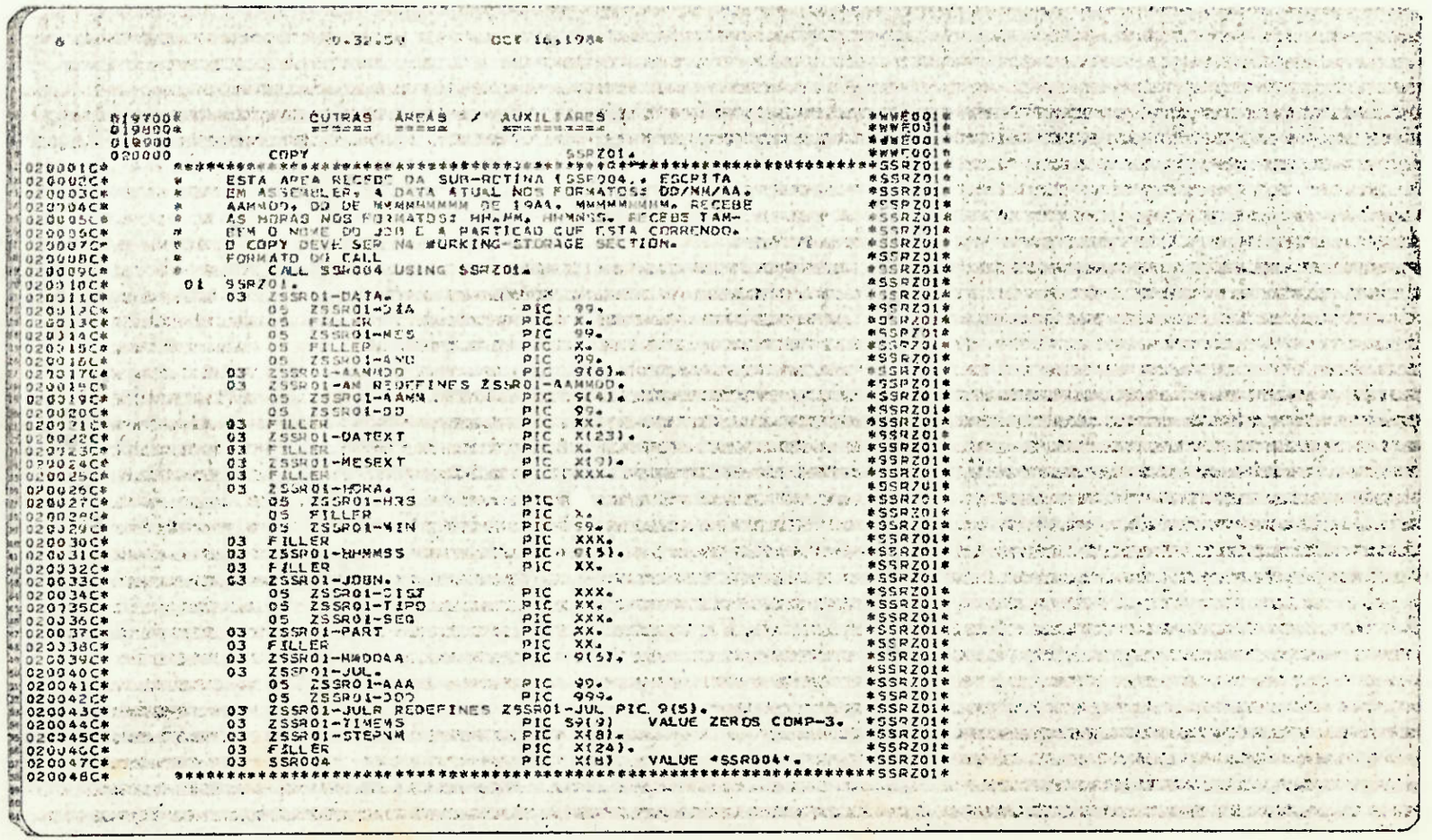
014500      03  WW1ZQR0-CHAVE                  VALUE  ALL  SPACES.          *WWE001*
014600          05  WW1ZQR0-PREFIXO             PIC  X(4).                   *WWE001*
014700          05  WW1ZQR0-RESTO.                                           *WWE001*
014800              07  WW1ZQR0-CODORD          PIC  X(10).                  *WWE001*
014900              07  FILLER                  PIC  X(56).                  *WWE001*
015000          05  WW1ZQR0-PC-ARQ  REDEFINES  WW1ZQR0-RESTO  PIC  X(66).   *WWE001*
015100      03  WW1ZQR0-DADOS.                                               *WWE001*
015200          05  WW1ZQR0-CODMAT              PIC  X(16)  VALUE  SPACES.   *WWE001*
015300          05  WW1ZQR0-DATAINI             PIC  9(06)  VALUE  ZEROS.    *WWE001*
015400          05  WW1ZQR0-DATAFIM             PIC  9(06)  VALUE  ZEROS.    *WWE001*

015600 01  WW1ZRQ0-REG.                                                     *WWE001*
015700      03  WW1ZRQ0-CHAVE                  VALUE  ALL  SPACES.          *WWE001*
015800          05  WW1ZRQ0-PREFIXO             PIC  X(04).                  *WWE001*
015900          05  WW1ZRQ0-RESTO.                                           *WWE001*
016000              07  WW1ZRQ0-CODORD          PIC  X(10).                  *WWE001*
016100              07  WW1ZRQ0-TIPO            PIC  X(02).                  *WWE001*
016200              07  WW1ZRQ0-CODMAT          PIC  X(16).                  *WWE001*
016300              07  FILLER                  PIC  X(38).                  *WWE001*
016400          05  WW1ZRQ0-PC-ARQ  REDEFINES  WW1ZRQ0-RESTO  PIC  X(66).   *WWE001*
016500      03  WW1ZRQ0-DADOS                  VALUE  ALL  ZEROS.           *WWE001*
016600          05  WW1ZRQ0-DATANEC             PIC  9(06).                  *WWE001*
016700          05  WW1ZRQ0-ULTREQ              PIC  9(06).                  *WWE001*
016800          05  WW1ZRQ0-QTDPLAN             PIC  9(06).                  *WWE001*
016900          05  WW1ZRQ0-QTDREQ              PIC  9(06).                  *WWE001*

017100 01  WW1E00-REG.                                                      *WWE001*
017200      03  WW1E00-CHAVE                   VALUE  ALL  SPACES.          *WWE001*
017300          05  WW1E00-PREFIXO              PIC  X(04).                  *WWE001*
017400          05  WW1E00-RESTO.                                            *WWE001*
017500              07  WW1E00-CODORD           PIC  X(10).                  *WWE001*
017600              07  WW1E00-TIPO             PIC  X(02).                  *WWE001*
017700              07  WW1E00-CODMAT           PIC  X(16).                  *WWE001*
017800              07  FILLER                  PIC  X(38).                  *WWE001*
017900      03  WW1E00-DADOS                   VALUE  ALL  ZEROS.           *WWE001*
018000          05  WW1E00-DATANEC              PIC  9(06).                  *WWE001*
018100          05  WW1E00-ULTREQ               PIC  9(06).                  *WWE001*
018200          05  WW1E00-QTDPLAN              PIC  9(06).                  *WWE001*
018300          05  WW1E00-QTDREQ               PIC  9(06).                  *WWE001*

018500*             SAVE - AREAS                                             *WWE001*
018600*             ====   =====                                             *WWE001*
018700                                                                       *WWE001*
018800 01  WS-DATANEC.                                                      *WWE001*
018900      03  WS-ANO                         PIC  X(02)  VALUE  ZEROS.    *WWE001*
019000      03  WS-MES                         PIC  X(02)  VALUE  ZEROS.    *WWE001*
019100      03  WS-DIA                         PIC  X(02)  VALUE  ZEROS.    *WWE001*
019200 01  XWS-QTDLIQ                         PIC  9(06)  VALUE  ZEROS.     *WWE001*
019300 77  WXCTE-FINAL                        PIC  X(24)                    *WWE001*
019400     VALUE  '*FIM DA WORKING-STORAGE*'.                                *WWE001*
019500 77  SSR046-FIM                         PIC  X(01).                   *WWE001*
```

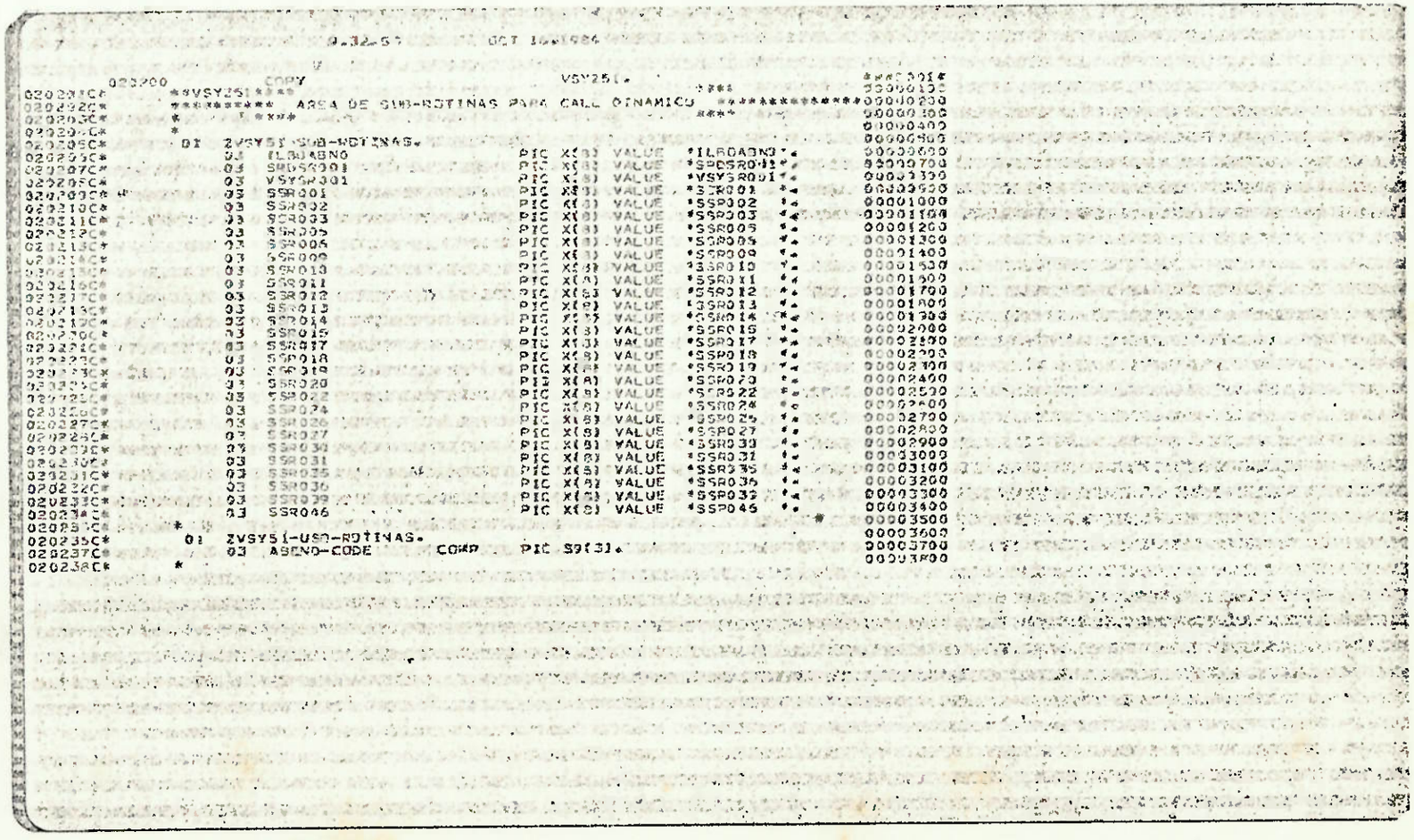
Continuação da figura 3.5



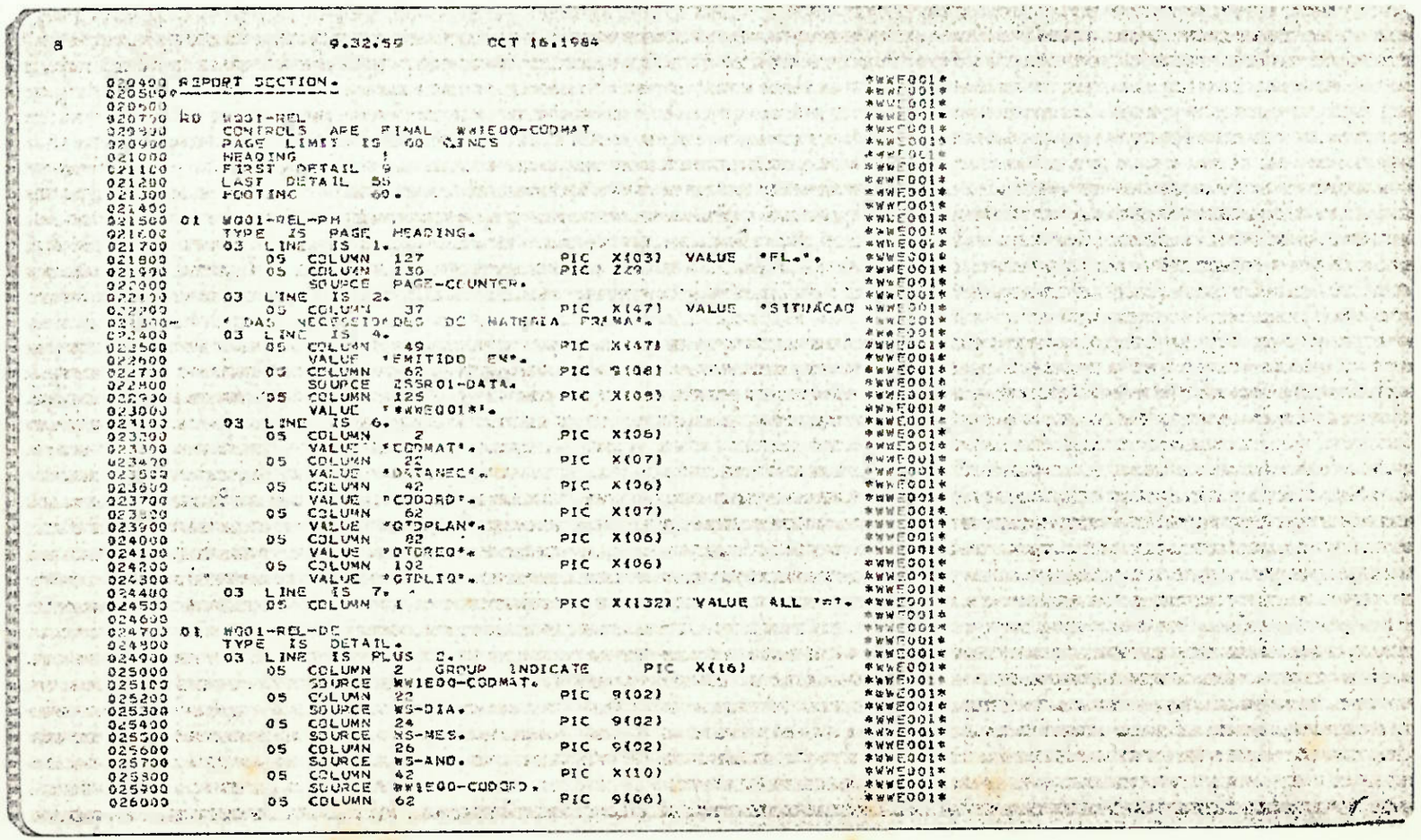
```
6                        9:32:59      OCT 16,1984

      019700*                                                                  *WWFE01*
      019800*                                                                  *WWFE01*
      019900                                                                   *WWFE01*
      020000          COPY                    $SRZ01.                          *WWFE01*
020001C*      ************************************************************      *SSRZ01*
020002C*      *    ESTA AREA RECEBE DA SUB-ROTINA (SSF004,) ESCRITA            *SSRZ01*
020003C*      *    EM ASSEMBLER, A DATA ATUAL NOS FORMATOS: DD/MM/AA,          *SSRZ01*
020004C*      *    AAMMDD, DD DE MMMMMMMMM DE 19AA, MMMMMMMMM, RECEBE         *SSRZ01*
020005C*      *    AS HORAS NOS FORMATOS: HH.MM, HHMMSS. RECEBE TAM-           *SSRZ01*
020006C*      *    BEM O NOME DO JOB E A PARTICAO QUE ESTA CORRENDO.           *SSRZ01*
020007C*      *    O COPY DEVE SER NA WORKING-STORAGE SECTION.                 *SSRZ01*
020008C*      *    FORMATO DO CALL                                             *SSRZ01*
020009C*      *        CALL SSF004 USING $SRZ01.                               *SSRZ01*
020010C*       01  $SRZ01.                                                     *SSRZ01*
020011C*           03  ZSSR01-DATA.                                           *SSRZ01*
020012C*               05  ZSSR01-DIA          PIC  99.                       *SSRZ01*
020013C*               05  FILLER              PIC  X.                        *SSRZ01*
020014C*               05  ZSSR01-MES          PIC  99.                       *SSRZ01*
020015C*               05  FILLER              PIC  X.                        *SSRZ01*
020016C*               05  ZSSR01-ANO          PIC  99.                       *SSRZ01*
020017C*           03  ZSSR01-AAMMDD           PIC  9(6).                     *SSRZ01*
020018C*           03  ZSSR01-AM REDEFINES ZSSR01-AAMMDD.                     *SSRZ01*
020019C*               05  ZSSR01-AAMM         PIC  9(4).                     *SSRZ01*
020020C*               05  ZSSR01-DD           PIC  99.                       *SSRZ01*
020021C*           03  FILLER                  PIC  XX.                       *SSRZ01*
020022C*           03  ZSSR01-DATEXT           PIC  X(23).                    *SSRZ01*
020023C*           03  FILLER                  PIC  X.                        *SSRZ01*
020024C*           03  ZSSR01-MESEXT           PIC  X(9).                     *SSRZ01*
020025C*           03  FILLER                  PIC  XXX.                      *SSRZ01*
020026C*           03  ZSSR01-HORA.                                           *SSRZ01*
020027C*               05  ZSSR01-HRS          PIC  99.                       *SSRZ01*
020028C*               05  FILLER              PIC  X.                        *SSRZ01*
020029C*               05  ZSSR01-MIN          PIC  99.                       *SSRZ01*
020030C*           03  FILLER                  PIC  XXX.                      *SSRZ01*
020031C*           03  ZSSR01-HHMMSS           PIC  9(5).                     *SSRZ01*
020032C*           03  FILLER                  PIC  XX.                       *SSRZ01*
020033C*           03  ZSSR01-JOBN.                                           *SSRZ01
020034C*               05  ZSSR01-DIGT         PIC  XXX.                      *SSRZ01*
020035C*               05  ZSSR01-TIPO         PIC  XX.                       *SSRZ01*
020036C*               05  ZSSR01-SEQ          PIC  XXX.                      *SSRZ01*
020037C*           03  ZSSR01-PART             PIC  XX.                       *SSRZ01*
020038C*           03  FILLER                  PIC  XX.                       *SSRZ01*
020039C*           03  ZSSR01-MMDDAA           PIC  9(5).                     *SSRZ01*
020040C*           03  ZSSR01-JUL.                                            *SSRZ01*
020041C*               05  ZSSR01-AAA          PIC  99.                       *SSRZ01*
020042C*               05  ZSSR01-DDD          PIC  999.                      *SSRZ01*
020043C*           03  ZSSR01-JULR REDEFINES ZSSR01-JUL PIC 9(5).             *SSRZ01*
020044C*           03  ZSSR01-TIMEMS           PIC  S9(9)   VALUE ZEROS COMP-3.*SSRZ01*
020045C*           03  ZSSR01-STEPNM           PIC  X(8).                     *SSRZ01*
020046C*           03  FILLER                  PIC  X(24).                    *SSRZ01*
020047C*           03  SSR004                  PIC  X(8)   VALUE *SSR004*.    *SSRZ01*
020048C*      ************************************************************      *SSRZ01*
```

Continuação da figura 3.5



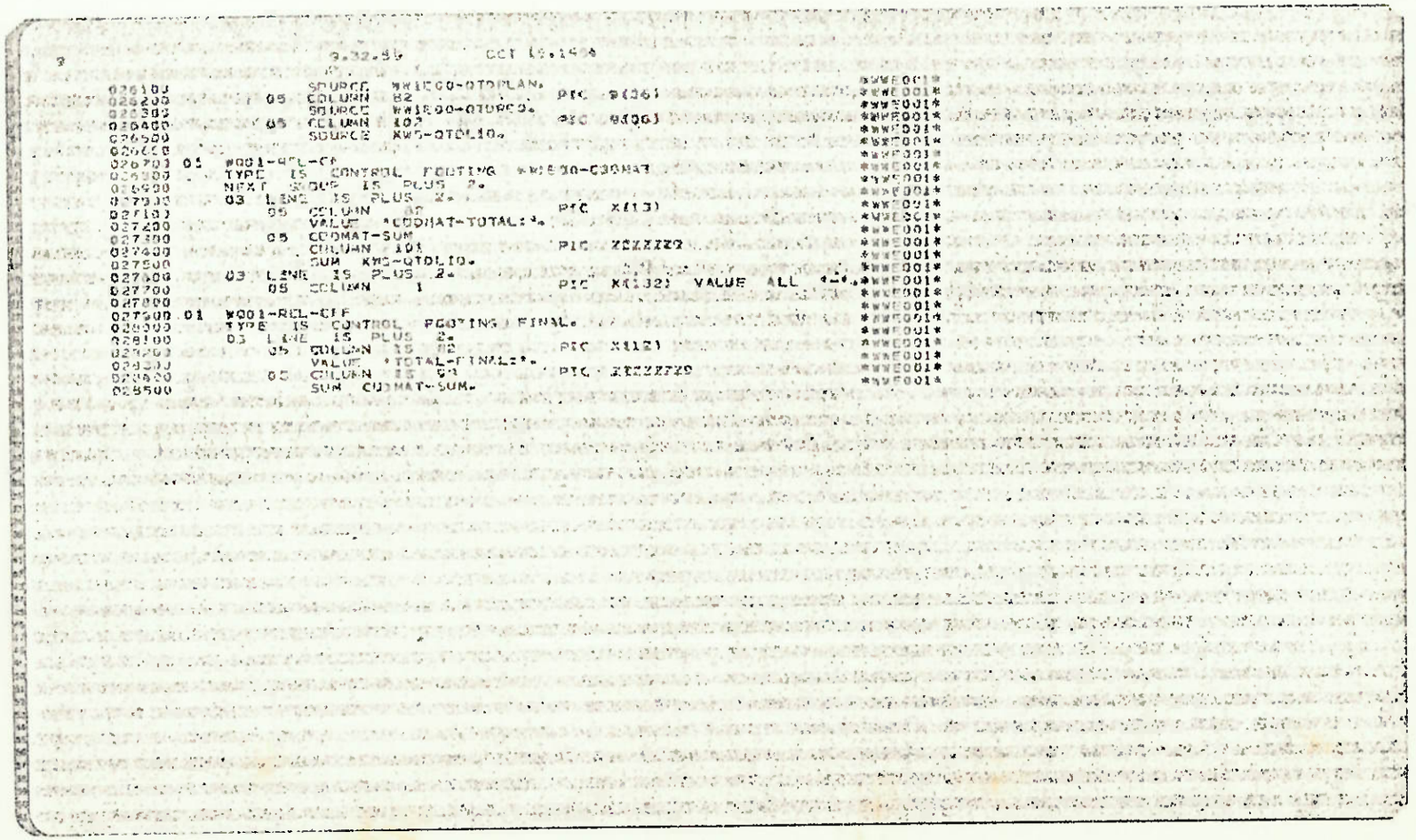
```
                                                
```



```
8                        9.32.59        OCT 16,1984

020400 REPORT SECTION.                                                  *WWE001*
020500*                                                                 *WWE001*
020600                                                                  *WWE001*
020700 RD  WQ01-REL                                                     *WWE001*
020800     CONTROLS  ARE  FINAL  WW1E00-CODMAT                          *WWE001*
020900     PAGE  LIMIT  IS  60  LINES                                   *WWE001*
021000     HEADING      1                                               *WWE001*
021100     FIRST  DETAIL  9                                            *WWE001*
021200     LAST  DETAIL  55                                            *WWE001*
021300     FOOTING      60.                                            *WWE001*
021400                                                                  *WWE001*
021500 01  WQ01-REL-PH                                                 *WWE001*
021600     TYPE  IS  PAGE  HEADING.                                    *WWE001*
021700     03  LINE  IS  1.                                            *WWE001*
021800         05  COLUMN  127              PIC  X(03)  VALUE  "FL.".  *WWE001*
021900         05  COLUMN  130              PIC  ZZ9                   *WWE001*
022000             SOURCE  PAGE-COUNTER.                               *WWE001*
022100     03  LINE  IS  2.                                            *WWE001*
022200         05  COLUMN  37               PIC  X(47)  VALUE  "SITUACAO *WWE001*
022300-    " DAS  NECESSIDADES  DE  MATERIA  PRIMA".                    *WWE001*
022400     03  LINE  IS  4.                                            *WWE001*
022500         05  COLUMN  49               PIC  X(47)                 *WWE001*
022600             VALUE  "EMITIDO  EM".                               *WWE001*
022700         05  COLUMN  62               PIC  9(08)                 *WWE001*
022800             SOURCE  ZSSR01-DATA.                                *WWE001*
022900         05  COLUMN  125              PIC  X(09)                 *WWE001*
023000             VALUE  "*WWE001*".                                  *WWE001*
023100     03  LINE  IS  6.                                            *WWE001*
023200         05  COLUMN  2                PIC  X(06)                 *WWE001*
023300             VALUE  "CODMAT".                                    *WWE001*
023400         05  COLUMN  22               PIC  X(07)                 *WWE001*
023500             VALUE  "DATANEC".                                   *WWE001*
023600         05  COLUMN  42               PIC  X(06)                 *WWE001*
023700             VALUE  "CODORD".                                    *WWE001*
023800         05  COLUMN  62               PIC  X(07)                 *WWE001*
023900             VALUE  "QTDPLAN".                                   *WWE001*
024000         05  COLUMN  82               PIC  X(06)                 *WWE001*
024100             VALUE  "QTDREQ".                                    *WWE001*
024200         05  COLUMN  102              PIC  X(06)                 *WWE001*
024300             VALUE  "QTDLIQ".                                    *WWE001*
024400     03  LINE  IS  7.                                            *WWE001*
024500         05  COLUMN  1                PIC  X(132)  VALUE  ALL "=". *WWE001*
024600                                                                  *WWE001*
024700 01  WQ01-REL-DE                                                 *WWE001*
024800     TYPE  IS  DETAIL.                                           *WWE001*
024900     03  LINE  IS  PLUS  2.                                      *WWE001*
025000         05  COLUMN  2   GROUP  INDICATE      PIC  X(16)         *WWE001*
025100             SOURCE  WW1E00-CODMAT.                              *WWE001*
025200         05  COLUMN  22               PIC  9(02)                 *WWE001*
025300             SOURCE  WS-DIA.                                     *WWE001*
025400         05  COLUMN  24               PIC  9(02)                 *WWE001*
025500             SOURCE  WS-MES.                                     *WWE001*
025600         05  COLUMN  26               PIC  9(02)                 *WWE001*
025700             SOURCE  WS-ANO.                                     *WWE001*
025800         05  COLUMN  42               PIC  X(10)                 *WWE001*
025900             SOURCE  WW1E00-CODORD.                              *WWE001*
026000         05  COLUMN  62               PIC  9(06)                 *WWE001*
```

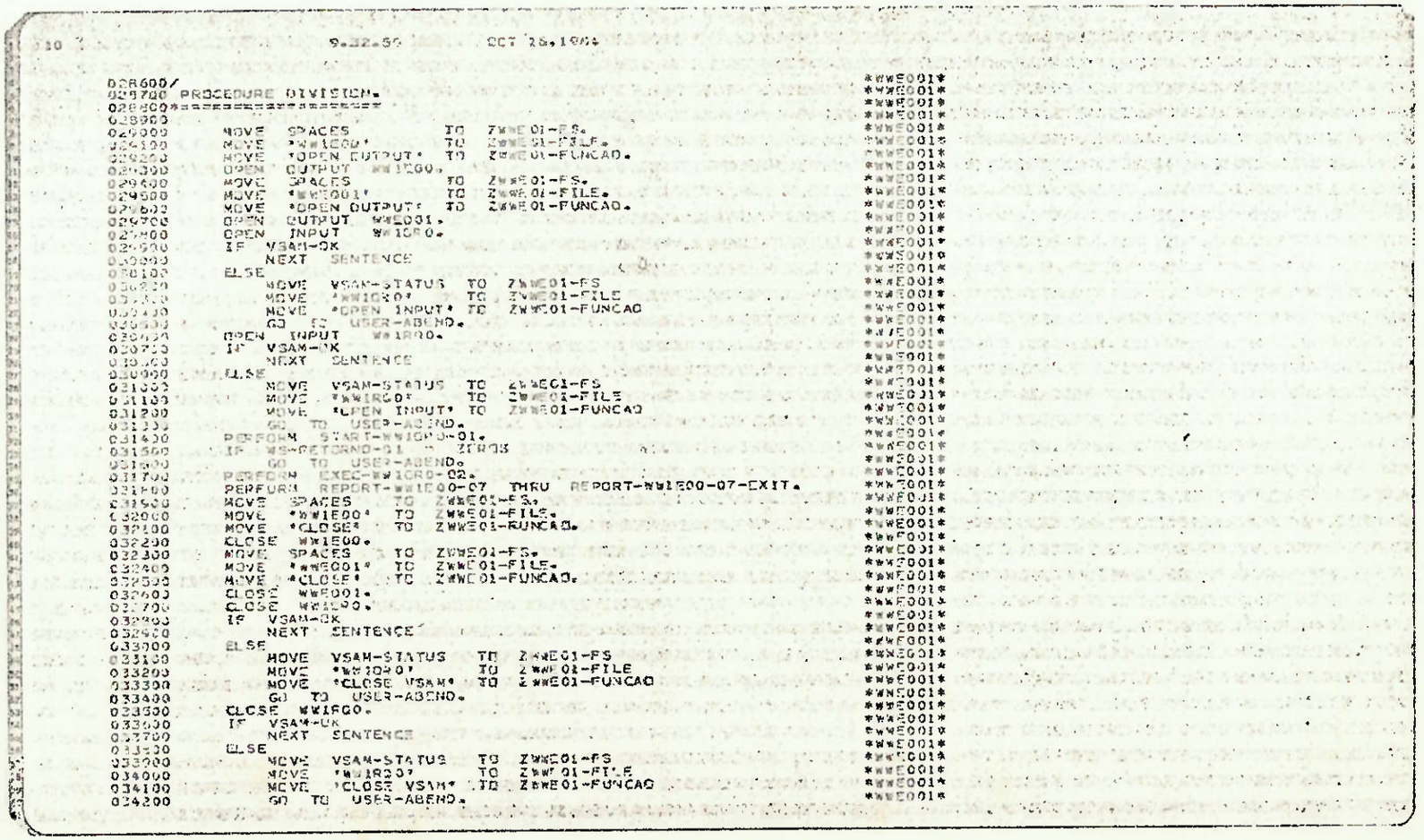
Continuação da figura 3.5



```
                    9.32.56      CCI 10.1.4.4

026100                 SOURCE  WW1E00-QTDPLAN.                  PIC  9(06)      *WWE001*
026200             05  COLUMN  82                                                *WWE001*
026300                 SOURCE  WW1E00-QTDPC9.                   PIC  9(06)      *WWE001*
026400             05  COLUMN  102                                               *WWE001*
026500                 SOURCE  KW5-QTDLIQ.                                        *WWE001*
026600                                                                            *WWE001*
026700 01  W001-REL-CF                                                           *WWE001*
026800     TYPE  IS  CONTROL  FOOTING  WW1E00-CODMAT                              *WWE001*
026900     NEXT  GROUP  IS  PLUS  2.                                              *WWE001*
027000     03  LINE  IS  PLUS  2.                                                *WWE001*
027100         05  COLUMN  42                                  PIC  X(13)        *WWE001*
027200             VALUE  'CODMAT-TOTAL:'.                                        *WWE001*
027300         05  CODMAT-SUM                                  PIC  ZZZZZZ9      *WWE001*
027400             COLUMN  101                                                    *WWE001*
027500             SUM  KW5-QTDLIQ.                                               *WWE001*
027600     03  LINE  IS  PLUS  2.                              PIC  X(132)  VALUE  ALL  '-'.*WWE001*
027700         05  COLUMN  1                                                     *WWE001*
027800                                                                            *WWE001*
027900 01  W001-REL-CF                                                           *WWE001*
028000     TYPE  IS  CONTROL  FOOTING  FINAL.                                     *WWE001*
028100     03  LINE  IS  PLUS  2.                              PIC  X(12)        *WWE001*
028200         05  COLUMN  35  PIC                                               *WWE001*
028300             VALUE  'TOTAL-FINAL:'.                      PIC  ZZZZZZ9      *WWE001*
028400         05  COLUMN  IS  91                                                *WWE001*
028500             SUM  CODMAT-SUM.                                               *WWE001*
```

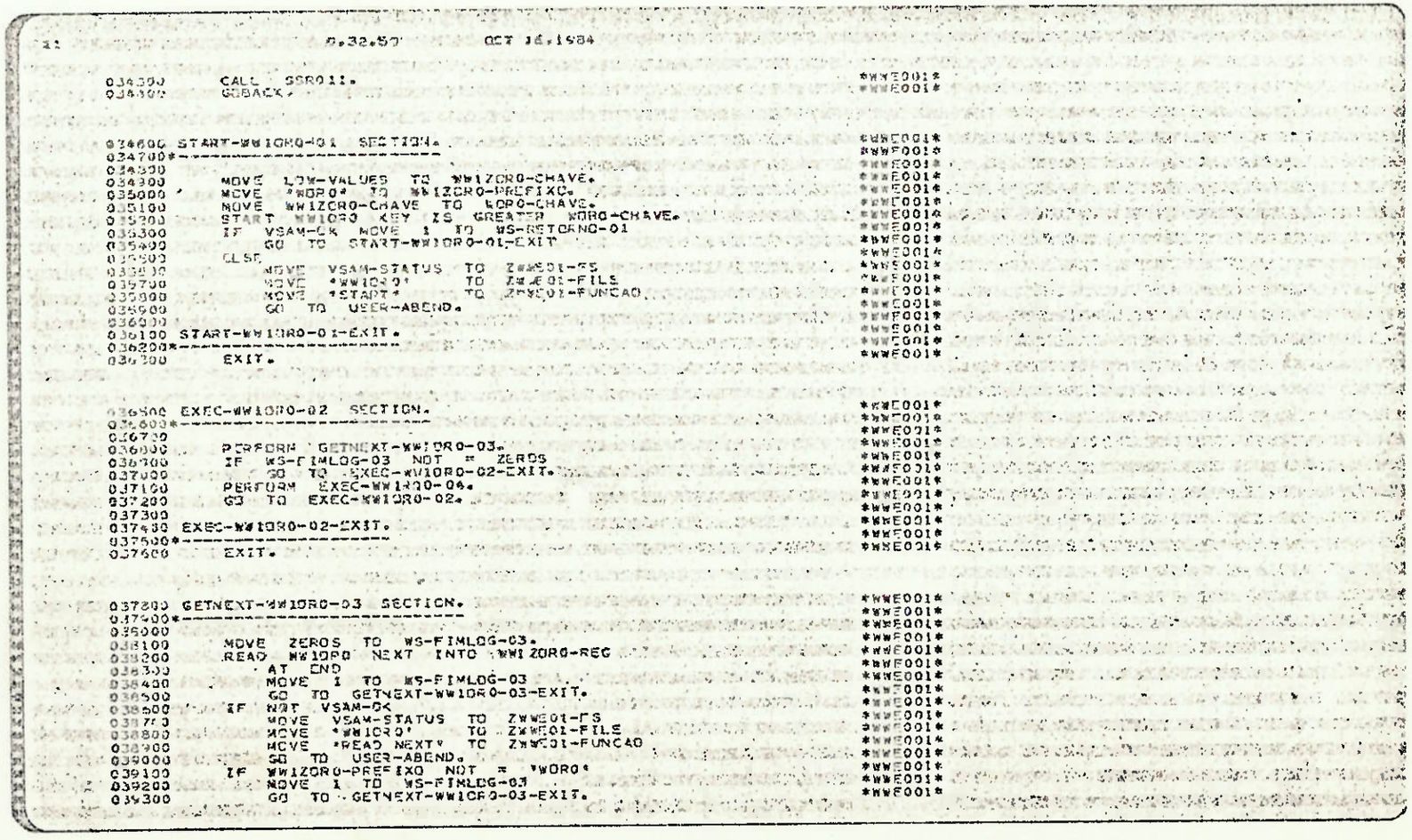
Continuação da figura 3.5



```
010 *                    9.32.50      OCT 28,1994

028600/                                                          *WWE001*
028700 PROCEDURE DIVISION.                                       *WWE001*
028800*=================                                        *WWE001*
028900                                                           *WWE001*
029000     MOVE  SPACES          TO  ZWWE01-FS.                 *WWE001*
029100     MOVE  'WW1E00'        TO  ZWWE01-FILE.               *WWE001*
029200     MOVE  'OPEN OUTPUT'   TO  ZWWE01-FUNCAO.             *WWE001*
029300     OPEN  OUTPUT  WW1E00.                                *WWE001*
029400     MOVE  SPACES          TO  ZWWE01-FS.                 *WWE001*
029500     MOVE  'WWE001'        TO  ZWWE01-FILE.               *WWE001*
029600     MOVE  'OPEN OUTPUT'   TO  ZWWE01-FUNCAO.             *WWE001*
029700     OPEN  OUTPUT  WWE001.                                *WWE001*
029800     OPEN  INPUT   WW1R00.                                *WWE001*
029900     IF  VSAM-OK                                          *WWE001*
030000         NEXT  SENTENCE                                   *WWE001*
030100     ELSE                                                 *WWE001*
030200         MOVE  VSAM-STATUS  TO  ZWWE01-FS                 *WWE001*
030300         MOVE  'WW1R00'     TO  ZWWE01-FILE               *WWE001*
030400         MOVE  'OPEN INPUT' TO  ZWWE01-FUNCAO             *WWE001*
030500         GO  TO  USER-ABEND.                              *WWE001*
030600     OPEN  INPUT   WW1R00.                                *WWE001*
030700     IF  VSAM-OK                                          *WWE001*
030800         NEXT  SENTENCE                                   *WWE001*
030900     ELSE                                                 *WWE001*
031000         MOVE  VSAM-STATUS  TO  ZWWE01-FS                 *WWE001*
031100         MOVE  'WW1R00'     TO  ZWWE01-FILE               *WWE001*
031200         MOVE  'OPEN INPUT' TO  ZWWE01-FUNCAO             *WWE001*
031300         GO  TO  USER-ABEND.                              *WWE001*
031400     PERFORM  START-WW1R00-01.                            *WWE001*
031500     IF  WS-RETORNO-01  =  ZEROS                          *WWE001*
031600         GO  TO  USER-ABEND.                              *WWE001*
031700     PERFORM  EXEC-WW1R00-02.                             *WWE001*
031800     PERFORM  REPORT-WW1E00-07  THRU  REPORT-WW1E00-07-EXIT.  *WWE001*
031900     MOVE  SPACES     TO  ZWWE01-FS.                      *WWE001*
032000     MOVE  'WW1E00'   TO  ZWWE01-FILE.                    *WWE001*
032100     MOVE  'CLOSE'    TO  ZWWE01-FUNCAO.                  *WWE001*
032200     CLOSE  WW1E00.                                       *WWE001*
032300     MOVE  SPACES     TO  ZWWE01-FS.                      *WWE001*
032400     MOVE  'WWE001'   TO  ZWWE01-FILE.                    *WWE001*
032500     MOVE  'CLOSE'    TO  ZWWE01-FUNCAO.                  *WWE001*
032600     CLOSE  WWE001.                                       *WWE001*
032700     CLOSE  WW1R00.                                       *WWE001*
032800     IF  VSAM-OK                                          *WWE001*
032900         NEXT  SENTENCE                                   *WWE001*
033000     ELSE                                                 *WWE001*
033100         MOVE  VSAM-STATUS   TO  ZWWE01-FS                *WWE001*
033200         MOVE  'WW1R00'      TO  ZWWE01-FILE              *WWE001*
033300         MOVE  'CLOSE VSAM'  TO  ZWWE01-FUNCAO            *WWE001*
033400         GO  TO  USER-ABEND.                              *WWE001*
033500     CLOSE  WW1R00.                                       *WWE001*
033600     IF  VSAM-OK                                          *WWE001*
033700         NEXT  SENTENCE                                   *WWE001*
033800     ELSE                                                 *WWE001*
033900         MOVE  VSAM-STATUS   TO  ZWWE01-FS                *WWE001*
034000         MOVE  'WW1R00'      TO  ZWWE01-FILE              *WWE001*
034100         MOVE  'CLOSE VSAM'  TO  ZWWE01-FUNCAO            *WWE001*
034200         GO  TO  USER-ABEND.                              *WWE001*
```

Continuação da figura 3.5



```
11                    0.32.57        OCT 16,1984

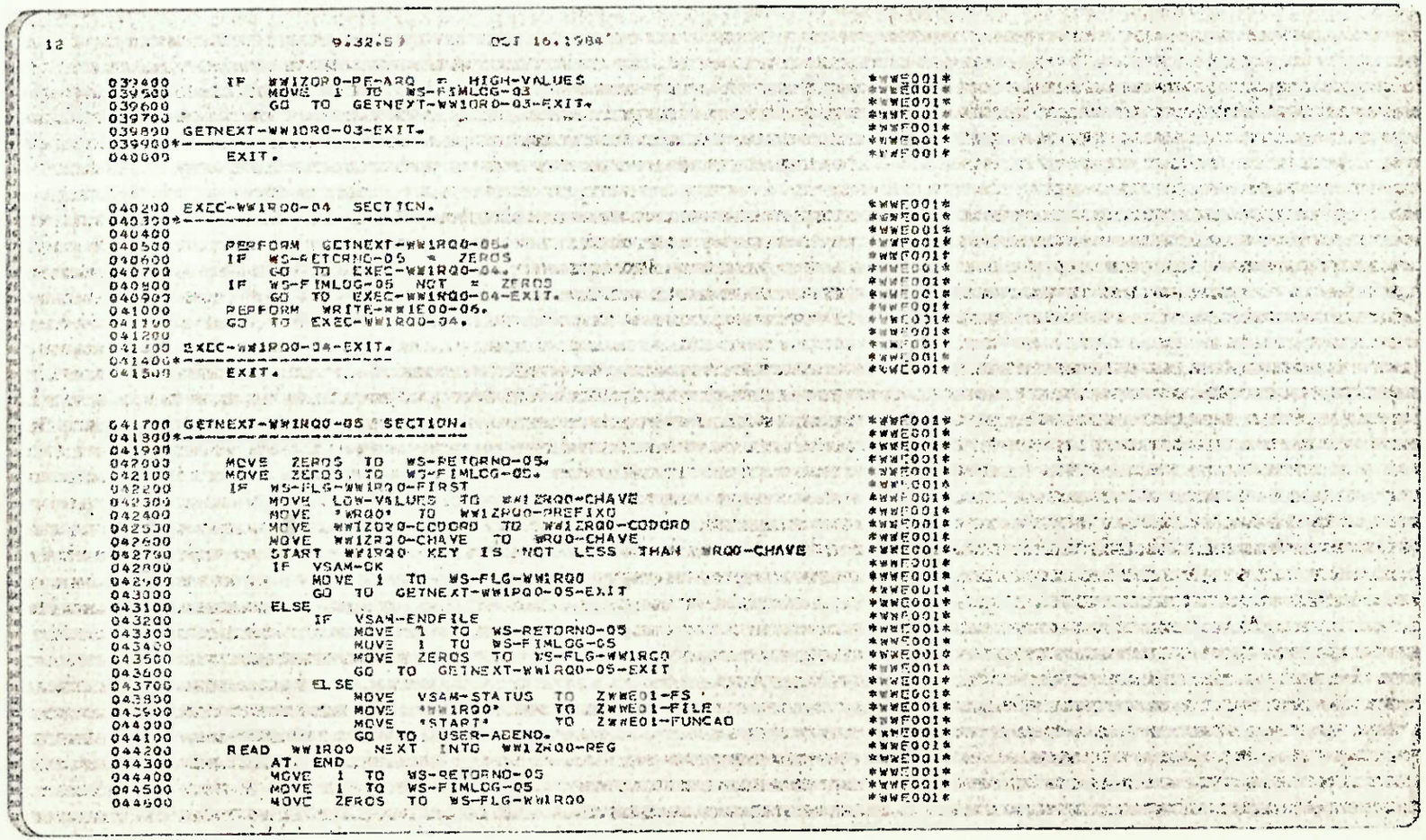
034500      CALL  GSR011.                                          *WWE001*
034600      GOBACK.                                                *WWE001*


034700 START-WW1ORO-01  SECTION.                                   *WWE001*
034800*-------------------------                                   *WWE001*
034900                                                             *WWE001*
035000      MOVE  LOW-VALUES  TO  WW1ZORO-CHAVE.                   *WWE001*
035100      MOVE  'WORO'  TO  WW1ZORO-PREFIXO.                     *WWE001*
035200      MOVE  WW1ZORO-CHAVE  TO  WORO-CHAVE.                   *WWE001*
035300      START  WW1ORO  KEY  IS  GREATER  WORO-CHAVE.           *WWE001*
035400      IF  VSAM-OK  MOVE  1  TO  WS-RETORNO-01                *WWE001*
035500          GO  TO  START-WW1ORO-01-EXIT                       *WWE001*
035600      ELSE                                                   *WWE001*
035700          MOVE  VSAM-STATUS  TO  ZWWE01-FS                   *WWE001*
035800          MOVE  'WW1ORO'     TO  ZWWE01-FILE                 *WWE001*
035900          MOVE  'START'      TO  ZWWE01-FUNCAO               *WWE001*
036000          GO  TO  USER-ABEND.                                *WWE001*
036100                                                             *WWE001*
036200 START-WW1ORO-01-EXIT.                                        *WWE001*
036300*-------------------------                                   *WWE001*
036400      EXIT.                                                  *WWE001*


036500 EXEC-WW1ORO-02  SECTION.                                     *WWE001*
036600*-------------------------                                   *WWE001*
036700                                                             *WWE001*
036800      PERFORM  GETNEXT-WW1ORO-03.                            *WWE001*
036900      IF  WS-FIMLOG-03  NOT  =  ZEROS                        *WWE001*
037000          GO  TO  EXEC-WW1ORO-02-EXIT.                       *WWE001*
037100      PERFORM  EXEC-WW1ORO-04.                               *WWE001*
037200      GO  TO  EXEC-WW1ORO-02.                                *WWE001*
037300                                                             *WWE001*
037400 EXEC-WW1ORO-02-EXIT.                                         *WWE001*
037500*-------------------------                                   *WWE001*
037600      EXIT.                                                  *WWE001*


037700 GETNEXT-WW1ORO-03  SECTION.                                  *WWE001*
037800*-------------------------                                   *WWE001*
037900                                                             *WWE001*
038000      MOVE  ZEROS  TO  WS-FIMLOG-03.                         *WWE001*
038100      READ  WW1ORO  NEXT  INTO  WW1ZORO-REG                  *WWE001*
038200          AT  END                                            *WWE001*
038300          MOVE  1  TO  WS-FIMLOG-03                          *WWE001*
038400          GO  TO  GETNEXT-WW1ORO-03-EXIT.                    *WWE001*
038500      IF  NOT  VSAM-OK                                       *WWE001*
038600          MOVE  VSAM-STATUS  TO  ZWWE01-FS                   *WWE001*
038700          MOVE  'WW1ORO'     TO  ZWWE01-FILE                 *WWE001*
038800          MOVE  'READ NEXT'  TO  ZWWE01-FUNCAO               *WWE001*
038900          GO  TO  USER-ABEND.                                *WWE001*
039000      IF  WW1ZORO-PREFIXO  NOT  =  'WORO'                    *WWE001*
039100          MOVE  1  TO  WS-FIMLOG-03                          *WWE001*
039200          GO  TO  GETNEXT-WW1ORO-03-EXIT.                    *WWE001*
039300
```

Continuação da figura 3.5



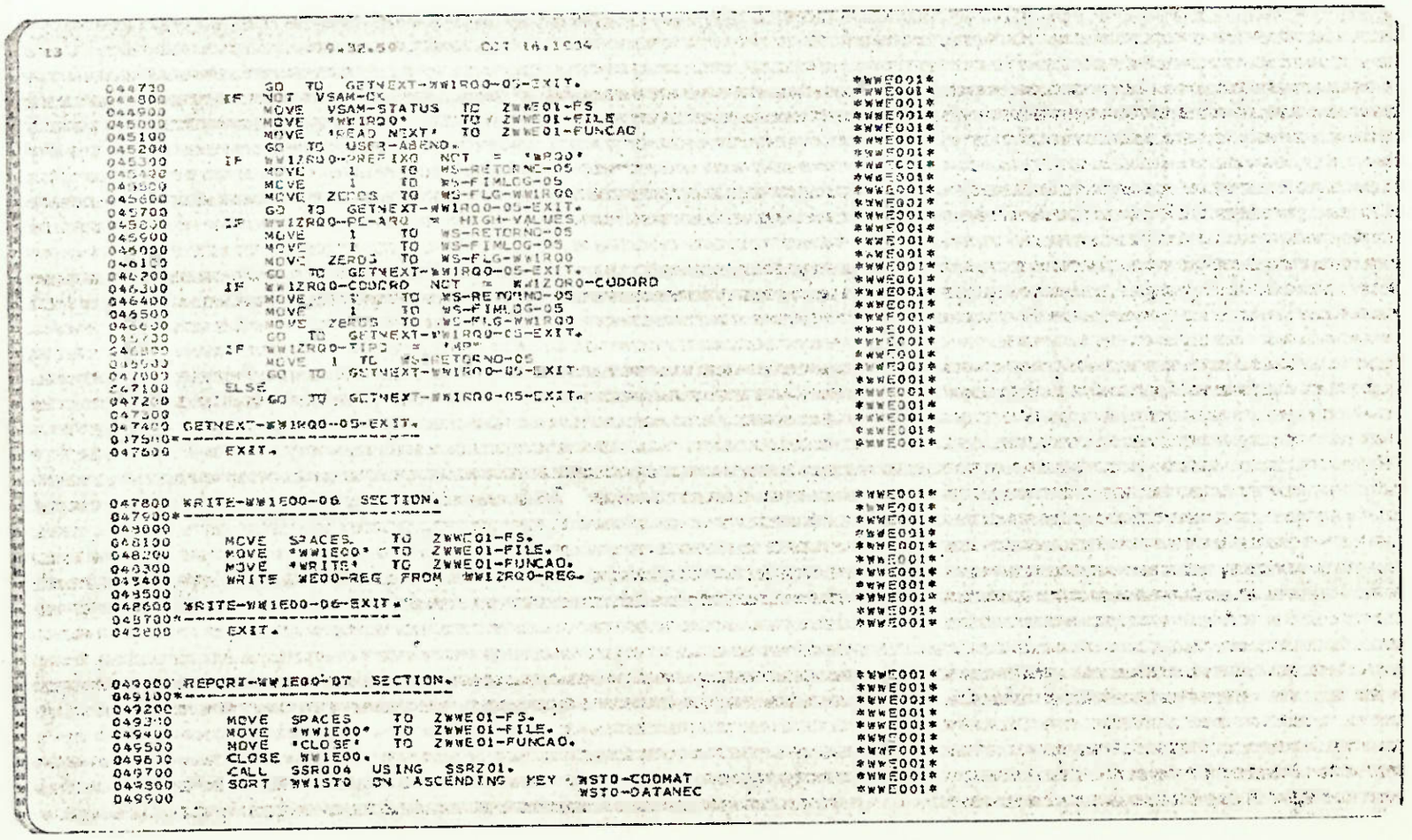
```
12                          9:32:59        OCT 16,1984

039400      IF  WW1ZRQ0-PE-ARQ  =  HIGH-VALUES                    *WWE001*
039500          MOVE  1  TO  WS-FIMLOG-03                         *WWE001*
039600          GO  TO  GETNEXT-WW1QRQ-03-EXIT.                   *WWE001*
039700                                                            *WWE001*
039800 GETNEXT-WW1QRQ-03-EXIT.                                    *WWE001*
039900*-----------------------                                   *WWE001*
040000      EXIT.                                                 *WWE001*


040200 EXEC-WW1RQ0-04  SECTION.                                   *WWE001*
040300*-----------------------                                   *WWE001*
040400                                                            *WWE001*
040500      PERFORM  GETNEXT-WW1RQ0-05.                           *WWE001*
040600      IF  WS-RETORNO-05  =  ZEROS                           *WWE001*
040700          GO  TO  EXEC-WW1RQ0-04.                           *WWE001*
040800      IF  WS-FIMLOG-05  NOT  =  ZEROS                       *WWE001*
040900          GO  TO  EXEC-WW1RQ0-04-EXIT.                      *WWE001*
041000      PERFORM  WRITE-WW1E00-06.                             *WWE001*
041100      GO  TO  EXEC-WW1RQ0-04.                               *WWE001*
041200                                                            *WWE001*
041300 EXEC-WW1RQ0-04-EXIT.                                       *WWE001*
041400*-----------------------                                   *WWE001*
041500      EXIT.                                                 *WWE001*


041700 GETNEXT-WW1RQ0-05  SECTION.                                *WWE001*
041800*-----------------------                                   *WWE001*
041900                                                            *WWE001*
042000      MOVE  ZEROS  TO  WS-RETORNO-05.                       *WWE001*
042100      MOVE  ZEROS  TO  WS-FIMLOG-05.                        *WWE001*
042200      IF  WS-FLG-WW1RQ0-FIRST                               *WWE001*
042300          MOVE  LOW-VALUES  TO  WW1ZRQ0-CHAVE               *WWE001*
042400          MOVE  'WRQ0'  TO  WW1ZRQ0-PREFIXO                 *WWE001*
042500          MOVE  WW1ZRQ0-CODORD  TO  WW1ZRQ0-CODORD          *WWE001*
042600          MOVE  WW1ZRQ0-CHAVE  TO  WRQ0-CHAVE               *WWE001*
042700          START  WW1RQ0  KEY  IS  NOT  LESS  THAN  WRQ0-CHAVE  *WWE001*
042800          IF  VSAM-OK                                       *WWE001*
042900              MOVE  1  TO  WS-FLG-WW1RQ0                    *WWE001*
043000              GO  TO  GETNEXT-WW1RQ0-05-EXIT                *WWE001*
043100          ELSE                                              *WWE001*
043200              IF  VSAM-ENDFILE                              *WWE001*
043300                  MOVE  1  TO  WS-RETORNO-05                *WWE001*
043400                  MOVE  1  TO  WS-FIMLOG-05                 *WWE001*
043500                  MOVE  ZEROS  TO  WS-FLG-WW1RQ0            *WWE001*
043600                  GO  TO  GETNEXT-WW1RQ0-05-EXIT            *WWE001*
043700              ELSE                                          *WWE001*
043800                  MOVE  VSAM-STATUS  TO  ZWWE01-FS          *WWE001*
043900                  MOVE  'WW1RQ0'  TO  ZWWE01-FILE           *WWE001*
044000                  MOVE  'START'  TO  ZWWE01-FUNCAO          *WWE001*
044100                  GO  TO  USER-ABEND.                       *WWE001*
044200      READ  WW1RQ0  NEXT  INTO  WW1ZRQ0-REG                 *WWE001*
044300          AT  END                                           *WWE001*
044400          MOVE  1  TO  WS-RETORNO-05                        *WWE001*
044500          MOVE  1  TO  WS-FIMLOG-05                         *WWE001*
044600          MOVE  ZEROS  TO  WS-FLG-WW1RQ0                    *WWE001*
```

Continuação da figura 3.5



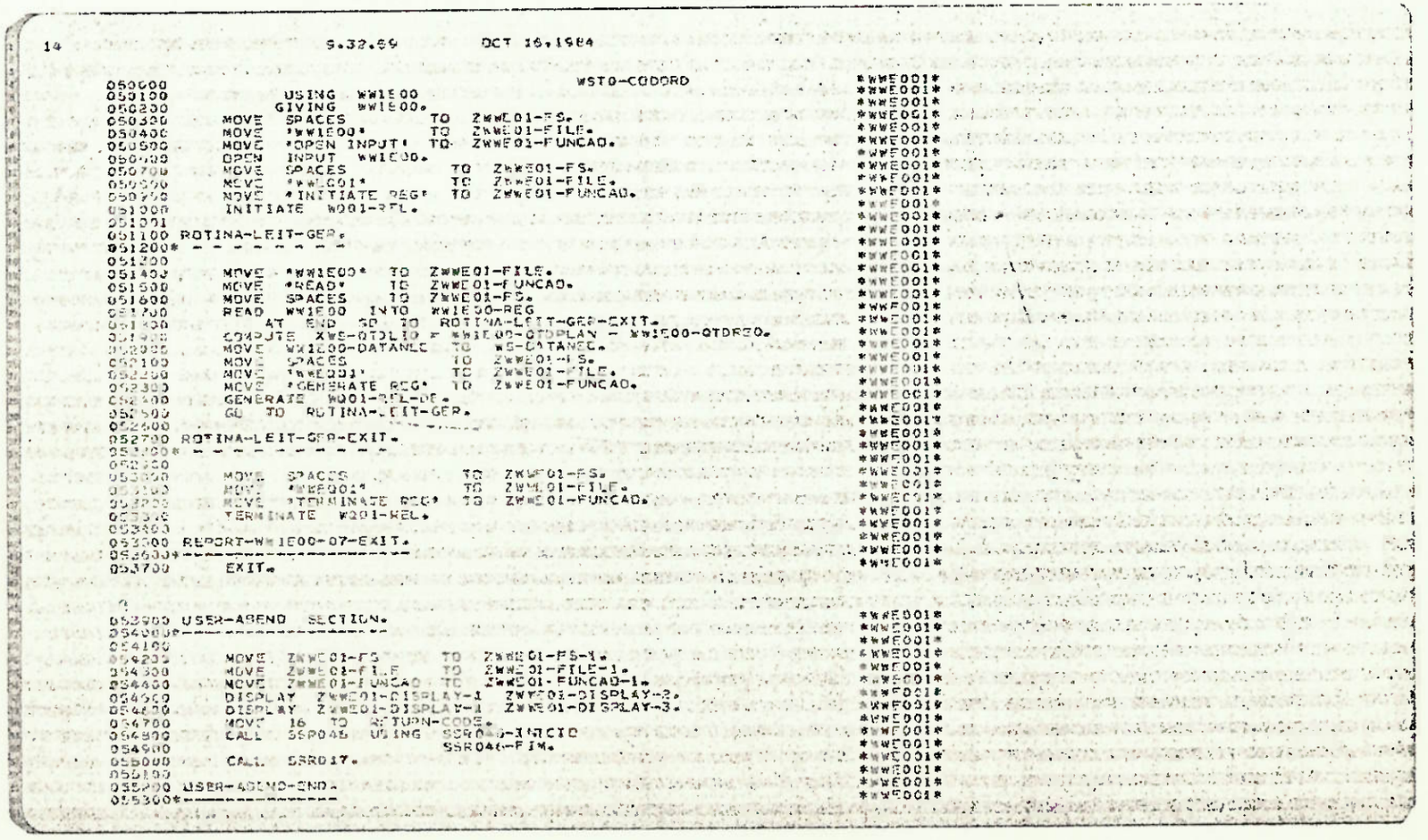
```
13                          9:32:59        OCT 16,1994

044700           GO  TO  GETNEXT-WW1RQ0-05-EXIT.                    *WWE001*
044800      IF  NOT  VSAM-OK                                         *WWE001*
044900          MOVE  VSAM-STATUS   TO  ZWWE01-FS                    *WWE001*
045000          MOVE  'WW1RQ0'      TO  ZWWE01-FILE                  *WWE001*
045100          MOVE  'READ NEXT'   TO  ZWWE01-FUNCAO                *WWE001*
045200          GO  TO  USER-ABEND.                                  *WWE001*
045300      IF  WW1ZRQ0-PREFIXO  NOT  =  'WRQ0'                      *WWE001*
045400          MOVE     1      TO  WS-RETORNO-05                    *WWE001*
045500          MOVE     1      TO  WS-FIMLOG-05                     *WWE001*
045600          MOVE  ZEROS     TO  WS-FLG-WW1RQ0                    *WWE001*
045700          GO  TO  GETNEXT-WW1RQ0-05-EXIT.                      *WWE001*
045800      IF  WW1ZRQ0-PE-ARQ  =  HIGH-VALUES                       *WWE001*
045900          MOVE     1      TO  WS-RETORNO-05                    *WWE001*
046000          MOVE     1      TO  WS-FIMLOG-05                     *WWE001*
046100          MOVE  ZEROS     TO  WS-FLG-WW1RQ0                    *WWE001*
046200          GO  TO  GETNEXT-WW1RQ0-05-EXIT.                      *WWE001*
046300      IF  WW1ZRQ0-CODORD  NOT  =  WW1ZOR0-CODORD               *WWE001*
046400          MOVE     1      TO  WS-RETORNO-05                    *WWE001*
046500          MOVE     1      TO  WS-FIMLOG-05                     *WWE001*
046600          MOVE  ZEROS     TO  WS-FLG-WW1RQ0                    *WWE001*
046700          GO  TO  GETNEXT-WW1RQ0-05-EXIT.                      *WWE001*
046800      IF  WW1ZRQ0-TIPO  =  'MP'                                *WWE001*
046900          MOVE     1      TO  WS-RETORNO-05                    *WWE001*
047000          GO  TO  GETNEXT-WW1RQ0-05-EXIT                       *WWE001*
047100      ELSE                                                     *WWE001*
047200          GO  TO  GETNEXT-WW1RQ0-05-EXIT.                      *WWE001*
047300                                                               *WWE001*
047400 GETNEXT-WW1RQ0-05-EXIT.                                       *WWE001*
047500*-------------------------------                              *WWE001*
047600      EXIT.                                                    *WWE001*


047800 WRITE-WW1E00-06  SECTION.                                     *WWE001*
047900*-------------------------------                              *WWE001*
048000                                                               *WWE001*
048100      MOVE  SPACES    TO  ZWWE01-FS.                           *WWE001*
048200      MOVE  'WW1E00'  TO  ZWWE01-FILE.                         *WWE001*
048300      MOVE  'WRITE'   TO  ZWWE01-FUNCAO.                       *WWE001*
048400      WRITE  WE00-REG  FROM  WW1ZRQ0-REG.                      *WWE001*
048500                                                               *WWE001*
048600 WRITE-WW1E00-06-EXIT.                                         *WWE001*
048700*-------------------------------                              *WWE001*
048800      EXIT.                                                    *WWE001*


049000 REPORT-WW1E00-07  SECTION.                                    *WWE001*
049100*-------------------------------                              *WWE001*
049200                                                               *WWE001*
049300      MOVE  SPACES    TO  ZWWE01-FS.                           *WWE001*
049400      MOVE  'WW1E00'  TO  ZWWE01-FILE.                         *WWE001*
049500      MOVE  'CLOSE'   TO  ZWWE01-FUNCAO.                       *WWE001*
049600      CLOSE  WW1E00.                                           *WWE001*
049700      CALL  SSR004  USING  SSRZ01.                             *WWE001*
049800      SORT  WW1ST0  ON  ASCENDING  KEY  WST0-CODMAT            *WWE001*
049900                                        WST0-DATANEC           *WWE001*
```

Continuação da figura 3.5



```
14                          9.32.69        OCT 16,1984

050000                                                  WSTO-CODORD                 *WWE001*
050100             USING  WW1E00                                                    *WWE001*
050200           GIVING  WW1E00.                                                    *WWE001*
050300      MOVE  SPACES          TO  ZWWE01-FS.                                   *WWE001*
050400      MOVE  'WW1E00'        TO  ZWWE01-FILE.                                 *WWE001*
050500      MOVE  'OPEN INPUT'    TO  ZWWE01-FUNCAO.                               *WWE001*
050600      OPEN  INPUT  WW1E00.                                                   *WWE001*
050700      MOVE  SPACES          TO  ZWWE01-FS.                                   *WWE001*
050800      MOVE  'WWLG01'        TO  ZWWE01-FILE.                                 *WWE001*
050900      MOVE  'INITIATE REG'  TO  ZWWE01-FUNCAO.                               *WWE001*
051000      INITIATE  WQ01-REL.                                                    *WWE001*
051001                                                                             *WWE001*
051100 ROTINA-LEIT-GER.                                                          *WWE001*
051200*- - - - - - - - -                                                         *WWE001*
051300                                                                             *WWE001*
051400      MOVE  'WW1E00'  TO  ZWWE01-FILE.                                       *WWE001*
051500      MOVE  'READ'    TO  ZWWE01-FUNCAO.                                     *WWE001*
051600      MOVE  SPACES    TO  ZWWE01-FS.                                         *WWE001*
051700      READ  WW1E00  INTO  WW1E00-REG                                         *WWE001*
051800          AT  END  GO  TO  ROTINA-LEIT-GER-EXIT.                             *WWE001*
051900      COMPUTE  WWS-QTDLIDO = WW1E00-QTDPLAN - WW1E00-QTDREQ.                 *WWE001*
052000      MOVE  WW1E00-DATANEC  TO  WS-DATANEC.                                  *WWE001*
052100      MOVE  SPACES          TO  ZWWE01-FS.                                   *WWE001*
052200      MOVE  'WWEQ01'        TO  ZWWE01-FILE.                                 *WWE001*
052300      MOVE  'GENERATE REG'  TO  ZWWE01-FUNCAO.                               *WWE001*
052400      GENERATE  WQ01-DEL-DE.                                                 *WWE001*
052500      GO  TO  ROTINA-LEIT-GER.                                               *WWE001*
052600                                                                             *WWE001*
052700 ROTINA-LEIT-GER-EXIT.                                                     *WWE001*
052800*- - - - - - - - - - -                                                     *WWE001*
052900                                                                             *WWE001*
053000      MOVE  SPACES          TO  ZWWE01-FS.                                   *WWE001*
053100      MOVE  'WWEQ01'        TO  ZWWE01-FILE.                                 *WWE001*
053200      MOVE  'TERMINATE REG'  TO  ZWWE01-FUNCAO.                              *WWE001*
053300      TERMINATE  WQ01-REL.                                                   *WWE001*
053400                                                                             *WWE001*
053500 REPORT-WW1E00-07-EXIT.                                                    *WWE001*
053600*-------------------------                                                  *WWE001*
053700      EXIT.                                                                  *WWE001*


052900 USER-AGEND  SECTION.                                                      *WWE001*
053000*--------------------                                                       *WWE001*
053100                                                                             *WWE001*
053200      MOVE  ZWWE01-FS       TO  ZWWE01-FS-1.                                 *WWE001*
053300      MOVE  ZWWE01-FILE     TO  ZWWE01-FILE-1.                               *WWE001*
053400      MOVE  ZWWE01-FUNCAO   TO  ZWWE01-FUNCAO-1.                             *WWE001*
053500      DISPLAY  ZWWE01-DISPLAY-1  ZWWE01-DISPLAY-2.                           *WWE001*
053600      DISPLAY  ZWWE01-DISPLAY-1  ZWWE01-DISPLAY-3.                           *WWE001*
053700      MOVE  16  TO  RETURN-CODE.                                             *WWE001*
053800      CALL  SSR046  USING  SSR046-INICIO                                     *WWE001*
053900                           SSR046-FIM.                                       *WWE001*
054000      CALL  SSR017.                                                          *WWE001*
054100                                                                             *WWE001*
054200 USER-AGEND-END.                                                           *WWE001*
054300*---------------                                                            *WWE001*
```

Continuação da figura 3.5

por um programador com uma experiência razoável, poderia ser gerado em questão de horas através do pré-compilador Meta-Cobol. Não há necessidade do leitor entender a particularidade dos comandos codificados no programa fonte, e muito menos ainda, os meta-comandos do programa original. O único aspecto que ressaltamos, é o de se perceber os ganhos de produtividade que estariam por trás do Meta-Cobol, e da consequência que seria oriunda dos mesmos, ou seja, grande redução do tempo efetivo de desenvolvimento de programas "batch".

Apenas analisando a figura 3.4 em comparação com a figura 3.5 , podemos observar que praticamente cada meta-comando do programa original, corresponderia a uma sub-rotina na Procedure Division do programa fonte gerado (em COBOL).Por exemplo, o primeiro meta-comando codificado é o

```
*START WW10RØ KEY(FIRST)  RETURN(OK);
```

O pré-compilador, ao detectá-lo, expandiria no programa fonte, a subrotina delimitada pelos seguintes nomes de parágrafos:

```
*START-WW10RØ-Ø1  SECTION.
                  .
                  .
                  .
                  .
*START-WW10RØ-Ø1-EXIT.
```

Observe-se pela figura 3.5 que essa sub-rotina seria constituída de 11 linhas de comandos normais do COBOL. A sua função seria o de se posicionar o arquivo WW10RØ, armazenado no DB VSAM, na posição inicial (primeiro registro) de modo a se proceder uma leitura sequencial. O programador não teria de se preocupar com os detalhes envolvidos nesta operação, bastaria sintetizar tudo através de um único meta-comando, o resto seria gerado automaticamente.

Para finalizar, diríamos que da figura 3.4, podemos observar mais três aspectos:

a) Esse programa em particular, podem ser completamente especificado em meta-linguagem, pois não houve necessidade de complementação com comandos normais do COBOL.

b) A "Identification Division", a "Environment Division" e a "Data Division" (inclusive a "Report-Section") seriam totalmente geradas pelo prē-compilador, pressupondo que as suas especificações estivessem todas armazenadas no Dicionārio de Dados.

c) Conforme a potencialidade do meta-comando a ser expandido o tamanho da sub-rotina a ser expandida varia muito. Em particular, o meta-comando "*REPORT" implica em gerar a "Report Section", alēm de uma rotina específica na "Procedure Division".

4. A SITUAÇÃO ATUAL

4.1. APRESENTAÇÃO DA SITUAÇÃO ATUAL

Neste capítulo, procuraremos apresentar a situação atual do ambiente de desenvolvimento de sistemas de informação, procurando localizar mais especificamente onde é que a alternativa em cogitação, deverá influir com maior intensidade para podermos mais adiante, elaborar um modelo representativo dessa realidade e partir para extrapolações quanto às possíveis repercussões no ambiente.

4.1.1. O SISTEMA DE TRABALHO DO PESSOAL DE DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO

O pessoal envolvido nessa atividade está organizado conforme o organograma apresentado na figura 4.1 e se divide basicamente em 2 grandes grupos: analistas de sistemas e programadores. Apresentaremos agora a participação de cada grupo nessa atividade e o seu inter-relacionamento.

A. Analistas de Sistemas

São agrupados em equipes sob a orientação de um supervisor. Cada equipe possui sala própria e está voltada às aplicações para empresas que constituem determinado setor do grupo. A única exceção é a equipe para Pacotes Padronizados.

Os sistemas são desenvolvidos geralmente sob a responsabilidade de apenas um ou dois analistas. Mas há uma constante interação de experiência e informações entre todos os analistas da equipe e mesmo com elementos de outras equipes. O supervisor procura acompanhar o desenvolvimento de todos os sistemas na sua equipe. Além de verificar o progresso dos trabalhos, normalmente é muito requisitado para dar assessoria técnica.

No desenvolvimento de novos sistemas, os analistas são responsáveis pelo levantamento dos seus requisitos (neces

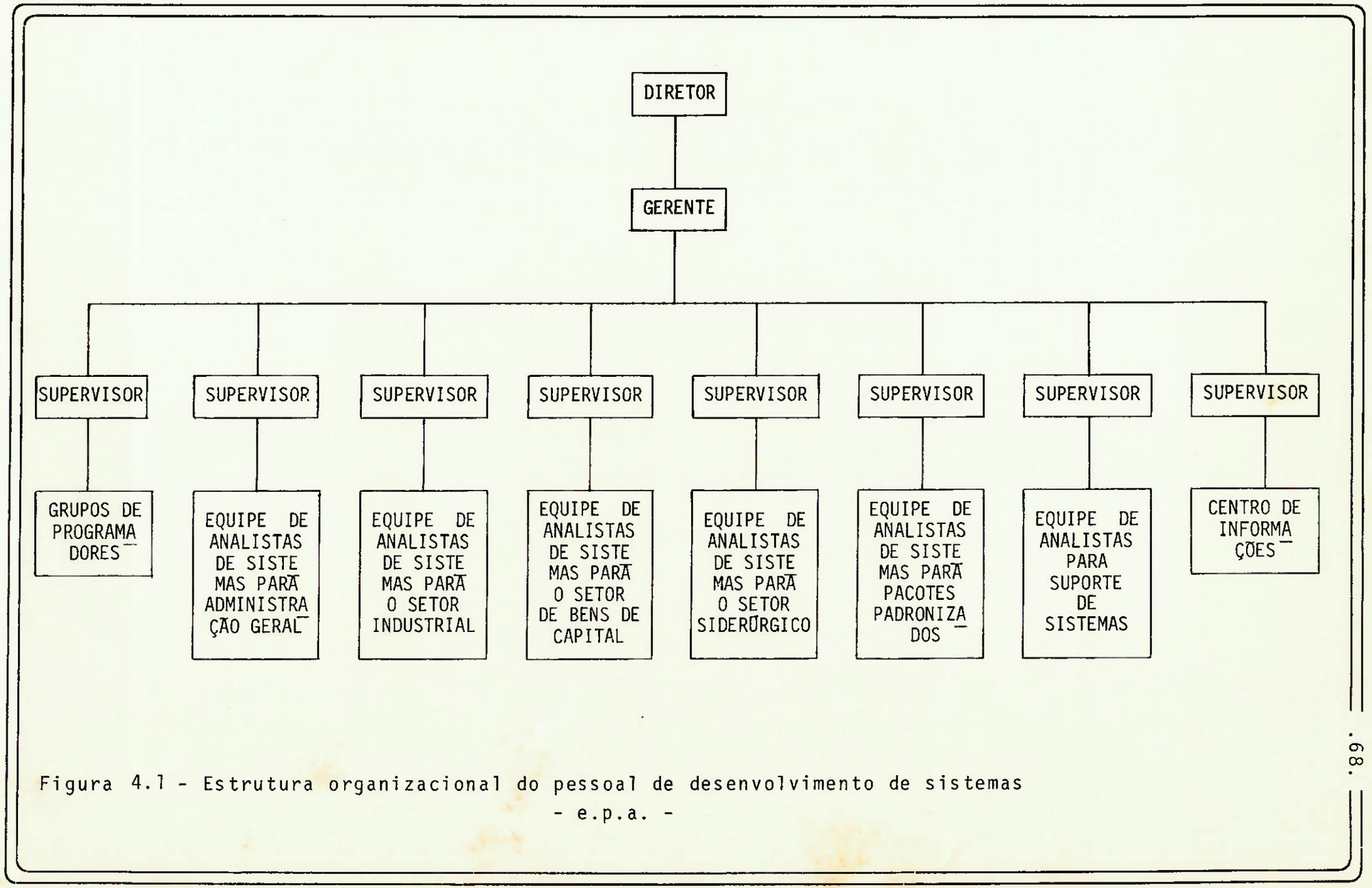


Figura 4.1 - Estrutura organizacional do pessoal de desenvolvimento de sistemas - e.p.a. -

sidades), pela análise do sistema, pelo projeto do sistema, e pela implantação e testes do sistema. Normalmente assumem também trabalhos de documentação. Durante todo o desenvolvimento, frequentemente, realizam visitas e reuniões com as pessoas solicitantes e futuros usuários do sistema. Na parte final do desenvolvimento, responsabilizam-se pelo preparo de manuais (os manuais são "desenhados" no próprio terminal de video, e gerados por meio de um programa "batch"), pela colocação do sistema em equipamentos de produção (todos os sistemas são desenvolvidos num ambiente à parte com equipamentos próprios) e pelas rotinas de segurança do sistema.

Quanto à atividade de programação os analistas tomam atitudes diferentes conforme o caso do programa. Se o programa for muito urgente e os programadores estiverem muito sobrecarregados ou trabalhando em programas igualmente prioritários, ele próprio assume o trabalho de programação. Os analistas são, afinal, os maiores responsáveis pelos sistemas que desenvolvem. Mas, normalmente, passam os programas para serem desenvolvidos pelo programador e para isso, preparam antes a chamada "Pasta do Programa", onde procura, objetivamente, especificar a função a ser desempenhada pelo programa, suas características e outras informações complementares. Os analistas não se limitam apenas a isso. Antes que o programador escolhido passe a trabalhar no desenvolvimento do programa, procuram ter uma breve reunião com o mesmo para se certificarem que tudo foi bem compreendido.

Quanto maior o tempo estimado para o desenvolvimento do programa, maior é a tendência do analista em delegar a sua programação a um programador. Quando o programa é simples e o tempo estimado para o seu desenvolvimento é curto, o analista geralmente procura assumir o seu desenvolvimento, principalmente, se pressentir que o tempo necessário para explicar ao programador o que deve ser feito não seria, relativamente, compensador. Um fator muito importante que os analistas têm observado antes de se decidir pela passagem ou não de um programa a um programador, é a disponibilidade de uma ferramenta de "software" que facilite o trabalho de sua programação. Exem

plos típicos são o SCOA e o Gerador de Relatórios. Se o programa desejado puder ser rapidamente desenvolvido por meio de uma dessas ferramentas, o analista, naturalmente, prefere assumir o seu desenvolvimento.

Os analistas constituem dentro do ambiente de processamento de dados, uma classe de funcionários relativamente bem remunerada, mas as suas responsabilidades são maiores, e, embora não possuam horário de trabalho rígido (não precisam "bater" cartão de ponto), normalmente trabalham mais que as 9 horas regulamentares de trabalho por dia, sem receberem remuneração "extra".

B. Programadores

Em comparação com os analistas, os programadores têm uma participação muito menor no trabalho de desenvolvimento de sistemas. A sua participação limita-se basicamente à continuação do desenvolvimento de programas cuja necessidade é identificada pelos analistas durante a fase de projeto de sistemas. Os analistas, uma vez decididos pela automatização de determinado processo (observando-se as restrições e os recursos disponíveis) mediante introdução de um sistema, procura defini-lo a princípio, em linhas gerais e em seguida, com maior detalhe especificando a Pasta do Programa. O programador dando continuidade ao desenvolvimento do programa, assume as seguintes etapas:

a) Estudo do programa, com base na Pasta do Programa e em outras informações passadas pelo analista responsável (na reunião) visando a sua codificação;

b) Codificação do programa para se criar um programa fonte;

c) Depuração do programa fonte, submetendo-o a compilação(ões) e link-edição(ões) e eliminando eventuais erros de sintaxe e/ou de lógica. O objetivo é o de se gerar o programa objeto correspondente.

d) Testes do Programa[1] com uma massa de dados representativa da realidade. O programador procura testar o programa sob todas as casuísticas possíveis que possam ocorrer na operação.

Devemos esclarecer ao leitor que, embora comumente se fale que o programador é que faz o desenvolvimento do programa, na realidade ele apenas o continua. A rigor, o papel assumido pelo programador é o de viabilizar e concretizar o programa. Mas de qualquer forma, por comodidade de expressão, continuaremos a chamar de "desenvolvimento do programa" todo o processo constituído por essas 4 etapas. Em particular, consideraremos o conjunto formado pelas etapas a) e b) como sendo a atividade de programação.

Entre as etapas b) e c) existe na realidade uma etapa intermediária que é a digitação, do programa fonte, pois o mesmo é armazenado em memória em discos via terminal de vídeo. Este trabalho é feito pelas digitadoras. Normalmente o prazo para a execução desse serviço é de um dia. A figura 4.2 ilustra esse processo de desenvolvimento de programas novos[2] que normalmente se aplica aos programas "batch".

Quando o programador necessita executar um serviço de digitação simples como correções no programa fonte, inserção de rotinas complementares, alterações e acabamento, ele

---

[1] Convém ressaltar que o trabalho de testes do sistema é realizado pelo analista. O programador testa o programa isoladamente. O analista é que verifica se o referido programa funciona satisfatoriamente do ponto de vista sistêmico, isto é, observando a sua integração e entrosamento com os demais componentes do sistema a que se destina.

[2] O processo aqui descrito refere-se ao desenvolvimento de programas novos seguindo a metodologia convencional, baseado na codificação do programa segundo uma linguagem tradicional e procedural como o COBOL. Isso porque é segundo esse processo que a maioria dos programas "batch" são atualmente desenvolvidos. O desenvolvimento de programas "on line" utilizando, por exemplo, uma ferramenta como o SCOA (técnica não procedural) envolve diferenças gritantes, mas não serão aqui abordados pois fogem do escopo do trabalho.

RESPONSÁVEL / ETAPA	Analista	Programador	Digitadora
Identificação da necessidade do programa	•		
Definição do programa em linhas gerais	•		
Especificação da pasta do programa com maior detalhe	•		
Estudo do programa		•	
Codificação do programa		•	
Digitação do programa			•
Compilação(ões) e depuração do programa		•	
Testes do programa		•	
Integração do programa ao sistema	•		

Figura 4.2 - Etapas do desenvolvimento de programas novos
- e.p.a. -

mesmo o faz, pois há um grande número de terminais de video à sua disposição na sala da Programação.

Convém observar que por trás dessa sistemática de delegar trabalhos menos nobres à mão-de-obra mais barata (analista passa a Pasta do Programa para programador, e este passa digitação para as digitadoras) há um argumento muito forte. A mão-de-obra que atua na área da Informática continua sendo relativamente cara (em relação aos padrões do mercado de trabalho), e somente as despesas com pessoal (salários e encargos sociais), segundo dados da SUCESU-SP (Sociedade dos Usuários de Computadores e Equipamentos Subsidiários de São Paulo), correspondem a cerca de 60% dos custos totais na área de processamento de dados. Deste percentual 46% cabem aos programadores e analistas.

Na Programação do Setor de Informática e Sistemas de Teleprocessamento, existem 6 grupos de programadores, sendo um grupo designado para cada equipe de análise (5) e um para cuidar especificamente de certos sistemas já em operação (Ex. Folha de Pagamentos).

Cada equipe de programadores é coordenada pelo seu elemento mais experiente, qualificado como líder de grupo. Todos os grupos respondem a um único supervisor. A designação de um grupo específico para cada equipe de análise, contribui para um melhor entrosamento entre analistas e programadores e dinamiza o seu inter-relacionamento. Geralmente um ou dois programadores são designados para acompanhar o desenvolvimento de um sistema. O(s) analista(s) responsável(is) pelo sistema procura(m) dar uma noção global do projeto ao(s) programador(es), mantê-lo(s) informado(s) quanto à evolução do mesmo e passar-lhe(s) serviços de programação quando necessário e conveniente.

A figura 4.3 ilustra este relacionamento entre um grupo de programadores e a respectiva equipe de análise. As linhas pontilhadas representam o vínculo existente entre analista(s) e programador(es) envolvidos no desenvolvimento de um mesmo sistema.

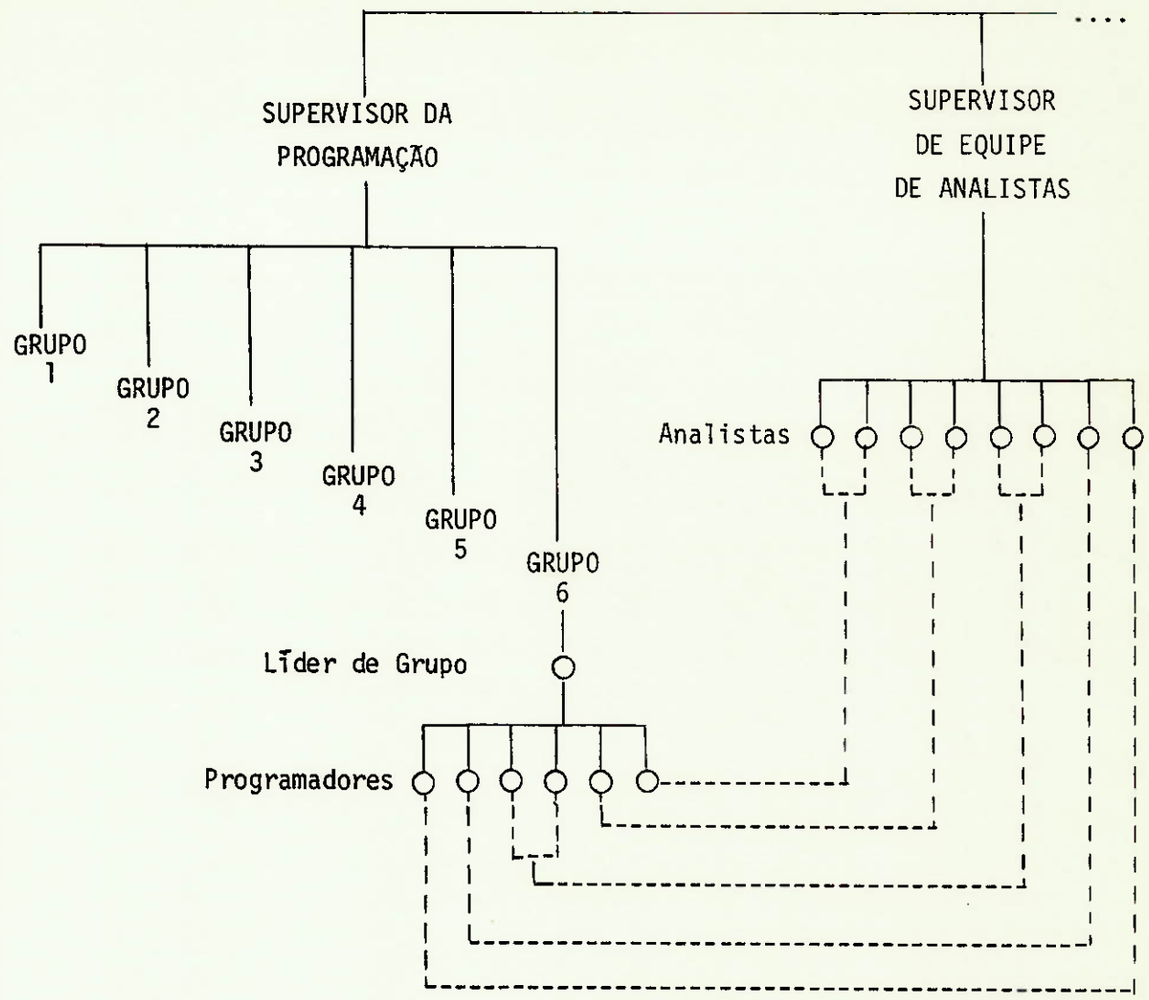


Figura 4.3 - Relacionamento do grupo de programadores com a respectiva equipe de analistas
- e.p.a. -

Os programadores recebem um tratamento distinto do que é dado aos analistas. O seu horário de trabalho é controlado por meio de cartões de ponto e normalmente trabalham somente durante o horário do expediente. O trabalho em regime de hora-extra é reconhecido e remunerado.

4.1.2. O USO DE FERRAMENTAS DE "SOFTWARE" NO AMBIENTE

Como o leitor já deve ter notado, o uso de ferramentas de "software" já faz parte integrante da "cultura" do Setor de Informática e Serviços de Teleprocessamento na atividade de desenvolvimento de sistemas.

É importante observar que elas foram concebidas para atender a necessidades bem específicas, não sendo de forma alguma (nem poderiam pretender ser) capazes de atender aos usuários para fins genéricos, ou seja, as ferramentas de "software" são limitadas.

Como exemplo, citaremos o caso do SCOA que se destina à geração de programas fonte em COBOL (não chega a criar o respectivo programa objeto. Essa tarefa é feita pelo próprio usuário submetendo o programa fonte gerado a uma compilação e link-edição) para aplicações "on-line", já levando-se em consideração características do ambiente como método de acesso a arquivos VSAM, monitor de TP CICS e padronização de códigos da empresa (norma). Observe que em ambientes não compatíveis ele simplesmente não seria aproveitável.

Outra importante constatação é a de que mesmo entre os programas "on-line", a aplicabilidade do SCOA é maior ou menor conforme as suas características. A figura 4.4 ilustra o fato de que quanto mais as rotinas padrões geradas pelo SCOA são suficientes para a lógica do programa, há menos necessidade de codificação e inserção de lógicas complementares (esses trechos a serem codificados pelo próprio usuário são chamados de "patches"). No outro extremo, existem aqueles programas que por exigirem "patches" demais no programa fonte a ser gerado, tornam-se inviáveis de serem desenvolvidos pelo SCOA, e são inteiramente "codificados na unha", ou seja, são programados proceduralmente na linguagem COBOL (método convencional).

Geralmente, os programas fáceis e rápidos de serem gerados têm ficado a cargo dos analistas. A maioria dos programas que apresentam média aplicabilidade da ferramenta, os

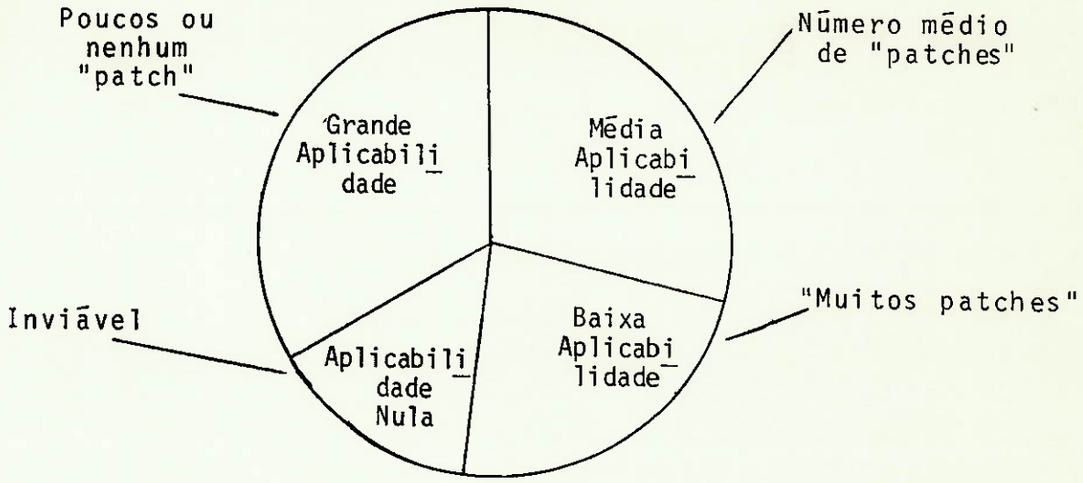


Figura 4.4 - Distribuição dos programas "on-line" conforme a aplicabilidade do SCOA para sua geração
- e.p.a. -

que são de baixa aplicabilidade e os de aplicabilidade nula são passados aos programadores.

Analogamente, as considerações feitas quanto ao aproveitamento do SCOA para a geração de programas "on-line", também podem ser extendidas ao aproveitamento do Gerador de Relatórios.

Outro aspecto interessante, é o de que o ganho de produtividade esperado com o aproveitamento dessas ferramentas é função das características do programa a ser desenvolvida. Quanto maior é a aplicabilidade da ferramenta ao desenvolvimento de um programa, maior é a redução no tempo necessário à programação do mesmo.

A figura 4.5 ilustra qualitativamente este fenômeno, pois as grandezas envolvidas são muito difíceis de se mensurar. Observe que na origem não há variação no tempo de programação

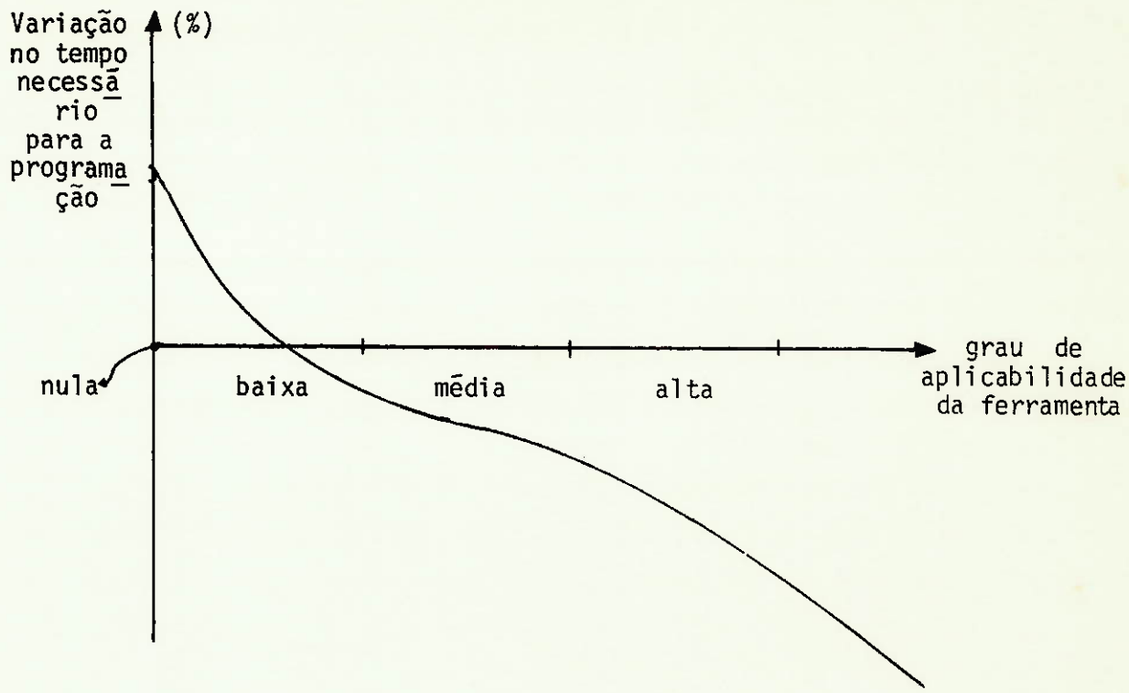


Figura 4.5 - Gráfico do ganho de produtividade proporcionável por uma ferramenta de "software" conforme a sua aplicabilidade

- e.p.a. -

pois corresponde ao caso onde a ferramenta não pode ser aplicada. Na região de baixa aplicabilidade um programa pode até mesmo ter um acréscimo no tempo de programação se o usuário optar por utilizar a ferramenta (gastaria-se muito tempo "remendando" o programa gerado).

Comentamos aqui somente o uso do SCOA e do Gerador de Relatórios, por serem os mais importantes, os mais antigos e os mais usados no ambiente.

4.1.3. TRABALHOS DE MANUTENÇÃO E OUTRAS ATIVIDADES

O desenvolvimento de novos Sistemas de Informação é apenas uma das atividades dos programadores e analistas. Uma outra atividade, não menos importante, é o da manutenção desses Sistemas de Informação. Vimos que um sistema que entra em operação, poderá apresentar durante a sua vida, necessidades de eventuais manutenções. É por meio dessas manutenções que se consegue prolongar a sua vida útil.

As manutenções variam de alterações simples nos programas até complicadas reformulações em considerável parte do programa. Às vezes, a necessidade de uma manutenção mais drástica pode determinar o fim de um programa e sua substituição por um outro novo.

Os trabalhos de manutenção para sistemas já instalados e em produção, são geralmente, mais prioritários que as tarefas relacionadas com o desenvolvimento de novos sistemas.

As atividades de manutenção e de desenvolvimento de novos sistemas são qualificadas como "atividades produtivas". Existem também outras atividade que não são encaradas como produtivas, mas que são também importantes, tais como: participação em cursos internos e palestras, reuniões, consulta a manuais, orientação e treinamento de pessoal, etc.

4.1.4. DADOS DO ATUAL NÍVEL DE PRODUÇÃO

Embora o Setor de Informática e Serviços de Teleprocessamento esteja interessado na questão da produtividade, surpreendemente, não possuía nenhuma base de dados, ou informações na qual pudéssemos apoiar o desenvolvimento do presente trabalho. Quando se pretende analisar o problema da produtividade, é necessário termos uma base de dados para fins de comparação com uma outra situação.

Foi, portanto, necessário desenvolver um trabalho de coleta de dados que pudesse refletir a atual situação. Os dados coletados são basicamente de dois tipos: dados do atual nível de produção e dados dos atuais tempos de atendimento. Neste subitem abordaremos o primeiro tipo e no subitem seguinte o outro.

Os dados de produção da Programação foram coletados dos Relatórios de Acompanhamento da Programação referentes ao período entre setembro de 1983 e abril de 1984. Estes relatórios foram preparados somente durante esses 8 meses, sendo depois abandonados. A figura 4.6 mostra um exemplar desses relatórios.

Como os seus dados, da maneira e pelo detalhe em que são apresentados, não permitem uma rápida e clara visualização do comportamento das atividades da Programação, com base em tais relatórios, elaborou-se uma série de tabelas auxiliares mais convenientes ao nosso trabalho. Essas tabelas mostram ao longo desse período, o número de programas novos e manutenções que foram recebidos e liberados pela Programação em cada mês. Elaboramos também, tabelas mostrando ao fim de cada mês o número de programas novos e manutenções que sobraram nas mãos da Programação (chamadas de programas "em mãos"). Diz se que um programa está "em mãos", quando ainda está na fila para ser atendido ou quando já começou a ser atendido por um programador, mas não foi ainda liberado.

Os programas novos e manutenções serão classificados segundo 5 tipos, quais sejam:

- "batch" de atualização;

- "batch" de relatório;

- "on-line" gerado pelo SCOA;

- "on-line" programado convencionalmente (sem SCOA);

- programado em outras linguagens (não COBOL).

Optou-se por tratar os dados de foram mais simplificada, à nível geral (totais da Programação) e não de cada grupo) e diminuindo a variedade da classificação (nos relatórios são considerados 7 tipos), pois estamos interessados na situação geral desse ambiente que é a Programação e não nos detalhes que cercam cada grupo. Os programas "batch", onde se concentra o interesse deste trabalho, continuam a ser tratados como na classificação original dos relatórios, ou seja, divididos em programas de atualização e de relatório. Os programas "batch" de atualização são aqueles que processam grande volume de dados (em lotes) para a atualização de arquivos. Conforme a sua função, processam cálculos, formatações, classificações, consistências cruzadas, etc. Os programas "batch" de relatório destinam-se à emissão de relatório.

Neste trabalho adotaremos a codificação constante na tabela 4.1 para nos referirmos às atividades de manutenção e desenvolvimento de programas novos do pessoal da Programação.

CÓDIGO	DESCRIÇÃO DA ATIVIDADE
10	Desenvolvimento de programa "batch" de atualização novo
11	Manutenção de programas "batch" de atualização
12	Desenvolvimento de programa "batch" de relatório novo
13	Manutenção de programas "batch" de relatório
14	Desenvolvimento de programas "on-line" com SCOA
15	Manutenção de programas "on-line" gerados com SCOA
16	Desenvolvimento de programas "on-line" sem SCOA
17	Manutenção de programas "on-line" programados convencionalmente (sem SCOA)
18	Desenvolvimento de programas novos em outras linguagens que não COBOL
19	Manutenção de programas em outras linguagens que não COBOL

Tabela 4.1 - Codificação das Atividades da Programação
- e.p.a. -

. ACOMPANHAMENTO DA EQUIPE/PROGRAMAÇÃO PROGRAMAÇÃO — MÊS/ANO 04 / 84

NOME DO PROGRAMADOR / NOME DA EQUIPE	DIAS GASTOS		SITUAÇÃO	BATCH ATUALIZAÇÃO		BATCH RELATORIO		ON-LINE SCOA CONSULTA		ON-LINE SCOA ATUALIZAÇÃO		ON-LINE SEM SCOA CONSULTA		ON-LINE SEM SCOA ATUALIZAÇÃO		OUTRAS LINGUAGENS		TOTAL	
	PROGR.	OUTROS		NOVOS	MANUT.	NOVOS	MANUT.	NOVOS	MANUT.	NOVOS	MANUT.	NOVOS	MANUT.	NOVOS	MANUT.	NOVOS	MANUT.	NOVOS	MANUT.
AG	46	30	EM MÃOS		1	1									1			1	2
			LIBERADOS NO MÊS	9	7	1	1	1							2		18	11	28
AVSA VIB	204	24	EM MÃOS	3	1	9	3	5		2	13	5		18	8			42	25
			LIBERADOS NO MÊS		3	1	2			3		8	8	12	47			24	60
EVSA	130	22	EM MÃOS			2			2					1				3	2
			LIBERADOS NO MÊS	5	4	7	26		8		2		9	4	48			16	97
IVSA	85	70	EM MÃOS	2	16	3	9				1		6	1	22	5		11	54
			LIBERADOS NO MÊS	2	13	3	6						6		20	3		8	45
PACOTE	80	30	EM MÃOS			2	1	1		22								25	1
			LIBERADOS NO MÊS	5	2	3	7			5	10		1		7	1		14	27
BI	50	24	EM MÃOS	2	5	1	10								2			3	17
			LIBERADOS NO MÊS	2	5	5	8								2			7	13
			EM MÃOS																
			LIBERADOS NO MÊS																
			EM MÃOS																
			LIBERADOS NO MÊS																
			EM MÃOS																
			LIBERADOS NO MÊS																
			EM MÃOS																
			LIBERADOS NO MÊS																
			EM MÃOS																
			LIBERADOS NO MÊS																
TOTAL	595	200	EM MÃOS	7	23	18	23	6	2	24	14	5	6	20	33	5		85	101
			LIBERADOS NO MÊS	23	34	20	50	1	8	8	12	8	24	16	126	4	18	80	272

Figura 4.6 - Exemplo da Folha de Acompanhamento da Programação

Observe-se que reservamos os números ímpares para os códigos das atividades de manutenção. Os números pares correspondem ao desenvolvimento de programas novos.

Como nos relatórios não são fornecidos dados quanto ao número de programas (tanto novos como os para manutenção) recebidos pela Programação em cada mês, esse dado foi obtido pela seguinte expressão (considerando-se a Programação como um volume de controle):

$$NPEM_i = NPEM_{i-1} + NPRC_i - NPLB_i$$

$$\boxed{NPRC_i = NPEM_i - NPEM_{i-1} + NPLB_i}$$

onde:

$NPRC_i$ = Número de Programas Recebidos no mês i;

$NPEM_i$ = Número de Programas Em Mãos ao final do mês i;

$NPLB_i$ = Número de Programas Liberados no mês i.

Todas as tabelas auxiliares com os dados de número de programas recebidos, liberados e em mãos são apresentadas nas páginas seguintes. Observe-se que nas últimas colunas são dados o número médio de programas novos ou manutenções no período por tipo e a participação relativa (distribuição percentual) dos mesmos.

Observe-se ainda a existência de uma outra tabela (tabela 4.8) que mostra o aproveitamento do tempo útil (tempo referente ao horário de trabalho regulamentado e remunerado dos programadores) entre atividades "produtivas" (em cima de programas) e outras atividades. Neste caso, chamamos de outras atividades, aquelas citadas no subitem 4.1.3, tais como participação em cursos e palestras, orientação e treinamento de pessoal, reuniões e atendimento de visitas, mais as even

PROGRAMAS RECEBIDOS PELA PROGRAMAÇÃO EM CADA MÊS

Ano	1983			1984				Média no Período	Distribuição Relativa (%)
Mês / Tipo	10	11	12	01	02	03	04		
"batch" de atualização	13	37	33	23	12	17	7	20,3	26,2
"batch" de relatório	0	30	18	25	0	11	21	15,0	19,4
"on-line" com SCOA	6	26	40	23	26	24	2	21,0	27,1
"on-line" sem SCOA	16	34	34	0	0	4	34	17,4	22,5
em outras linguagem (não COBOL)	5	1	0	2	6	10	2	3,7	4,8
TOTAL	40	128	125	73	44	66	66	77,4	100,0

Tabela 4.2 - Número de programas novos recebidos por mês para serem desenvolvidos
- e.p.a. -

Ano	1983			1984				Média no Período	Distribuição Relativa (%)
Mês / Tipo	10	11	12	01	02	03	04		
"batch" de atualização	35	82	66	70	35	81	21	55,7	18,1
"batch" de relatório	90	84	122	8	50	117	25	70,9	23,1
"on-line" com SCOA	10	70	24	1	42	60	29	33,7	11,0
"on-line" sem SCOA	38	197	165	269	15	96	158	134,0	43,5
em outras linguagem (não COBOL)	4	7	12	8	19	27	16	13,3	4,3
TOTAL	177	440	389	356	161	381	249	307,6	100,0

Tabela 4.3 - Número de programas para manutenção recebidos por mês
- e.p.a. -

PROGRAMAS LIBERADOS PELA PROGRAMAÇÃO EM CADA MÊS

Ano	1983				1984				Média no Período	Distribuição Relativa (%)
Mês / Tipo	09	10	11	12	01	02	03	04		
"batch" de atualização	20	24	27	34	15	18	24	23	23,1	28,3
"batch" de relatório	33	27	32	19	23	14	15	20	22,9	28,2
"on-line" com SCOA	4	3	14	16	43	21	18	9	16,0	19,6
"on-line" sem SCOA	8	18	31	24	3	17	10	24	16,9	20,7
em outra lingua gem (não COBOL)	0	5	1	0	2	5	4	4	2,6	3,2
TOTAL	65	77	105	93	86	75	71	80	81,5	100,0

Tabela 4.4 - Número de programas novos liberados por mês
- e.p.a. -

Ano	1983				1984				Média no Período	Distri-buição Relativa (%)
Mês / Tipo	09	10	11	12	01	02	03	04		
"batch" de atualização	15	32	72	69	56	62	63	34	50,4	15,6
"batch" de relatório	38	90	92	105	51	68	93	50	73,4	22,7
"on-line" com SCOA	22	7	47	26	20	34	66	20	30,2	9,4
"on-line" sem SCOA	231	99	140	157	230	153	98	150	157,3	48,7
em outra lingua gem (não COBOL)	0	4	5	12	7	21	26	18	11,6	3,6
TOTAL	306	232	356	369	364	338	346	272	322,9	100,0

Tabela 4.5 - Número de programas para manutenção liberados por mês
- e.p.a. -

PROGRAMAS EM MÃOS DA PROGRAMAÇÃO AO FIM DE CADA MÊS

Ano	1983				1984				Média no Período	Distribuição Relativa (%)
Mês / Tipo	09	10	11	12	01	02	03	04		
"batch" de atualização	30	19	29	28	36	30	23	7	25,3	22,3
"batch" de relatório	63	36	34	33	35	21	17	18	32,1	28,2
"on-line" com SCOA	7	10	22	46	26	31	37	30	26,1	23,0
"on-line" sem SCOA	30	28	31	41	38	21	15	25	28,6	25,1
em outra linguagem (não COBOL)	0	0	0	0	0	1	7	5	1,6	1,4
TOTAL	93	130	116	148	135	104	99	85	113,7	100,0

Tabela 4.6 - Número de programas novos para desenvolvimento em mãos ao fim de cada mes

- e.p.a. -

Ano	1983				1984				Média no Período	Distribuição Relativa (%)
Mês / Tipo	09	10	11	12	01	02	03	04		
"batch" de atualização	21	24	34	31	45	18	36	23	29,0	15,4
"batch" de relatório	76	76	68	85	42	24	48	23	55,3	29,4
"on-line com SCOA	0	3	26	24	5	13	7	16	11,8	6,3
"on-line sem SCOA	128	67	124	132	171	33	31	39	90,6	48,2
em outra linguagem (não COBOL)	0	0	2	2	3	1	2	0	1,2	0,7
TOTAL	225	170	254	274	266	89	124	101	187,9	100,0

Tabela 4.7 - Número de programas para manutenção em mãos ao fim de cada mês

- e.p.a. -

Ano	1983				1984				Média no Período	Distribuição Relativa (%)
Mês / Atividade	09	10	11	12	01	02	03	04		
Atividade com Programas	778	745	717	824	720	599	667	595	705,6	79,9
Outras Atividades	174	175	156	144	218	213	140	200	177,5	20,1
Total	952	920	873	968	938	812	807	795	883,1	100,0
Parcela do tempo com outras atividades(%)	18,3	19,0	17,9	14,9	23,2	26,2	17,4	25,2		

Tabela 4.8 - Aproveitamento do tempo útil da Programação para atividades com programas e outras atividades. Os números correspondem ao produto entre o número de dias úteis no mês pelo efetivo de programadores no mesmo mês.

- e.p.a. -

tuais faltas e atrasos (critério da Programação para o preenchimento dos relatórios).

Os dados globais (totais para a Programação) obtidas das mesmas são mostradas em forma gráfica nas figuras 4.7 e 4.8 que se referem respectivamente ao trabalho da Programação de desenvolvimento de programas novos (tabelas 4.2, 4.4 e 4.6) e de manutenção de programas (tabelas 4.3, 4.5 e 4.7). Esses gráficos dão uma melhor noção a respeito do comportamento do fluxo dos programas que passam pela Programação. A figura 4.9 ilustra os dados de aproveitamento do tempo útil fornecidos pela tabela 4.8.

As linhas pontilhadas na horizontal representam as médias de cada caso no período em questão.

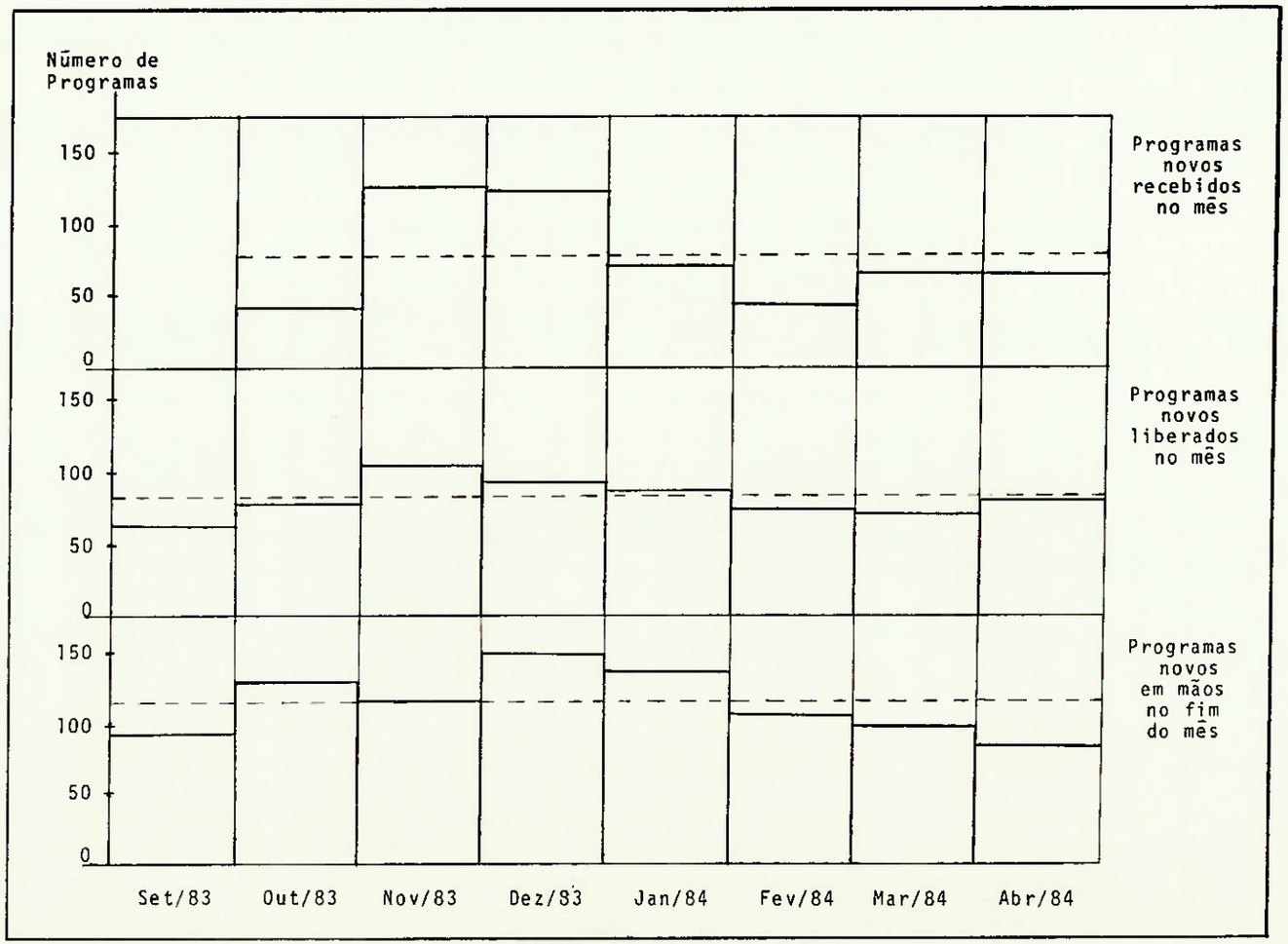


Figura 4.7 - Representação gráfica do fluxo de programas novos que passa pela Programação - e.p.a. -

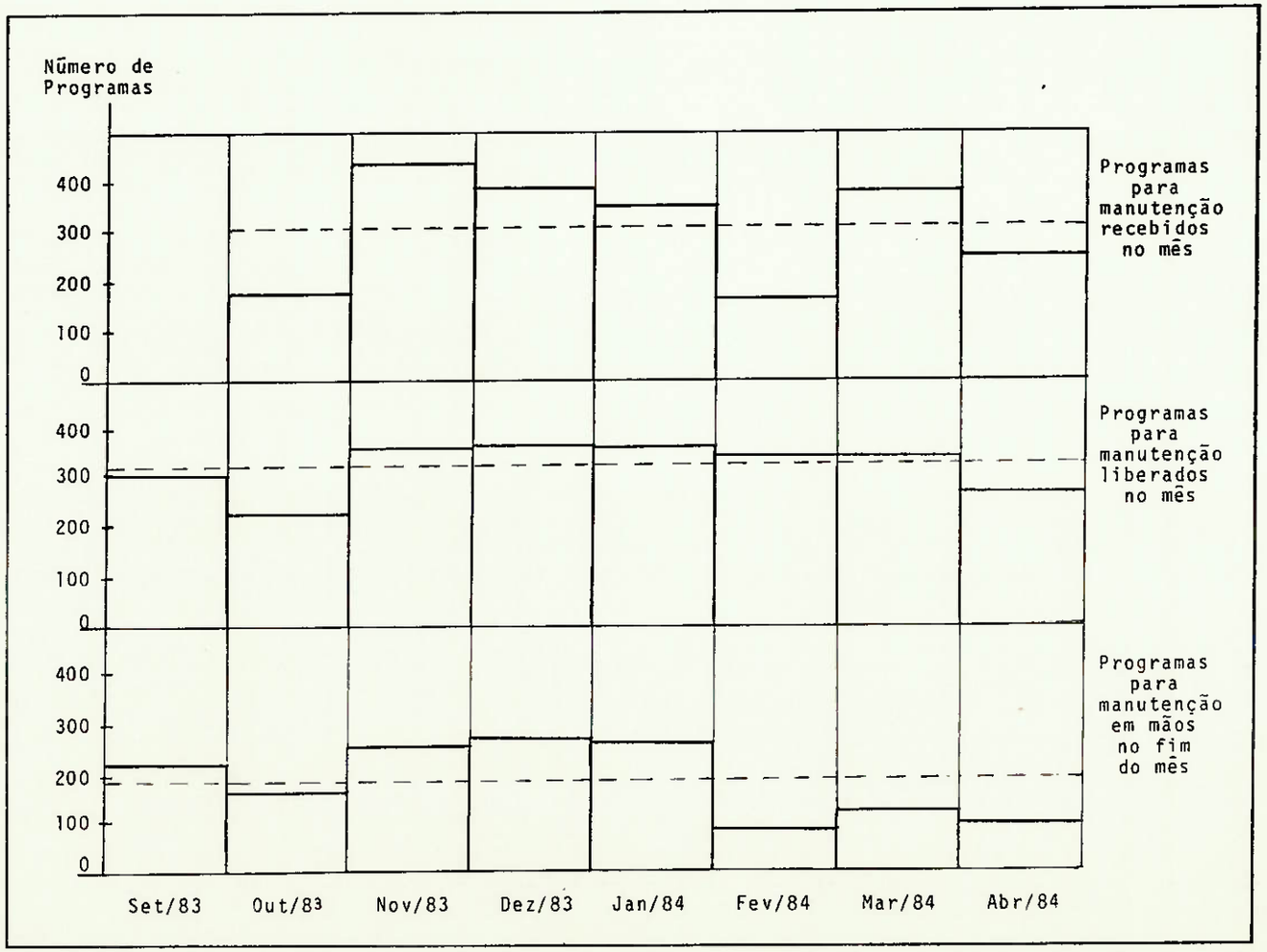


Figura 4.8 - Representação gráfica do fluxo de programas para manutenção que passa pela Programação - e.p.a. -

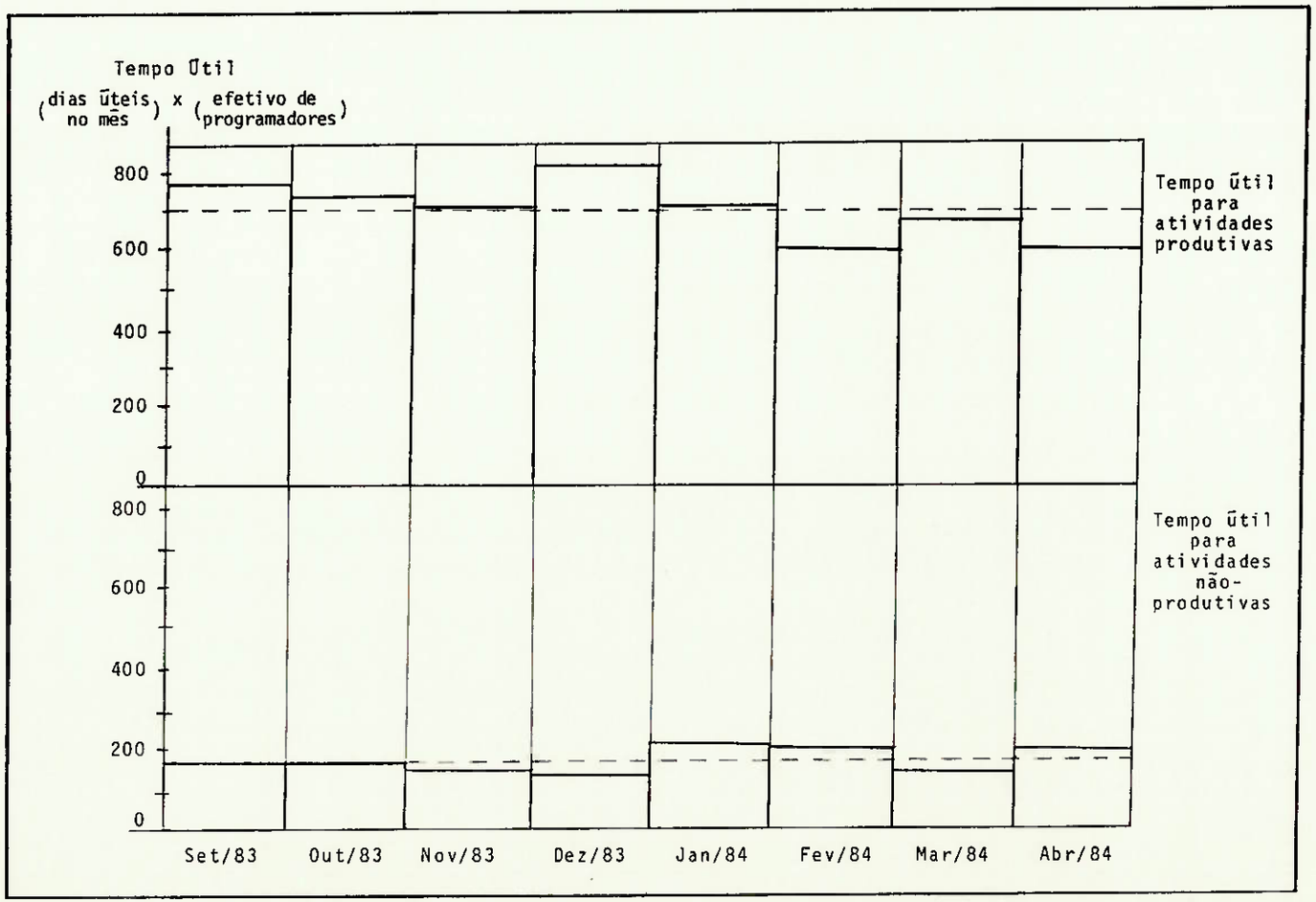


Figura 4.9 - Representação gráfica da distribuição do tempo útil entre atividades produtivas e não produtivas

- e.p.a. -

Todos os dados compilados foram sumarizados e colocados em forma de médias globais mensais na tabela 4.9.

	Média de Recebidos (Prog/mês)	Média de Liberados (Prog/mês)	Média em mãos (Prog/mês)	Média em mãos)/(Média Liberados)
Programa novo	77,4	81,5	113,7	1,40
Manutenção de programa	307,6	322,9	187,9	0,58
TOTAL	385,0	404,4	301,6	0,75

Tabela 4.9 - Dados globais médios referentes ao fluxo de programas pela Programação
- e.p.a. -

4.1.5 - DADOS DOS ATUAIS TEMPOS DE ATENDIMENTO

Colhidos os dados referentes ao número de programas manipulados pela Programação partimos para o levantamento dos tempos envolvidos que é uma grandeza mais importante do ponto de vista da produtividade. O número de programas manipulados, sejam novos ou para manutenção, são mais adequados para se examinar o nível de produção do ambiente e não de sua produtividade.

Na ausência de dados específicos a respeito, identificamos 3 alternativas para o levantamento dos dados que são comentadas a seguir:

A. Coleta de Informações Verbais

Consistiria em levantar amostras dos tempos envolvidos no desenvolvimento de programas novos e execução de manutenções por tipo, além dos tempos envolvidos em outras ativi

dades, com base em informações pessoais a serem fornecidas por programadores e analistas. O processo seria questionável pois ficaria muito sujeito à subjetividade dos informantes e porque os mesmos tenderiam a fornecer apenas dados "médios" assimilados durante a sua experiência no trabalho. A única informação razoavelmente confiável que poderiam fornecer é o tempo que levaram para a execução do(s) último(s) serviço(s) e o tempo dispendido até o momento em que fossem abordados, na atividade em que se encontravam. Mesmo assim, por exemplo, apenas para se coletar uma amostra de cerca de 50 dados de desenvolvimento de programas "batch" de atualização, levar-se-ia mais de 2 meses já que como mostra a tabela 4.4 a média mensal de produção desse tipo de programa tem sido de 23,1.

B. Aplicação de Técnicas de Estudo de Tempos

O Estudo de Tempos (vide ref. 5) envolve interessantes técnicas de Medida do Trabalho como Cronometragem, Amostragem do Trabalho, sistemas pré-determinados de tempos sintéticos e uso do Registrador Automático de Freqüência e de Tempo (aplicando técnicas computacionais). No entanto, estas técnicas estão voltadas à determinação do tempo-padrão para executar uma tarefa específica, auxiliando o pessoal da Engenharia de Métodos, a determinar o método ideal ou o que mais se aproxima do ideal para ser usado na prática e a sua aplicabilidade se limita ao estudo de operações repetitivas (de ciclo curto) e padronizáveis encontradas sobretudo em áreas industriais. As atividades da Programação são notadamente intelectuais e não braçais e cíclicas, são impossíveis de padronizar por completo pois cada programa é um caso à particular, e envolvem tempos muito longos, não da ordem de segundos, mas de dias e até de semanas.

C. Verificação das Informações Fornecidas pela Folha de Acompanhamento do Programador

Todos os programadores preenchem um formulário denominado de Folha de Acompanhamento do Programador e o entregam

ao líder do seu grupo a cada mês. Neste formulário registram todas as atividades executadas e em execução, a classificação dos serviços segundo a codificação impressa no cabeçalho as datas envolvidas e observações complementares. A figura 4.10 mostra um exemplo preenchido desse formulário.

Conversando com os líderes de grupo, a respeito dos critérios utilizados para o preenchimento desses formulários, foi possível levantar uma série de particularidades a serem observadas para uma perfeita interpretação dos mesmos:

- quando se executa serviços que envolvem trabalhos com programas (atividades 01 e 02) o campo "SERVIÇO" é preenchido com o código do programa.

- são considerados como outras atividades (03) todas aquelas referidas no subitem 4.1.3. mais as eventuais faltas e atrasos.

- no campo "DATA ENTRADA" o programador marca a data em que cada serviço passou a ser executado por ele. Portanto, essa data corresponde ao início do atendimento de cada serviço que na maioria dos casos não corresponde à data em que o serviço lhe foi passado. O tempo que esse serviço teve de aguardar em fila não está registrado no formulário.

- no campo "DATA PREVISTA LIBERAÇÃO" é marcada a data acertada com o analista responsável pelo programa para liberar o programa.

- no campo "DATA TÉRMINO" é marcada a data em que o programa foi efetivamente liberado.

Esse formulário tem sido preenchido regularmente pelos programadores desde 1983 e constitui um acervo de dados considerável. Entretanto, apresenta a limitação de não fornecer com a precisão de horas, o tempo efetivamente gasto em cada serviço o que se torna ainda mais delicado, se acrescentar

FOLHA DE ACOMPANHAMENTO DO PROGRAMADOR

MÊS/ANO 04/84

NOME Mario Antonio

CARACTERÍSTICAS DO PROGRAMA

01 - BATCH ATUALIZAÇÃO
02 - BATCH RELATÓRIO
03 - ON-LINE SCOA-CONSULTA
04 - ON-LINE SCOA-ATUALIZAÇÃO
05 - ON-LINE SEM SCOA-CONSULTA
06 - ON-LINE SEM SCOA-ATUALIZAÇÃO
07 - OUTRAS LINGUAGENS

ATIVIDADES

01 - PROGRAMA NOVO
02 - MANUTENÇÃO
03 - OUTRAS

SERVIÇO	CARACT	ATIV.	DATA ENTRADA	DATA PREV LIBERAÇÃO	DATA TÉRMINO	OBSERVAÇÕES
VDU001	01	01	28.03.84	05.04.84	04.04.84	
VDU002	01	01	02.04.84	09.04.84	06.04.84	
VDU201	03	01	04.04.84	06.04.84	06.04.84	
		03	09.04.84		09.04.84	½ período para curso
PCT008	03	02	10.04.84	10.04.84	10.04.84	
PCT001	04	02	10.04.84	11.04.84	11.04.84	
PCT010	04	02	11.04.84	12.04.84	11.04.84	
PCT017	05	02	11.04.84	12.04.84	12.04.84	
PCT012	05	02	12.04.84	13.04.84	13.04.84	
		03				Cheguei às 9:30h
AC1036	02	02	16.04.84	18.04.84	17.04.84	
AC1030	02	02	17.04.84	18.04.84	19.04.84	
VDU009	07	01	18.04.84	25.04.84	26.04.84	
AC1010	02	02	23.04.84	24.04.84	25.04.84	
AC1206	04	02	26.04.84	27.04.84	26.04.84	
		03	27.04.84		27.04.84	Reunião à tarde sobre o sistema "CPT"
AC1045	06	02	27.04.84	30.04.84	27.04.84	
VDU012	03	01	30.04.84	01.05.84		
		05	30.04.84		30.04.84	15:30 h palestra

Figura 4.10 - Exemplo da Folha de Acompanhamento do Programador

mos o fato de que os programadores costumam atender a mais de um serviço por dia, procurando concatenar as etapas de cada atividade.

Examinadas essas alternativas, optamos por utilizar esta última, pois apesar das limitações existentes, trata-se da única viável em termos práticos, e da massa de formulários existente, podemos obter dados suficientes para se alcançar os resultados desejados.

Para se levantar o tempo dispendido em cada serviço, procuramos conhecer melhor os hábitos de trabalho do programador e definimos os seguintes critérios de cálculo:

1) Quanto o programador atende a outras atividades, ou quando atrasa ou falta, verificamos as informações marcadas no campo "OBSERVAÇÕES", pois com base nelas, às vezes, é possível reconhecer o número de horas dispendidas. Por exemplo: os cursos internos dos quais eles participam duram meio período por dia (4,5 horas úteis), faltas acarretam perda de 9 horas úteis por dia, etc.

2) No caso de desenvolvimento de programas novos, execução de manutenção e outras atividades que não fornecem informações quanto ao tempo efetivamente dispendido, adotamos a seguinte forma de cálculo.

   - pelas datas de entrada e término, verificamos o período de atendimento do serviço.

   - com a ajuda de um calendário, identificamos os dias do referido período que foram feriado ou sábado ou domingo.

   - para cada dia útil que sobrou, rateamos igualmente o tempo útil (9 horas por dia útil) entre os serviços que estavam em atendimento. Se num dado dia útil houver um único serviço em atendimento, considerar que o mesmo absorveu as 9 horas úteis. Assim, o tempo efetivo para o aten

dimento de determinado serviço seria dado (de forma aproximada) em horas úteis pela seguinte expressão:

$$\text{(tempo efetivo de atendimento)} = \sum_i \frac{9}{N_i} \text{ horas úteis,}$$

onde:

i = índice que corresponde à cada dia útil de atendimento,

N_i = número de serviços em atendimento no dia útil i.

O critério de se ratear igualmente o tempo entre os serviços em atendimento (quando há mais de um num dia) pode não ser o mais válido, mas mesmo que o mais sensato fosse ponderar esse rateio, as informações que seriam necessárias para isso não estavam ao nosso alcance e uma tentativa de contornar o problema poderia complicar ainda mais a tarefa de coleta de dados, correndo o perigo de se tornar subjetiva ou mesmo inviável.

Os critérios estabelecidos são simples, mas mesmo assim, como a coleta de dados foi feita manualmente, ela foi bastante trabalhosa (foram manuseadas cerca de 300 formulários).

Na prática, como veremos mais adiante, os dados assim coletados mostraram-se bem consistentes. Talvez isso passa ser explicado pelo fato de que quando dividimos o tempo por igual entre as atividades em atendimento, podemos em dados dias estar superestimado o tempo para um dado serviço, mas em outros pode ocorrer o inverso (estimar para menos) e, eventualmente, até um acerto, de modo que se considerarmos o período de atendimento como um todo, as compensações acabem tornando o resultado próximo do real.

Antes de partirmos para a coleta dos tempos efetivos de atendimento, foi necessário esclarecer alguns pontos importantes, de modo a podermos levar adiante um bom trabalho de

Amostragem (vide ref. 6):

- a população na qual estamos interessados são as atividades executadas pela Programação e que estão registradas nos formulários.

- a variável que nos interessa é o tempo efetivo gasto no atendimento dessas atividades.

- a referida população de atividades se divide em estratos, pois na Programação existem programadores com diferentes níveis de experiência (estagiário, "trainee", júnior, pleno, senior). Assim, por exemplo, entre os programas "batch" de atualização, haveria a grosso modo 5 estratos, cada qual correspondendo a uma sub-população de programas desenvolvidos por programadores de uma determinada qualificação. É de se supor que, de estrato para estrato, a variável de interesse apresente um comportamento substancialmente diverso, tendo, entretanto, comportamento razoavelmente homogêneo dentro de cada estrato. Consultando alguns líderes de grupo e a secretária do Setor de Informática e Serviços de Teleprocessamento, constatou-se que a distribuição dos programadores entre os diversos níveis tem sido aproximadamente uniforme (com um leve desequilíbrio para menos no número de estagiários). Com base nesta informação) e assumindo que as atividades estão uniformemente distribuídas entre os vários níveis de programadores, optamos por aplicar a técnica de Amostragem Estratificada Uniforme (sorteio de igual número de elementos em cada estrato). Na realidade, os programas estão estratificados também conforme o setor a que se destina, ou seja, conforme o grupo que o atende (além de outras formas de estratificação (vide ref. 7) como o grau de dificuldade do serviço, características do programa como porte e função etc. Entretanto, esse preciosismo não seria praticável pela falta de informações nos formulários, à demanda ainda maior de trabalho que seria necessário, etc.). Como, no entanto, nem todos os grupos cederam os formulários para a pesquisa (por não autorização ou por falha de documentação) recaímos

no problema da inacessibilidade a toda a população. Surge aqui, portanto, uma distinção a ser feita entre população-objeto e população-amostrada. A primeira é aquela que temos em mente no presente estudo, e a segunda é a parte da primeira que está acessível a nós, ou seja, os serviços atendidos e registrados nos formulários que nos foram cedidos (4 grupos).

Devido a essas restrições, o que se fez foi admitir que os efeitos da estratificação por grupo pudessem ser desconsiderados e, mesclar os formulários dos grupos acessíveis, supondo que as características da variável de interesse, fossem as mesmas nessa população amostrada e na população-objeto.

Definidas então, a população-amostrada, a técnica de amostragem e os critérios para o cálculo da variável de interesse (tempo efetivo de atendimento), partimos para o levantamento das amostras. Logo, se observou que as manutenções duram em média 1 dia útil, que os programas novos exigem muito mais tempo (da ordem de dias úteis) e que o atendimento às outras atividades duravam no máximo 1 dia útil. Optou-se então, por adotar as seguintes amplitudes para as classes de freqüência das distribuições de freqüência:

1) 1/4 de dia útil (2,25 horas úteis) para a distribuição de tempo efetivo de atendimento às outras atividades.

2) 1/4 de dia útil ou 1/2 dia útil (4,5 horas úteis), conforme a precisão com que os dados foram obtidos, para as distribuições de tempo efetivo de atendimento às manutenções.

3) 1 dia útil (9,0 horas úteis) para as distribuições de tempo efetivo de atendimento para o desenvolvimento de programas novos.

As amostras coletadas foram gravadas em arquivos em disco e utilizando-se o pacote SPSS (Statistical Package for the

Social Sciencies), obtivemos a representação gráfica das amostras em forma de histogramas, com os dados agrupados em classes de freqüências segundo as amplitudes adotadas.

Segue-se como exemplo (figuras 4.11 e 4.12) a listagem obtida processando-se os dados da amostra referente ao tempo efetivo de atendimento para o desenvolvimento de programas novos do tipo "batch" de atualização. Os outros resultados desses processamentos constam no Anexo 1 (inclusive para o atendimento de outras atividades). Em cada processamento, aproveitamos os recursos do pacote SPSS, também para obtermos várias estatísticas importantes como a média da amostra.

4.2 - ANÁLISE DA ATUAL SITUAÇÃO

Com base no que foi apresentado nos itens anteriores e principalmente nos dados coletados, procuraremos agora analisar melhor a atual situação. Toda a análise, foi direcionada no sentido de enfocarmos melhor o problema de modo a direcionar, da melhor maneira possível, a continuidade dos estudos.

4.2.1 - COM RELAÇÃO AO NÍVEL DE ATIVIDADE DA PROGRAMAÇÃO

Analisando os dados globais constantes na tabela 4.9 podemos dizer que a médio prazo (o período abrangido pelos dados é de 8 meses), a Programação se comporta, aproximadamente, como um "sistema em regime". Com efeito, temos que em média, a Programação recebeu mensalmente 385 programas e liberou 404,4 programas. A discrepância entre os valores é de apenas 5%, o que nos permite dizer que os fluxos de entrada e de saída praticamente se equivalem. Como o fluxo de saída foi um pouco maior que o de entrada observa-se que tanto os programas novos como as manutenções em mãos nos últimos meses do período situaram-se em níveis abaixo da média (vide figuras 4.7 e 4.8), ou seja, houve um pequeno desafogo na fila de serviços. Observe ainda que, o período terminou com um número de progra

TEMPO PARA DESENVOLVIMENTO DE PROGRAMAS BATCH/ATUALIZACAO

FILE NONAME (CREATION DATE = 11/06/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

CATEGORY LABEL	CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
	9.	6	9.5	9.5	9.5
	18.	10	15.9	15.9	25.4
	27.	9	14.3	14.3	39.7
	36.	9	14.3	14.3	54.0
	45.	9	14.3	14.3	68.3
	54.	8	12.7	12.7	81.0
	63.	4	6.3	6.3	87.3
	72.	2	3.2	3.2	90.5
	81.	2	3.2	3.2	93.7
	90.	1	1.6	1.6	95.2
	99.	1	1.6	1.6	96.8
	103.	1	1.6	1.6	98.4
	117.	1	1.6	1.6	100.0
	TOTAL	63	100.0	100.0	

Figura 4.11 - Distribuição de frequências do tempo para o desenvolvimento de programas "batch" de atualização

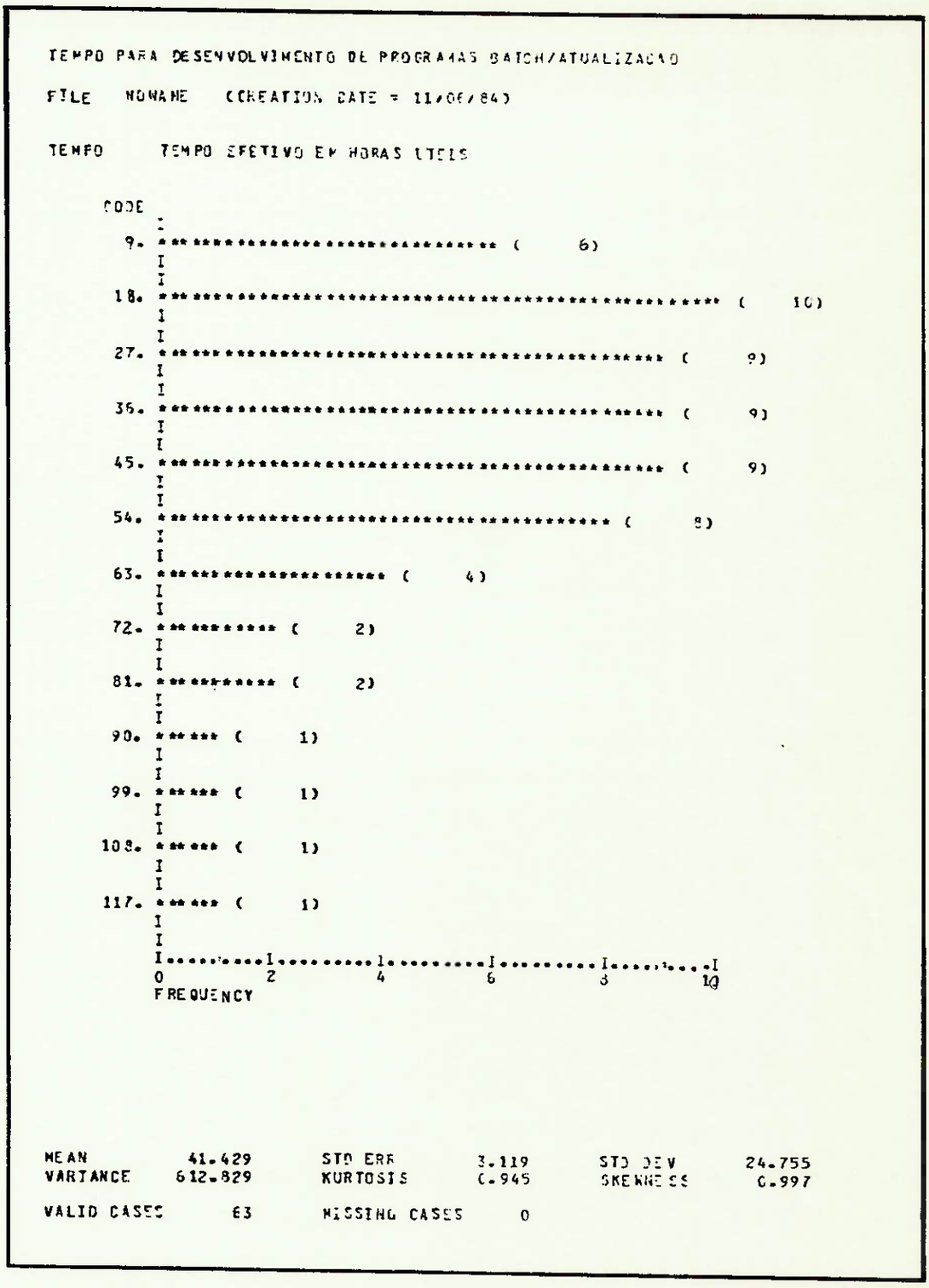


Figura 4.12 - Histograma e estatísticas da distribuição do tempo para o desenvolvimento de programas "batch" de atualização

mas em mãos menor do que no começo. Isso ocorreu porque nos últimos meses a Programação recebeu um número de programas abaixo do número médio de programas liberados.

Outro aspecto importante, é o de que os programadores têm sido forçados a trabalhar sob pressão, pois os analistas têm-lhes passado serviços mesmo que já estejam ocupados no atendimento de outros. Isso se reflete pelo grande número de programas em mãos na Programação que no período foi de 301,6 programas, o que dá uma média de 7,2 por programador (considerando que no período o efetivo foi de cerca de 42 programadores). Assim, mesmo que o programador trabalhe concatenando o atendimento de 2 serviços por vez, ainda sobrariam cerca de 5 programas em fila. Desta forma, os analistas têm conseguido forçar a Programação a trabalhar, praticamente sem ociosidade.

Continuando a analisar as figuras 4.7 e 4.8 verificamos que o número médio de programas recebidos oscila muito em torno de sua média, o que é um fato normal, pois é comum (apesar do planejamento das equipes de análise), de tempos em tempos, ocorrer alívio ou sobrecarga (coincidência no cronograma de diferentes sistemas em desenvolvimento, alguma alteração conjuntural que implique em manutenções em massa, etc.) de serviços na Programação. O número de programas liberados, no entanto, é muito mais estável (oscila menos em torno da média), o que também é normal, pois essa grandeza reflete de certa maneira, a capacidade prática (não é a máxima) de atendimento da Programação, e como vimos, ela tem sido obrigada a trabalhar no limite dessa capacidade. Devido a esta estabilidade no fluxo de saída, o número de programas em mãos acaba também oscilando muito, de modo a absorver a instabilidade do fluxo de entrada.

Assim como qualquer equipamento, acionado em turno(s) pré-determinado(s), apresenta uma capacidade prática de produção (em ton/hora, peça/hora, etc.) abaixo da sua capacidade nominal de placa, a Programação também não deixa de ser um

sistema semelhante que trabalha em regime de turno único de 9 horas diárias e que, na atual configuração (recursos humanos, de "software", de "hardware", etc.) tem conseguido liberar na prática, uma média de 404,4 programas por mês. Isso, naturalmente, é menos do que se conseguiria se, teoricamente, a Programação pudesse trabalhar concatenando perfeitamente as suas atividades, maximizando o aproveitamento do seu tempo útil e das ferramentas disponíveis, e se não houvesse limitação de equipamentos (eventuais esperas por um terminal, fila para processamento, etc.).

A unidade (programa liberado)/(mês), não é na realidade a mais correta para se medir essa capacidade prática de atendimento pois, como vimos, o tempo efetivo necessário para o atendimento dos serviços varia de caso para caso. Se, por exemplo, num dado mês a Programação se concentrasse em atender a, substancialmente, mais manutenções e num outro a mais programas novos, é claro que no primeiro o número de programas liberados seria muito maior que a média, e no outro, muito menor, sem que no entanto, ocorresse qualquer variação na sua capacidade de trabalho.

Apesar dessas restrições, essa unidade continuará a ser adotada pois é de fácil obtenção (praticidade) e porque a médio prazo, acaba sendo uma forma de medida bem representativa.

Convém observar que, quando utilizamos esta medida, estamos, a rigor, fazendo uma alusão à sua capacidade de atendimento à atividades produtivas, sem incluir outras atividades. Estas correspondem a atividades paralelas que embora importantes e necessárias (às vezes até imprescindíveis como apoio a atividades produtivas), não resultam diretamente num produto (programa) e ainda incluem os atrasos e faltas. Desta forma, não faria sentido considerar uma hipotética "capacidade de atendimento à atividades não-produtivas". Neste trabalho, optou-se por adotar o percentual do tempo útil dispendido em outras atividades, como na tabela 4.8, como índice para sua ava

liação. Este índice tem sido de cerca de 20%. Consultando os formulários de acompanhamento do programador acessíveis, obtivemos ainda um outro dado importante: em média, os programadores executam 7.6 outras atividades por mês.

Da tabela 4.9, analisando os valores do quociente (Média em Mãos)/(Média de Liberados) constata-se que em média o número de programas em mãos é 40% superior que o número de liberados para programas novos. No caso das manutenções, ocorre uma tendência inversa, ou seja, o número de programas em mãos é 42% menor em número. Isso é reflexo direto da natureza dos trabalhos da Programação, onde em geral as manutenções têm atendimento mais prioritário.

4.2.2 - COM RELAÇÃO À DISTRIBUIÇÃO DE ATIVIDADES DA PROGRAMAÇÃO

Pelos dados globais da tabela 4.9 percebe-se que o fluxo de programas para manutenção é quase 4 vezes maior que o de programas novos. Isso não implica em dizer, no entanto, que do tempo útil gasto em atividades produtivas, a parcela que cabe às atividades de manutenção seja muito maior. Na prática, como os tempos efetivos para o atendimento de programas novos são também, em média, cerca de 4 vezes maior que os tempos para manutenção, o que tem ocorrido é um certo equilíbrio entre o tempo útil dispendido para ambas as atividades. Mesmo assim, as taxas de manutenção que tem sido verificadas são consideráveis e constituem um motivo de alerta.

No presente trabalho, o interesse maior recai nas atividades de desenvolvimento de programas "batch" de atualização e de relatório, em linguagem COBOL. Das tabelas 4.2 e 4.4 temos que dos programas recebidos e liberados, respectivamente 45,6% e 56,5% correspondem aos referidos tipos (como vimos, a diferença corresponde ao desafogo que houve na fila). Disso conclui-se que, a alternativa que é o objeto de estudo do presente trabalho, terá o seu campo de aplicação limitado a cerca da metade do fluxo de programas novos que passa pela Programação. Evidentemente, o seu campo de aplicação é, na

realidade, mais amplo, pois ainda existem os programas que são desenvolvidos pelos próprios analistas.

De qualquer forma, pelo que foi aqui exposto, levanta-se, nessa abordagem inicial, uma leve suspeita quanto à eficácia da alternativa que foi selecionada e que é o objeto de estudo do presente trabalho. Afinal, se do fluxo de serviços, apenas 20% correspondem a programas novos, e se destes os programas "batch" correspondem à metade, o problema abordado nesse trabalho (produtividade em programação "batch") atinge somente a parcela de 10% dos programas manipulados pela Programação.

As conclusões, porém, serão adiadas para o Capítulo 7, após a apresentação de todo o estudo realizado.

4.2.3 - COM RELAÇÃO DOS TEMPOS EFETIVOS DE ATENDIMENTO

Da forma como os dados de tempo efetivo de atendimento foram calculados (vide critérios no subitem 4.1.5), deve-se observar que os mesmos já incorporam o tempo para todas as tolerâncias necessárias. Os programadores, evidentemente, não passam todo o tempo efetivo de atendimento, completamente absorvidos na execução do serviço . Como qualquer indivíduo, os programadores estão sujeitos a interrupções no seu trabalho. As tolerâncias para essas interrupções no trabalho podem ser classificadas em (vide ref. 5).

a) Tolerância Pessoal: Para a satisfação das necessidades pessoais.

b) Tolerância para a fadiga: Os trabalhos da Programação exigem considerável esforço mental e portanto necessitam de pausas para o relaxamento durante o período de trabalho. Um exemplo típico dessas pausas corresponde às breves paradas para se tomar café que é servido 4 vezes ao dia. Como se trata de uma atividade mental não há nenhum controle sobre as mesmas. Cada programador administra o andamento das suas próprias atividades, sem abusar nos descansos, pois

como vimos, são forçados a trabalhar sob pressão. Além disso, a sala da Programação é ampla e cada um possui uma mesa larga e bem espaçada das outras, ficando todas as mesas de frente para o supervisor, o que dificulta, naturalmente, o desperdício de tempo com comunicações irrelevantes.

c) Tolerância para espera: Os programadores não contam com recursos ilimitados e infalíveis, e nem sempre a concatenação é perfeita quando se procura atender a mais de um serviço por vez. Estão, portanto, sujeitos à esperas inevitáveis como por falta de terminal vago, manutenção em equipamentos ("hardware"), etc.

Esses dados de tempo efetivo de atendimento são muito importantes, pois neste estudo, procuraremos considerar o efeito do aumento de produtividade na atividade de desenvolvimento de programas "batch", pela redução a se conseguir nesses tempos. Existem outras maneiras de se medir produtividade em Programação (vide ref. 7) como o quociente

$$\frac{\text{Número de linhas de codificação}}{\text{Homens/Dia}}$$

ou

$$\frac{\text{Tempo real de desenvolvimento}}{\text{Tempo Estimado}} .$$

Mas ambas apresentam fatores muito subjetivos e pouco fundamentados, tendo sido alvo de muita controvérsia em ambientes de processamento de dados. Além disso, a sua aplicabilidade está mais direcionada a processos de estimativa de prazos ou de controle, o que não condiz com os objetivos do presente trabalho.

A opção adotada, permitirá verificar as repercussões na Programação de um eventual aumento de produtividade no desenvolvimento de programas "batch", partindo-se diretamente

do seu efeito mais imediato que é, exatamente, a redução no tempo efetivo necessário para o atendimento dessas atividades.

Observando os histogramas das amostras coletadas de tempos efetivos de atendimento (figura 4.13 e Anexo I) verificamos que a distribuição densidade de probabilidade desses tempos devem seguir, aproximadamente, o formato da curva ilustrada na figura 4.13.

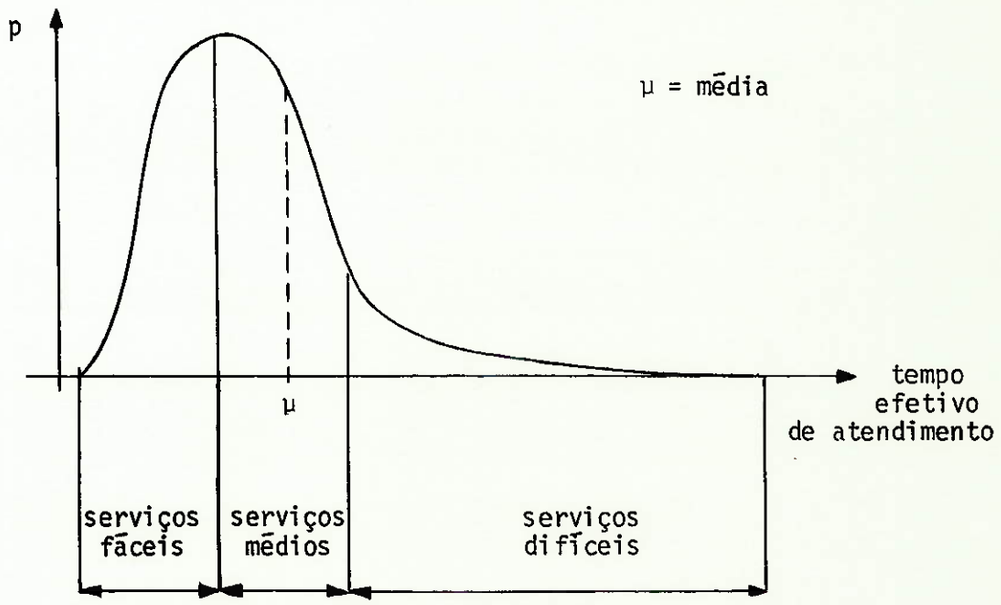


Figura 4.13 - Formato aproximado da curva das distribuições densidade de probabilidade dos tempos efetivos de atendimento - e.p.a. -

As curvas são truncadas em ambos os lados, pois nenhum serviço da Programação pode ser executado em período de tempo nulo, nem se prolongar, por mais difícil que seja, até tempo infinito.

As curvas apresentam este formato pelo fato de que a maioria dos serviços que são enviados à Programação, são de

pequena (rāpido de se atender) ou de mēdia complexidade (cujo atendimento requer um tempo efetivo prōximo da mēdia), sendo que, no entanto, existem alguns poucos que demandam um tempo significativamente maior, forçando o alongamento para à direita.

5. AVALIAÇÃO DO IMPACTO NO AMBIENTE

Este Capítulo é o "coração" do trabalho desenvolvido pelo autor.

Procuraremos neste Capítulo, avaliar o impacto que se verificaria no ambiente, se o Meta-Cobol viesse a ser efetivamente implantado e oferecido como mais uma ferramenta de desenvolvimento ao pessoal da Programação[1].

Não se trata de uma tarefa fácil, pois o sistema a ser estudado (Programação) é muito complexo, cheio de inter relacionamentos, fatores limitantes, fluxos de atividades variadas, critérios imprecisos e tempos não determinísticos. Toda essa complexidade, tem como origem a natureza dos trabalhos executados pela Programação. São trabalhos predominantemente mentais, que exigem muito tempo e que, de forma alguma, são padronizáveis pois cada serviço é único e normalmente não é repetitivo.

As técnicas disponíveis para se estudar tais tipos de sistemas são poucas, e a adoção de uma ou de outra depende de uma série de fatores, tais como, recursos disponíveis, adequação da técnica ao caso abordado, tempo disponível e capacidade e experiência da equipe ou pessoa responsável. Dentre essas técnicas poderíamos citar as seguintes (vide ref. 8).

- experimentação real
- utilização de modelos analógicos
- simulação com computadores
- utilização de modelos da Teoria das Filas
- utilização de outros modelos matemáticos.

---

[1] A ferramenta não se destinaria somente aos programadores, pelo contrário, estaria disponível a todo o pessoal do Setor de Informática e Serviços de Informação, mas neste trabalho estamos interessados especificamente no seu uso dentro da Programação.

Destas, optamo-nos pela aplicação de simulação em computadores. As demais são todas inviáveis e foram excluídas pelos mais diversos motivos: a experimentação real não seria possível, pois o Meta-Cobol ainda não passa de uma idéia; a utilização de modelos analógicos ou da Teoria das Filas também não, pois não existe nenhum aplicável ao nosso caso, e, finalmente, a formulação de outros modelos matemáticos (Ex. abstração por meio de equações diferenciais) foi desconsiderada pelas suas enormes dificuldades conceituais e completa inexperiência do autor em tal técnica.

A simulação é uma das técnicas de análise mais poderosas que está ao alcance das pessoas envolvidas no desenvolvimento (projeto) ou administração (operação) de sistemas de grande complexidade. Trata-se de uma técnica numérica de experimentação sobre um modelo, envolvendo extensos intervalos de tempo (simulado) de operação do sistema que se quer estudar (vide ref. 9). O modelo a ser simulado pode ser representativo de uma situação real ou fictícia (proposta), e pode ser usado com o propósito de simplesmente descrever o comportamento de um dado sistema (melhor entendimento) ou para testar a sua reação sob diversas condições (critérios) diferentes de funcionamento.

Um bom trabalho de simulação requer muita organização e planejamento. No presente trabalho, tomaremos como base a metodologia apresentada por Robert E.Shannon no seu livro "System Simulation - the art and Science" (vide ref. 8), onde sugere que todo o processo seja dividido em estágios sucessivos e progressivos, como é apresentado a seguir:

1) Definição do sistema

2) Formulação do modelo

3) Tratamento dos dados

4) Desenvolvimento de um programa para o modelo

5) Validação do modelo

6) Planejamento das simulações

7) Processamento das simulações

8) Interpretação dos resultados

9) Implementação

10)Documentação.

No presente trabalho o estágio da implementação não foi incluído, pois não faz parte do escopo do trabalho. O estágio da documentação correspondeu ao preparo do presente Capítulo.

Os demais estágios (do 1 ao 8) serão tratados cuidadosamente em itens separados a seguir.

Do estágio 1 ao 4, preocupamo-nos em desenvolver um modelo para a simulação da atual situação, de modo que pudéssemos no estágio 5 calibrá-lo com base nos dados atuais e chegarmos à sua validação.

No estágio 6 planejamos o aproveitamento do modelo desenvolvido para a simulação e avaliação do impacto da introdução do Meta-Cobol no ambiente da Programação. As adaptações em cima do programa testado no estágio 5 que se fizeram necessárias para as simulações planejadas nesse estágio, serão comentadas no respectivo item.

No estágio 7 procedemos o processamento das simulações planejadas e, finalmente, no estágio 8 interpretamos os resultados obtidos, preparando os dados necessários ao Capítulo seguinte, no qual trataremos da viabilidade econômica do desenvolvimento do Meta-Cobol.

5.1. DEFINIÇÃO DO SISTEMA

O sistema cujo comportamento procuraremos simular é o do ambiente da Programação. A Programação é um sistema formado por 3 entidades básicas os quais são descritos a seguir.

A. Programadores

A Programação tem mantido um efetivo girando em torno de 42 programadores desde o final de 1983. Cada programador funciona como um posto de atendimento a qualquer tipo de serviço, ou seja, é "multi-funcional", e, embora possa concatenar o atendimento de dois ou mais serviços ao longo do tempo, num dado instante, naturalmente, está atendendo a somente um deles. Assim, se tirássemos uma fotografia da Programação num dado instante, ela estaria atendendo no máximo, a 42 serviços. Os programadores trabalham em parceria com um ou dois analistas, recebendo serviços de desenvolvimento de programas novos diretamente dos mesmos, seja para aplicação "batch" ou "on-line", seja "complexo" ou "simples". Os trabalhos de manutenção que lhe são passados são normalmente referentes a um sistema de informações em cujo desenvolvimento tomou parte ou com o qual possui familiaridade. Os programadores têm expediente de trabalho rigidamente controlado por cartões de ponto, trabalhando 9 horas úteis por dia (já excluindo horário de almoço) e muito raramente trabalham em horas-extras.

Maiores detalhes a seu respeito podem ser consultados no subitem 4.1.1..

B. Atividades dos Programadores

Essas atividades são todas aquelas com as quais os programadores lidam no seu dia-a-dia de trabalho. Existem basicamente dois tipos de atividades: as produtivas e as não-produtivas. As atividades produtivas são aquelas que correspondem ao atendimento de um serviço propriamente dito, e que são

de dois tipos: serviços de manutenção e de desenvolvimento de programa novo. As atividades não-produtivas são as que são chamadas de "outras atividades" como participação em cursos e palestras, reuniões e atendimento de visitas (vide subitem 4.1.3.) e ainda incluem faltas ou atrasos. Neste trabalho consideraremos ainda um terceiro tipo de atividade que é o cumprimento do direito às férias.

C. Fila de Atividades

Como vimos no Capítulo 2, a existência de programas em fila é fato marcante na Programação. Isso será levado em consideração na formulação do modelo e de forma um pouco mais ampla, extendido ao conceito de fila de atividades.

Todas essas entidades se relacionam, a grosso modo, da seguinte maneira: no decorrer do tempo surge uma série de atividades direcionadas aos programadores, e como no geral, o seu total é maior que a capacidade de atendimento (uma atividade por programador por vez), estas são colocadas em fila conforme a sua ordem de chegada, sendo no entanto, atendidas por um critério de prioridades, conforme o tipo da atividade e ocasião.

O ideal, neste trabalho, seria o de se considerar o sistema da Programação como um todo, mas mantendo um tratamento personalizado para cada programador. No entanto, tal sofisticação, esbarraria nas seguintes dificuldades:

a) Quanto mais o modelo é detalhado, ele se torna mais representativo da realidade (a confiabilidade e precisão de seus resultados aumenta), no entanto, os recursos necessários para a sua elaboração também aumentam assustadoramente como é mostrado na figura 5.1.

b) Uma vez formulado o modelo, é necessário desenvolver um programa que o incorpore para o processamento da simulação. Não se trata de uma tarefa fácil e quanto maior a sofisti

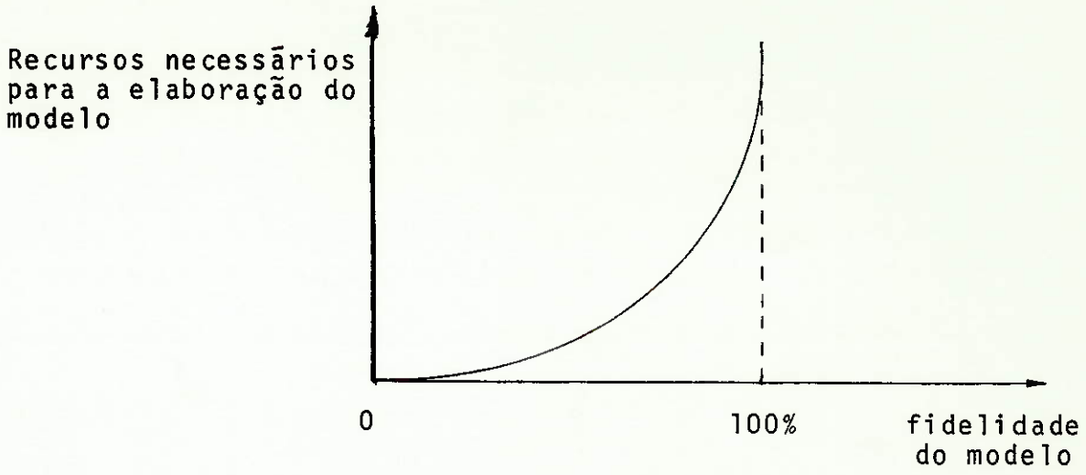


Figura 5.1 - Recursos necessários para a elaboração do modelo em função de sua fidelidade à realidade
- e.p.a. -

cação do modelo, maiores são as dificuldades encontradas. Mesmo que se consiga desenvolver um programa sofisticadíssimo, ainda há o perigo de o sistema em uso, não suportar o seu processamento (quanto maior a sofisticação do modelo, maior é a quantidade de instruções e dados a serem manipulados e, consequentemente, de tempo e memória de computador a ser alocada).

Desta forma, optou-se por tratar o sistema de forma mais simples, mas plenamente viável. Nesse trabalho, consideraremos o sistema a ser simulado como sendo formado por um grupo típico de n programadores e pelas demais entidades já mencionadas. n é dado pelo quociente entre o número total de programadores na Programação e o número de grupos existente (6). O número assim obtido deve ser aproximado a um inteiro, já que não faz sentido falarmos em número de programadores fracionário.

Entretanto, na realidade, os programadores estão divididos em 6 grupos de tamanho variado, portanto, estamos nos referindo a um grupo fictício de tamanho médio, ou seja, ana

lisando de outra forma, a Programação como um todo, mas em escala 6 vezes menor. Evidentemente, isso exigirá posteriormente, que alguns dados globais, também sejam ajustados convenientemente.

Esse grupo fictício de tamanho médio, também será considerado como um grupo com "características médias". Isso porque dentre os grupos da Programação, conforme o setor a que atendem e/ou conforme a época do ano, a distribuição relativa das atividades desempenhadas varia muito (mais manutenções ou mais programas novos, mais programas "on-line" ou mais "batch"), o que é um fenômeno muito difícil de se levar em consideração. Consideraremos para todos os efeitos, que para esse grupo médio, as atividades a serem desenvolvidas em tempo de simulação obedecerão a distribuição relativa média observada para toda a Programação durante o período abrangido pelos dados coletados e apresentados no subitem 4.1.4.. Como as simulações a serem processadas envolverão extensos intervalos de tempo simulado, tal simplificação é perfeitamente aceitável, já que o interesse maior é em relação aos resultados observados a longo prazo, os quais são mais fáceis de serem visualizados e analisados pelos dados globais médios do período.

A figura 5.2 mostra esquematicamente o sistema aqui definido e que será tomado como base ao modelo de simulação.

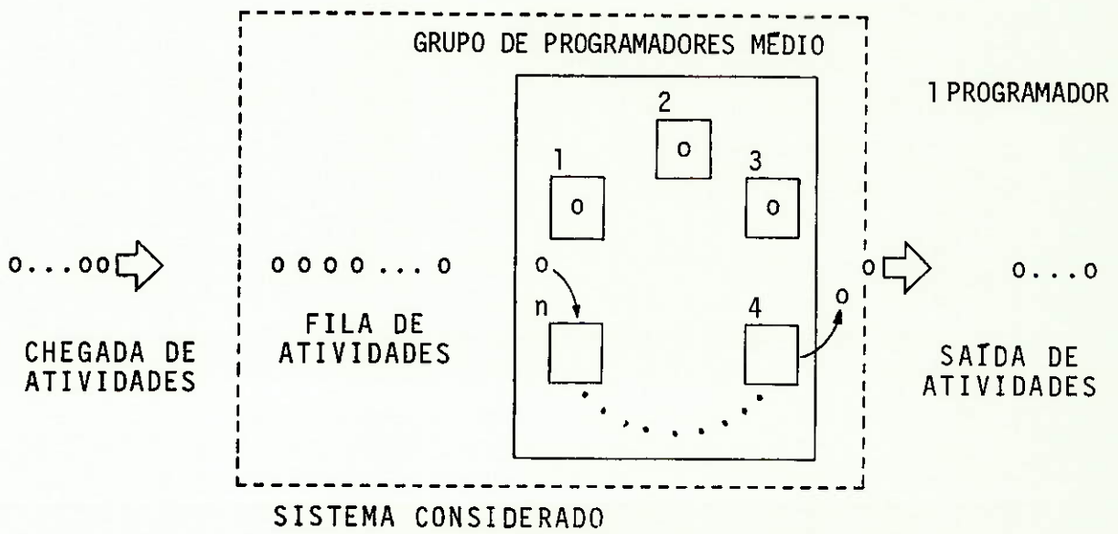


Figura 5.2 - Esquema do sistema a ser considerado na formulação do modelo - e.p.a. -

Note-se que os limites do sistema não incluem as equipes de analistas. O sistema considerado está apenas indiretamente, a elas ligado, por meio da recepção e liberação dos serviços de manutenção e de desenvolvimento de programas novos, que estão incluídas entre as atividades que entram e saem dos limites do sistema.

Pelo menos na atual situação, esse sistema está praticamente em regime (vide subitem 4.2.1.), com os fluxos de entrada e saída de atividades (e de programas) bem equilibrados.

5.2. FORMULAÇÃO DO MODELO

5.2.1. LINGUAGEM ESCOLHIDA PARA SIMULAÇÃO

A formulação do modelo foi executada já se tendo em mente as características, recursos e limitações da linguagem escolhida para o desenvolvimento do programa para simulação. A linguagem que foi adotada é o GPSS (General Purpose Simulation System) da IBM (vide referências 8, 9, 10, 11 e 12, concebida especialmente para estudos de simulação. Existem várias outras linguagens voltadas para este tipo de aplicação como: SIMULA, CSMP, DYNAMO, GASP e SIMSCRIPT, mas optou-se pelo GPSS pelos seguintes motivos:

a) Trata-se de uma linguagem de fácil entendimento e uma das mais utilizadas nessa área.

b) O Setor de Informática e Sistemas de Teleprocessamento dispõe de uma versão (instalada) do mesmo.

c) O autor já o conhecia e possuía razoável experiência em sua utilização, eliminando assim a necessidade de treinamento com uma linguagem desconhecida.

O trabalho de se programar usando o GPSS é relativamente fácil e prático (em comparação) com o uso de outras linguagens), pois este permite que o programa seja desenvolvido com base no fluxo de "transações" que varre o sistema a ser simulado. Para o GPSS, as transações são as entidades móveis que fluem (chegam, entram em fila, são atendidos e vão embora) pelo sistema. No nosso caso, as transações correspondem as atividades a serem desempenhadas pelos programadores (no nosso modelo os programadores serão os postos de atendimento). Assim, qualquer sistema possível de ser representado por meio de um fluxograma de transações (como é o caso da Programação), pode facilmente ser modelado usando o GPSS.

Num fluxograma, as decisões lógicas que devem ser tomadas e os processos ou eventos que ocorrem à medida que as transações fluem pelo sistema, são representados por meio de blocos. O GPSS oferece uma série de comandos específicos que podem ser associados a esses blocos. Cabe, então, ao modelador, montar o programa estruturando esses comandos de modo a representar a lógica e o fluxo observados no sistema que se deseja simular. O programa fonte assim codificado, antes de ser processado é compilado para se tornar executável em computador, e durante o processamento, se encarrega de criar transações, de movê-las através dos blocos especificados executando os processos ou eventos a eles associados e efetuando os sorteios aleatórios sempre que necessário, avançando simultaneamente o tempo simulado (relógio), conforme um mecanismo próprio e transparente ao usuário. A única preocupação que o usuário deve ter com relação ao tempo simulado, é a de se definir todos os dados referentes a tempo, numa mesma unidade.

Definido o aproveitamento do GPSS para o desenvolvimento do programa, podemos antecipar algumas características básicas do modelo:

- tipo estocástico (vide ref. 8).

- mudança de estado discreta (vide ref. 11 e 12).

- método de incremento de tempo variável (vide ref. 11 e 12).

- geração de variáveis aleatórias pelo método da transformada inversa (vide ref. 9, 11 e 12).

5.2.2. GERAÇÃO DE TRANSAÇÕES

As transações quando são geradas pelo programa são todas iguais entre si. Mas vimos que elas representarão todas as atividades desempenhadas pelo programador. Há portanto, a necessidade de se diferenciá-las entre si marcando-as com atributos próprios. Isso é feito por meio da parametrização das transações. Cada transação pode possuir uma série de parâmetros os quais são referenciados por meio de 3 caracteres alfanuméricos. Nesses parâmetros podemos armazenar valores numéricos que tenha para nós algum significado especial (funcionando como código). Nesse trabalho utilizaremos alguns desses parâmetros para particularizar cada transação, dos quais os mais importantes são os seguintes:

a) parâmetro PH1 (Halfword Parameter 1)

Este parâmetro será usado para guardar o código da atividade que a transação deve representar conforme a tabela 5.1.

b) parâmetro PH2 (Halfword Parameter 2)

Este parâmetro conterá o tempo efetivo de atendimento total da transação.

Da tabela 5.1, temos que o modelo lidará com 12 tipos diferentes de atividades e assim optamo-nos por gerar as transações por meio de rotinas de geração específicas para cada tipo. Existe uma maneira mais simples que seria a de se gerar todas as transações através de uma única rotina, diferenciando-as depois, entretanto, o processo escolhido permite

CÓDIGO	DESCRIÇÃO DA ATIVIDADE
7	Férias
9	Outras atividades
10	Desenvolvimento de programas "batch" de atualização novos
11	Manutenção de programas "batch" de atualização
12	Desenvolvimento de programas "batch" de relatório novos
13	Manutenção de programas "batch" de relatório
14	Desenvolvimento de programas "on-line" com SCOA
15	Manutenção de programas "on-line" gerados com SCOA
16	Desenvolvimento de programas "on-line" sem SCOA
17	Manutenção de programas "on-line" programados convencionalmente (sem SCOA)
18	Desenvolvimento de programas novos em outras linguagens que não COBOL
19	Manutenção de programas em outras linguagens que não COBOL

Tabela 5.1 - Códigos para interpretação do parâmetro PH1
- e.p.a. -

que a geração de cada atividade fique independente das demais, facilitando as alterações.

Para se gerar as transações, assumiremos que as chegadas de todos os tipos de atividades, são eventos que ocorrem independentemente a uma taxa constante durante o expediente de trabalho da Programação, de modo que possam ser representa

das por uma distribuição de Poisson. A rigor, isso é uma simplificação da realidade, mas para efeito de simulação (e não de determinação de probabilidades) envolvendo um extenso intervalo de tempo simulado, isso não deixa de ser uma boa aproximação, desde que se respeite a taxa média de chegadas referente ao período.

O ideal seria o de se acompanhar as chegadas de tais atividades durante um certo intervalo para se tentar levantar distribuições mais próximas do real. Isso no entanto, não seria viável pelo excessivo tempo que seria necessário para se tentar levantar uma amostra razoável (vide dados de produção no subitem 4.1.4.) e pelo trabalho de controle que isso exigiria, além de ser inviável sem a compreensão e contribuição dos programadores.

5.2.3. ESTABELECIMENTO DE CONDIÇÕES INICIAIS

O modelo leva em conta que a Programação tem trabalhado com razoável número de programas em mãos. Isso será fundamental para o estabelecimento das condições iniciais de simulação, pois a sua desconsideração implicaria em iniciar o processamento com o sistema em vazio, o que poderia distorcer os resultados do período considerado.

Assim, o modelo terá rotinas que gerarão manutenções de programas e programas novos logo no começo da simulação (tornando-se inativas depois) para estabelecer, aproximadamente, as condições iniciais do período. Essa situação inicial, será apenas aproximada pois as transações serão geradas todas num mesmo instante, e, assim, as primeiras que passarem a ser atendidas pela Programação estarão todas na "estaca zero" neste instante inicial. A rigor, todos os programas deveriam estar em estágios diferentes de atendimento, mas essa consideração complicaria em muito o programa de simulação.

5.2.4. MODELOS DE ATENDIMENTO

Os programadores atendem a cada atividade de maneira diferente, conforme o seu tipo. As atividades de manutenção de programas, cumprimento de férias e as chamadas outras atividades são atendidas ininterruptamente, ou seja, quando um programador começa a executá-las, ele inicia o atendimento e não muda de atividade até o fim do atendimento. Conforme a duração do atendimento, a atividade pode ser interrompida para fins de almoço ou por fim do expediente do trabalho, mas se se considerar somente o tempo útil, o atendimento se passa como se fosse contínuo.

O atendimento dos serviços de desenvolvimento de programas novos é diferente. Isso ocorre porque este tipo de atividade envolve tempos muito mais longos que os serviços de manutenção e as chamadas outras atividades, e porque o desenvolvimento de programas novos, como já foi comentado (vide item 4.1.), é dividido em várias etapas distintas e progressivas (vide figura 4.2), o que motiva o parcelamento do seu atendimento, mediante algumas interrupções. Quando se adota este tipo de atendimento para o desenvolvimento de programas novos (procedimento muito comum em vários outros tipos de sistema , como por exemplo, execução de programas em computadores) ocorre dois efeitos importantes: sacrifica-se o prazo de atendimento desses serviços, mas por outro lado, nessas interrupções momentâneas consegue-se atender a outras atividades mais rápidas ou adiantar o desenvolvimento de um outro programa novo. Esse segundo efeito é muito importante, pois ajuda a desafogar a fila da Programação e, consequentemente, diminuir o tempo médio em fila.

Essa prática da concatenação de atividades é mostrado pela figura 5.3 que apresenta uma espécie de Gráfico de Ganth das atividades desempenhadas por um programador que começa a desenvolver um programa novo. Observe que o desenvolvimento inicial de um programa novo, que corresponde às etapas de estudo e codificação do programa (vide figura 4.2), é

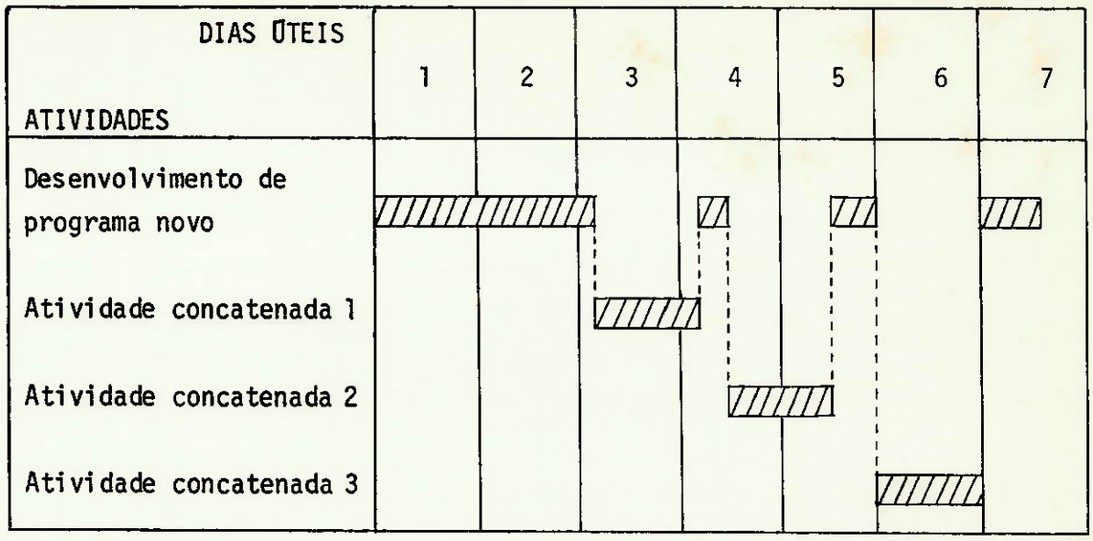


Figura 5.3 - Exemplo de concatenação de atividades por um programador - e.p.a. -

executado sem interrupção de modo a não quebrar a concentração e o raciocínio do programador. Na figura 5.3 isto começa no dia 1 e termina no começo do dia 3. Se o programa foi desenvolvido convencionalmente pelo programador, a interrupção entre os dias 3 e 4 deve ser interpretada como sendo devida à necessidade de se enviar o programa para à Digitação. As demais interrupções são devido a uma norma interna que obriga que cada programa fonte novo, seja compilado ou processado para testes, uma única vez ao dia, de modo a se evitar processamentos desnecessários ou precipitados (prejudicando a fila e sobrecarregando o computador) e forçar o parcelamento das etapas de depuração e testes do programa.

O modelo leva em consideração essa necessidade de concatenação, de uma forma mais simples, pois a sua perfeita representação é praticamente impossível dado a diversidade dessas situações e a inexistência de dados a respeito.

Com base na orientação dada pelo supervisor da Equipe de Pacotes Padronizados e com as informações colhidas em

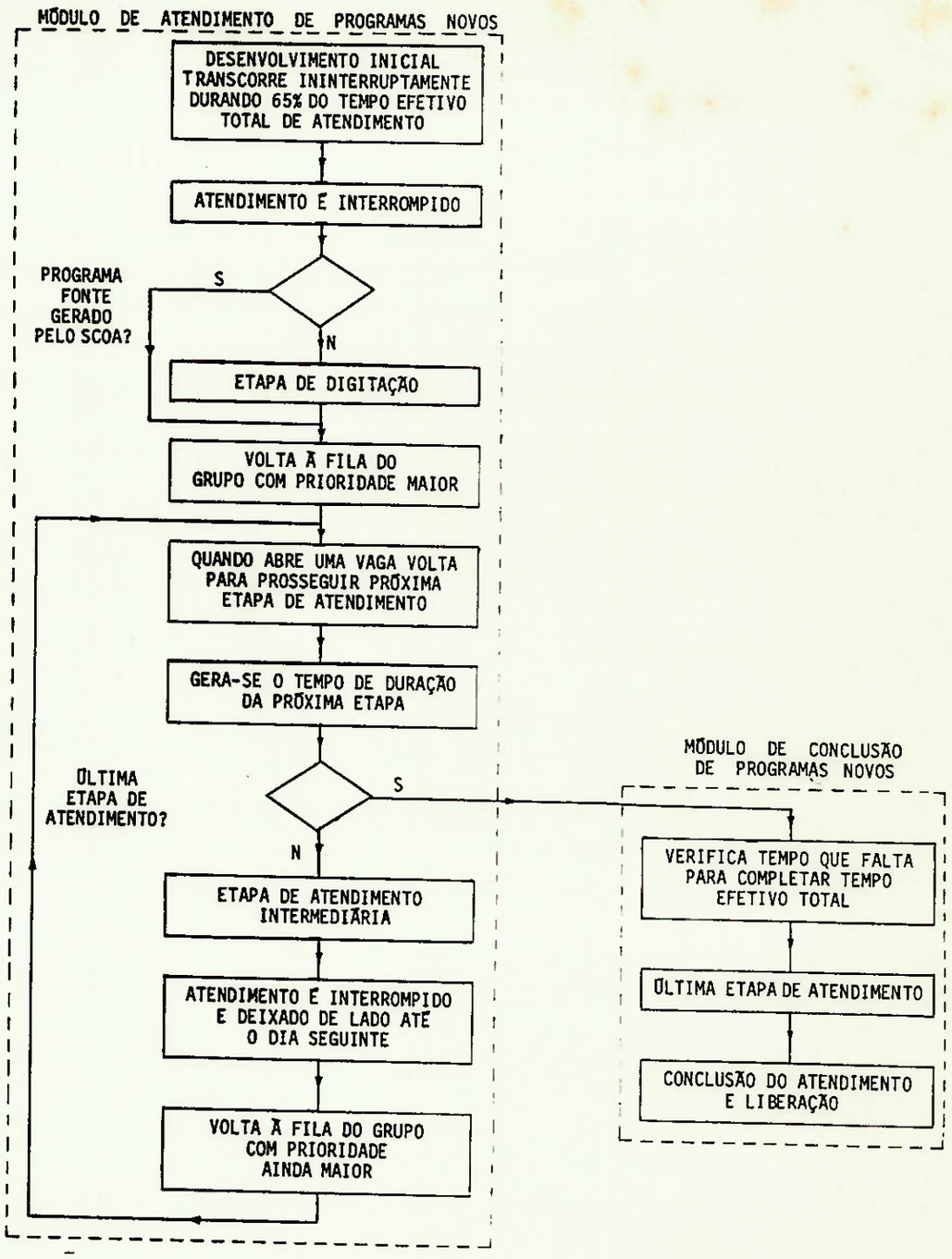


Figura 5.4 - Modelo de atendimento das atividades de desenvolvimento de programa novo

- e.p.a. -

consultas a programadores e analistas, elaboramos o modelo de atendimento das atividades de desenvolvimento de programas novos que é apresentado na figura 5.4.

É um modelo simplificado, mas que representa razoavelmente o que se passa realmente na Programação e relativamente fácil de se codificar em GPSS. O coeficiente de 65% do tempo efetivo total, que é usado na determinação do tempo a ser dispendido na etapa de desenvolvimento inicial de programas novos é o valor médio das estimativas apresentadas pelas pessoas consultadas. Observe-se que o modelo prevê a possibilidade de se desenvolver programas novos para aplicações "on-line" utilizando-se o SCOA (o que dispensaria a etapa de digitação). O mesmo, no entanto, não é feito com relação ao Gerador de Relatórios, já que se trata de uma ferramenta com uso praticamente limitado entre os analistas.

Para se utilizar esse modelo são necessários os seguintes dados de tempo:

a) tempo efetivo de atendimento total;

b) tempo necessário para a digitação do programa;

c) tempo necessário para o atendimento de uma etapa intermediária;

d) tempo de espera entre o fim de uma etapa intermediária e o começo do dia seguinte;

e) tempo da última etapa de atendimento.

Destes somente os 3 primeiros necessitam a especificação de uma distribuição de probabilidade e serão tratados no item 5.3.. O tempo dado em d é facilmente calculável consultando-se somente o valor do relógio simulado. O tempo dado em e é obtido pela diferença entre o tempo efetivo de atendimento total e o tempo efetivo de atendimento acumulado até a penúltima etapa (inclusive). No tempo efetivo de atendimento acumulado, não se inclui o tempo necessário para a digitação.

É necessário ainda considerar que existe a possibilidade de um serviço de manutenção ou de desenvolvimento de programa novo não ser executado até o seu fim, ou seja, ser liberado sem, no entanto, ter sido concluído. Isto ocorre com cerca de 5% dos programas (médias das estimativas apresentadas pelas pessoas consultadas) e pode ser provocado por alguma reconfiguração de projeto, especificação, falha, rearranjo de cronograma (perda de prioridade), reconsideração do analista responsável, etc. O programador percebe a impossibilidade de sua execução ou recebe instruções para abandonar o programa, ainda no começo do seu atendimento. O tempo que o programador levaria nesse atendimento incompleto é tratado no item 5.3..

A figura 5.5 apresenta o modelo global de atendimento de atividades. Observe-se que o modelo de atendimento de programas novos apresentado na figura 5.4 (caso particular), faz parte integrante do mesmo.

5.2.5. FILAS E CRITÉRIOS DE PRIORIDADE

Vimos no item 5.1. que o sistema a ser simulado, seria uma espécie de "Programação em miniatura", sendo representado por um grupo médio (quantitativa e qualitativamente) de programação.

Ficou estabelecido que no respectivo modelo, esse sistema seria tratado ao nível de grupo como um todo, devido à inviabilidade de se considerar o tratamento personalizado de cada programador.

Desta forma, optamo-nos por modelar o relacionamento entre o grupo médio de programação e a respectiva fila de atividades, apoiando-se nos seguintes recursos oferecidos pelo GPSS:

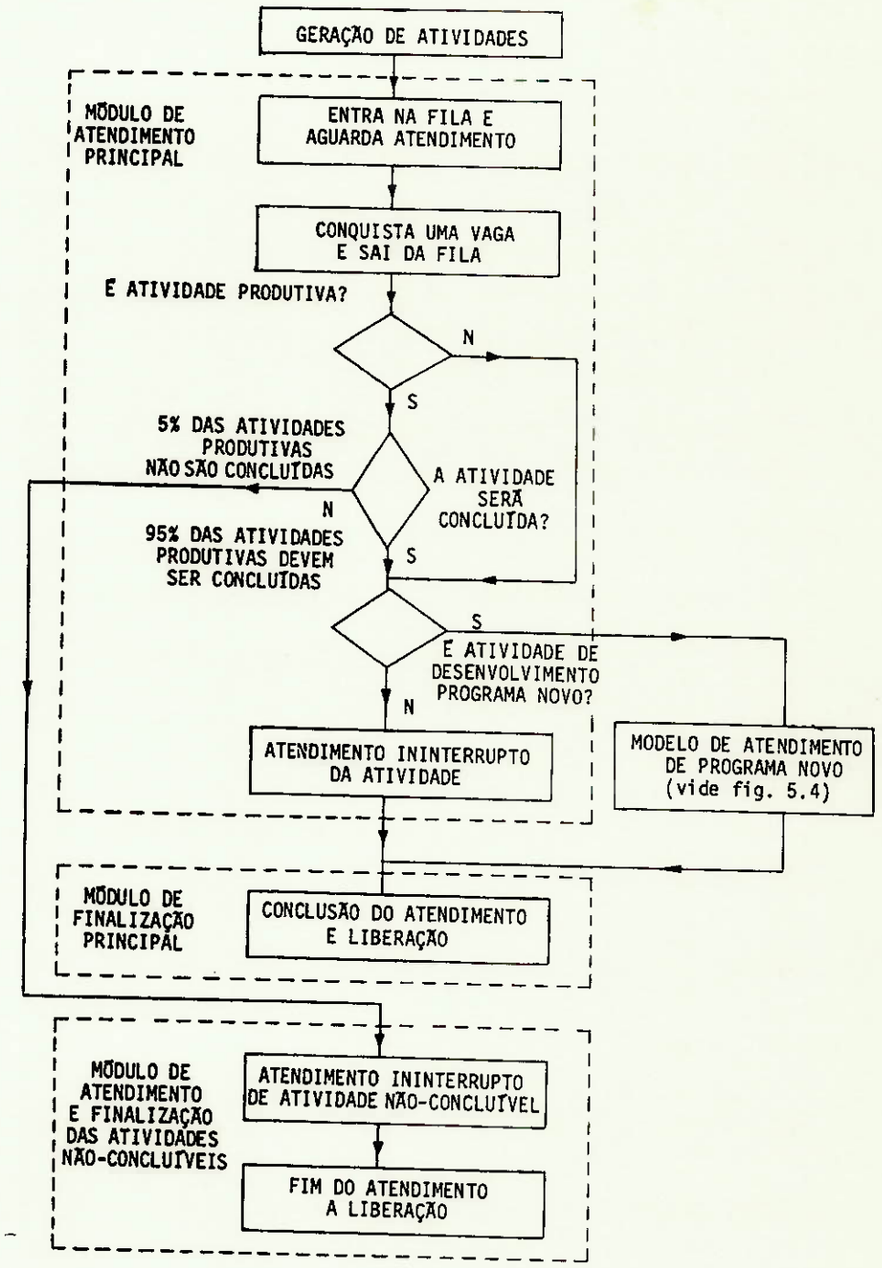


Figura 5.5 - Modelo global de atendimento de atividades - e.p.a. -

A. Depósito de Transações (Storage)

O depósito de transações representa uma entidade com capacidade de armazenar múltiplas transações. Definindo-se um depósito e especificando-lhe uma capacidade máxima, o GPSS se encarrega de manter no seu interior, um número de transações menor ou igual a essa capacidade, ou seja, durante a simulação, a capacidade de um depósito jamais é ultrapassada. Num depósito propriamente dito, o material armazenado fica geralmente estático (estoques). Mas para o GPSS, as entidades que se encontram no interior de um depósito, no caso as transações, são dinâmicas. Assim, aos olhos do modelador, tudo se passa como se os depósitos fossem "postos de atendimento com capacidade maior ou igual a 1". Qualquer transação que entra no seu interior, lá permanece pelo tempo que lhe foi gerado, partindo imediatamente em seguida (fim do atendimento ou de uma etapa do atendimento) e, consequentemente, abre-se uma vaga para uma outra transação.

Como no modelo as transações são consideradas como as atividades desempenhadas pelos programadores, o conceito de depósito de transações se ajusta muito bem ao grupo médio de programação. Assim sendo, no modelo, esse grupo será associado a um depósito com capacidade igual ao seu número de programadores.

B. Fila (Queue)

É necessário considerar a possibilidade de se formar fila junto à entrada do depósito que irá representar o grupo médio de programação. Isso será feito, mas considerando-a como única para o grupo como um todo.

Essa simplificação, além de viabilizar o desenvolvimento de um programa em GPSS não distorce a realidade, pois na Programação todos trabalham sob pressão devido ao fato de as equipes de análise recorrerem ao artifício de enviar mais serviços aos programadores, mesmo que estes já estejam aten

dendo a outros, de modo a manter um número razoável de programas em mãos dos mesmos e evitar ociosidade. No sistema real, cada programador controla a sua própria fila. Entretanto, como todos são exigidos da mesma forma, ninguém fica mais sobrecarregado ou mais aliviado que os outros e no período englobado na coleta de dados mantiveram uma média de 7 programas em mãos (vide subitem 4.2.1.) ao longo do tempo. Mesmo que, momentaneamente, alguém fique menos sobrecarregado que os demais, nada impede que ele ajude os seus companheiros ou que receba serviços de outros analistas aos quais não esteja diretamente alocado.

Desta forma, é mais do que razoável definir uma fila única para o grupo médio de programação, de modo que as atividades aguardem pela liberação de qualquer vaga no grupo sem se preocupar se corresponde a este ou àquele programador. Outra vantagem de tal simplificação é a possibilidade de se obter, mais facilmente, dados globais referentes à simulação que são mais adequados que os dados muito minuciosos, para fins de análise e interpretação dos resultados.

No modelo consideraremos a formação de duas filas. Uma fila, a que chamaremos de principal, corresponderá àquela que se forma devido às chegadas (gerações) de atividades(transações) na Programação (depósito), quando a mesma já está totalmente ocupada (fila de espera de transações ainda não atendidas). A outra é a formada pelas atividades de desenvolvimento de programas novos que já tiveram o seu atendimento iniciado, mas que tiveram de ser deixados de lado de um dia para outro, devido à uma interrupção. Essa diferenciação é necessária, já que o tratamento dado às atividades que tiveram o seu atendimento interrompido e voltam novamente à fila é distinto ao prestado àquelas que estão na fila principal e não começaram a ser atendidas. Em geral, as atividades interrompidas que aguardam a continuação do atendimento, tornam-se mais prioritárias, pois como os programadores estão autorizados a compilar ou processar (testar) um programa novo somente uma vez ao dia, todos os dias eles procuram retomar o seu desenvolvimento antes de iniciar outra atividade. A figura 5.6

mostra a formação dessas duas filas junto ao grupo a ser simulado.

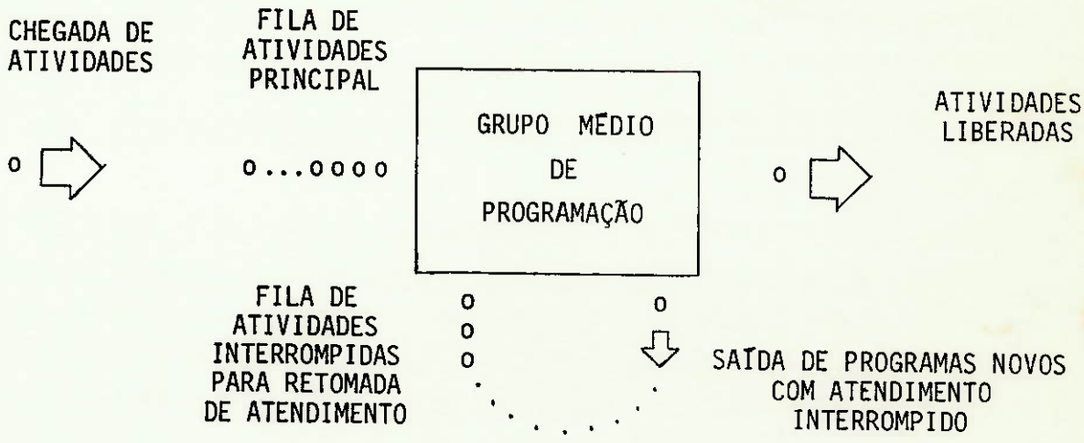


Figura 5.6 - Filas a serem tratadas pelo grupo médio de programação - e.p.a. -

Convém ainda observar outra diferença fundamental entre essas duas filas. A que é formada pelas atividades interrompidas contém somente desenvolvimentos de programas novos, e a fila principal contém todos os tipos de atividades. Os tipos de atividades também necessitam de tratamento diferenciado entre si, pois são de natureza completamente distinta. Observe-se que a questão dos critérios de atendimento de atividades é complexa. Isso será abordado a seguir.

C. Critérios de Prioridade

O problema da necessidade de se estabelecer critérios de atendimento para as atividades nas duas filas será tratado, mediante atribuição de uma prioridade adequada para cada tipo de atividade. Faremos isso, aproveitando o recurso que o GPSS oferece de diferenciar as transações (atividades) também por meio de atribuição de prioridade (outro modo de diferenciação, o qual também adotamos, é via parametrização, vi

de subitem 5.2.2.). Cada transação pode ter prioridade de 1 a 127 (sendo 127 o valor máximo).

Apresentamos a seguir, a prioridade com que cada tipo de atividade será tratada e as devidas justificativas:

a) Cumprimento de férias

Os programadores sempre procuram gozar do direito às férias anuais, dificilmente preferem continuar trabalhando e recebê-las em dinheiro.

No modelo, a geração de uma transação que represente o cumprimento de férias deve ser interpretada como a chegada do momento previsto para o seu início, e não do momento em que programador resolve marcá-lo (com antecedência). Devemos assim garantir que, se o modelo for especificado para gerar no máximo n (número de programadores no grupo) transações de cumprimento de férias por ano, que as tais transações geradas durante a simulação, sejam efetivamente atendidas (ou ser pelo menos recebidas) pelo grupo no período (ano) em que foram geradas. Isso será feito fixando-se uma prioridade muito alta para o cumprimento de férias. Neste modelo adotaremos prioridade 100.

b) Outras atividades

As chamadas outras atividades, reunem diversas atividades que não têm caráter produtivo e as eventuais faltas ou atrasos, que no total, têm ocupado cerca de 20% do tempo útil da Programação (vide subitem 4.2.1.). As faltas ou atrasos muitas vezes não são previsíveis nem planejáveis. Outras atividades como reuniões marcadas na última hora (maioria das casas), assistência ou orientação a outras pessoas e atendimento de visitas, também, no geral, surgem repentinamente e requerem atendimento com urgência. Já as outras atividades como participação em cursos ou palestras, são marcadas com alguma antecedência e no(s) dia(s) de sua realização o seu horário é respeitado com certo rigor pelos programadores. Existem

ainda alguns tipos de outras atividades que não requerem tanto rigor no cumprimento do horário ou que não merecem muita urgência. De qualquer forma, é evidente que as transações que representam essas outras atividades, requerem uma prioridade relativamente alta, e, portanto, foi-lhes atribuída a prioridade 50. O ideal seria considerar que alguns tipos dessas outras atividades, pudessem contar com atendimento "preemptivo" (forçar que o grupo retire de atendimento alguma atividade de menor prioridade e abrir vaga para uma atividade mais urgente), mas trata-se de uma sofisticação não permitida pelo GPSS na manipulação de depósitos de transações.

c) Manutenção de programa

Essas atividades são classificadas de serviços produtivos, pois envolvem diretamente um produto concreto: um programa enviado para manutenção.

As atividades de manutenção e de desenvolvimento de programas novos são as únicas que existem fisicamente na realidade, as quais são facilmente visualizáveis olhando-se as pilhas de programas à espera de atendimento que se amontoam nas mesas da Programação.

Em comparação com as atividades comentadas em a e b que, em geral, chegam em cima da hora e com urgência ou com antecedência, mas com dia e horário já marcados, estas atividades podem ser tratadas com prioridade muito menor. Como as manutenções envolvem Sistemas de Informação já em operação, comumente são tratados com maior prioridade que as atividades de desenvolvimento de programas novos. Nesse modelo, consideraremos que um percentual p% das manutenções receberão prioridade 5 (maior que o das atividades de desenvolvimento de programa novo) e que as demais ficarão com prioridade 1 (mais baixa). Esse valor p é desconhecido e muito abstrato. Sabemos que não é zero nem 100, mas a sua estimação é realmente muito difícil. Optamo-nos por achar um valor adequado experimental

mente, durante a etapa de validação do modelo (item 5.5.) na qual este será "calibrado".

Esperamos com esse artifício, fazer com que o modelo reproduza também o interessante fenômeno descrito no item 4.2.. Lá comentamos que a média mensal de programas em mãos é 40% superior que a média mensal de programas liberados, no caso de desenvolvimento de programas novos; e 42% inferior no caso de manutenções. Isso é reflexo de prioridade normalmente maior dada às manutenções.

d) Desenvolvimento de programa novo com atendimento ainda não iniciado

Como vimos, enquadram-se nesta categoria as atividades de desenvolvimento de programas que se encontram na fila principal do modelo.

Estas são as atividades que, relativamente, têm menos urgência no dia-a-dia da Programação e portanto, atribuiremo-lhes a prioridade mínima 1. Observe-se que não serão as únicas com tal prioridade. Pelo que vimos em c , (1-p)% das atividades de manutenção também terão prioridade 1.

e) Desenvolvimento de programa novo com atendimento interrompido

Como vimos, o desenvolvimento inicial de um programa novo é feito de maneira ininterrupta e o restante é parcelado. Um programa novo, enquanto na fila principal, tem baixa prioridade, mas a partir do momento em que passa a ser atendido, a sua prioridade aumenta drasticamente. Isso ocorre porque na realidade, como o programador é obrigado a aplicar interrupções o que ele faz, para evitar que o prazo de seu desenvolvimento total se estique muito, é procurar no dia seguinte de cada interrupção, dedicar-lhe prioridade muito maior de modo a retomar o seu atendimento o quanto antes, e somente

após outra ou a última interrupção (conclusão), atender alguma outra atividade.

No modelo estas atividades iniciam o atendimento com prioridade 1, mas logo depois do fim da etapa de desenvolvimento inicial, receberão prioridade 60. Observe-se que se trata de uma prioridade maior que a dada às chamadas "outras atividades", para que para efeito do modelo, as respectivas transações recuperem lugar no depósito (grupo) o mais breve possível de modo a forçar que cada etapa de atendimento parcelado, ocorra progressivamente em dias sucessivos (vide figura 5.3).

Há ainda um outro fator a considerar. Um programa novo que já está quase no fim do seu desenvolvimento, é mais prioritário que um programa novo cujo atendimento acaba de ser interrompido pela primeira vez. Assim sendo, à cada interrupção que o programa novo sofrer após o seu desenvolvimento inicial, a sua prioridade será acrescida de 1 (61, 62, 63,...)de modo a se diferenciar de outros que estejam em estágios diferentes de atendimento.

A tabela 5.2 resume os critérios de prioridade apresentados e que serão embutidos no modelo.

Estabelecida a prioridade de cada transação, o que o GPSS faz (durante a simulação) sempre que se abre uma vaga no depósito (grupo), é procurar na fila aquela com maior prioridade e dar-lhe a vaga. Embora as duas filas consideradas no modelo sejam lineares e se formem à medida que as transações vão chegando e se acumulando, na realidade o GPSS utiliza critério de atendimento por prioridade e para as transações com mesmo grau de prioridade, ele aplica o critério "Primeiro a Entrar Primeiro a Sair" (First In First Out - FIFO).

Observe-se que todas as considerações tratadas no presente subitem, formam um complexo mecanismo de manipulação e atendimento de transações (atividades), e que se aproxima em muito da realidade. Isso acontece para que o modelo seja o mais fiel possível à realidade.

CÓDIGO DE ATIVIDADE	ATIVIDADE	PRIORIDADE
7	Cumprimento de férias	100
10/12/14/ 16/18	Desenvolvimento de programa novo em fase de depuração ou de testes	60 + (n-1)
10/12/14/ 16/18	Desenvolvimento de programa novo já atendido até a fase de codificação	60
9	Outras atividades	50
11/13/15/ 17/19	p% das manutenções de programas	5
11/13/15/ 17/19	(1-p)% das manutenções de programas	1
10/12/14/ 16/18	Desenvolvimento de programa novo ainda a se iniciar	1

Tabela 5.2 - Critérios de prioridade
(n representa o número de interrupções já efetivadas)
- e.p.a. -

5.3. TRATAMENTO DOS DADOS

Neste item trataremos os dados com os quais alimentaremos o modelo para se realizar a simulação.

Basicamente, os dados se dividem em 4 categorias conforme a sua aplicação:

a) Dados para o estabelecimento das condições iniciais;

b) Dados para a geração de transações;

c) Dados de tempos de atendimento;

d) Dados complementares.

Nos subitens seguintes trataremos os dados coletados para a simulação da situação atual para fins de calibração do modelo (item 5.5.). Para se processar as simulações necessárias para a avaliação do impacto que se verificaria, caso se introduzisse o Meta-Cobol no ambiente da Programação, o modelo precisa ser realimentado com outros dados. Isso será tratado mais adiante.

Observe-se que como o modelo corresponde ao sistema da Programação em escala reduzida (6 vezes menor) ou a um suposto grupo médio de programação, alguns dados terão de ser devidamente ajustados.

Os dados coletados e apresentados nos subitens 4.1.4. e 4.1.5. referem-se ao período entre setembro de 1983 e abril de 1984, mas assumiremos que "as médias globais obtidas a partir dos mesmos, sejam válidos para todo o ano de 1984", já que as condições reinantes naquele período mantiveram-se estáveis no decorrer deste ano (até novembro de 1984 quando do fechamento deste trabalho).

5.3.1. DADOS PARA O ESTABELECIMENTO DAS CONDIÇÕES INICIAIS

A simulação deve iniciar com alguns programas já em mãos do grupo desde o instante inicial. Por simplificação consideraremos que nenhum programador do grupo estará em férias ou atendendo a uma outra atividade no instante inicial.

Supondo que seja razoável tomarmos os números médios por grupo dados pelas tabelas 4.6 e 4.7 para se iniciar a simulação temos que:

$$\left(\begin{array}{c}\text{número de programas em}\\ \text{mãos para manutenção}\end{array}\right) = \frac{113,7}{6} \cong 19$$

$$\left(\begin{array}{c}\text{número de programas em}\\ \text{mãos para desenvolvimento}\end{array}\right) = \frac{187,9}{6} \cong 31$$

Para se determinar a composição destes programas, o modelo utilizará a distribuição relativa dos programas em mãos dadas na última coluna das tabelas 4.6 e 4.7. Na ocasião da simulação, o GPSS gerará as respectivas transações e determinará se representa este ou aquele tipo de programa, randomicamente, com base em tais distribuições.

5.3.2. DADOS PARA A GERAÇÃO DE TRANSAÇÕES

A geração de transações representa as chegadas de atividades, segundo uma distribuição de Poisson. Seja λ_i a média da distribuição de chegadas para a atividade i, então o tempo entre chegadas distribui-se exponencialmente com média θ_i, onde:

$$\text{tempo médio entre chegadas} = \theta_i = \frac{1}{\lambda_i} .$$

Adotando-se como base um mês de 20,3 dias úteis(mês médio segundo calendário do Setor de Informática e Serviços de Teleprocessamento) e lembrando que o expediente de trabalho é de 9 horas úteis, temos que:

mês médio = 20,3 dias úteis $\cong$ 183 horas úteis/mês.

Então, lembrando que a Programação conta com 42 programadores, o que dá um grupo médio de 7 elementos, levantamos os valores de tempo médio entre gerações (em horas úteis) para cada tipo de atividade conforme mostra a tabela 5.3.

O número médio de chegadas das chamadas outras atividades foi estimado com base em informações obtidas, consultando-se as Folhas de Acompanhamento do Programador (vide exemplo na figura 4.10).

CÓDIGO	ATIVIDADE		MÉDIA DE CHEGADAS POR MÊS ($\frac{\text{atividades}}{\text{mês}}$)	MÉDIA DE CHEGADAS POR HORA ÚTIL ($\frac{\text{atividades}}{\text{hora útil}}$)	TEMPO MÉDIO ENTRE CHEGADAS (hora útil)
7		Cumprimento de férias	0,58	0,00319	313,7
9		Outras atividades	53,20	0,29071	3,4
10	DESENVOLVIMENTO	Programa "batch" de atualização	3,38	0,01849	54,1
12	DESENVOLVIMENTO	Programa "batch" de relatório	2,50	0,01366	73,2
14	DESENVOLVIMENTO	Programa "on-line" com SCOA	3,50	0,01913	52,3
16	DESENVOLVIMENTO	Programa "on-line" sem SCOA	2,90	0,01585	63,1
18	DESENVOLVIMENTO	Programa em outra linguagem (não COBOL)	0,62	0,00337	296,8
11	MANUTENÇÃO	Programa "batch" de atualização	9,28	0,05073	19,7
13	MANUTENÇÃO	Programa "batch" de relatório	11,82	0,06457	15,5
15	MANUTENÇÃO	Programa "on-line" com SCOA	5,62	0,03069	32,6
17	MANUTENÇÃO	Programa "on-line" sem SCOA	22,33	0,12204	8,2
19	MANUTENÇÃO	Programa em outra linguagem (não COBOL)	2,22	0,01211	82,4

Tabela 5.3 - Tempo médio entre chegadas por tipo de atividade para o ano de 1984
- e.p.a. -

Os dados a respeito das chegadas de atividades com programas (serviços produtivos) foram obtidas das tabelas 4.2 e 4.3.

5.3.3. DADOS DE TEMPOS DE ATENDIMENTO

Vimos no subitem 5.2.4. que o modelo de atendimento desenvolvido precisará de uma série de dados relativos aos tempos envolvidos. Como o modelo é estocástico, será necessário alimentá-lo com a especificação da distribuição desses tempos. Durante o processamento da simulação, o GPSS gerará randomicamente valores de tempos com base em tais distribuições pelo método da transformada inversa (vide ref. 9), portanto, estamos interessados na especificação dessas distribuições.

A. Distribuição dos Tempos Efetivos de Atendimento às Atividades Produtivas e Outras Atividades

A coleta de tais dados foi comentada no subitem 4.1.5., e as listagens obtidas processando-se o tratamento estatístico das amostras levantadas por meio do pacote SPSS (Statistical Package for the Social Sciences) estão organizadas no Anexo I.

Para fins de simulação, seria interessante verificarmos se os dados das amostras levantadas se adequam, com bom nível de significância, a um modelo de distribuição de probabilidade. Tal procedimento é denominado de Teste de Aderência e é realizado com base em um dos seguintes métodos (vide ref. 6): Teste de Aderência pelo qui-quadrado, Método de Kolmogorov Smirnov e verificação gráfica com o uso de folhas adequadas como o "papel de probabilidade normal" (verificação da aderência a uma distribuição normal) e "papel semilogarítmico" (verificação da aderência a uma distribuição exponencial).

Neste trabalho utilizaremos o método apresentado por Hahn e Shapiro no livro "Statistical Models in Engineering" (vide ref. 13).

Normalmente, não é fácil decidir contra qual tipo de modelo as amostras serão testadas. Muitas vezes, mesmo escolhendo um modelo cuja forma da distribuição é parecida com a da amostra (histograma), após a realização de um teste de aderência, chega-se à conclusão de que na realidade é inadequada para representar a distribuição da população. Por outro lado, existe ainda a possibilidade de num dado nível de significância, encontrarmos dois ou até três modelos distintos, que se adequam ao caso. Para se evitar este tipo de problema, Hahn e Shapiro sugerem o uso de uma carta elaborada por E.S.Pearson, que, com base no par de valores dadas pelas estatísticas β_1 e β_2 da distribuição da população a ser examinada, dá uma boa idéia de qual distribuição empírica contínua, ela mais se aproxima ou a completa impossibilidade de aderir aos níveis de significância usuais, a qualquer uma das mesmas. Neste último caso, o que se sugere é o aproveitamento dos dados da própria amostra.

As estatísticas β_1 e β_2, são respectivamente o quadrado do coeficiente de assimetria (skewness) e o coeficiente de curtose (kurtosis) de uma distribuição.

O coeficiente de assimetria serve como medida de assimetria e para as distribuições alongadas à direita, como a maioria das que são aqui tratadas, seu valor é positivo ocorrendo o inverso para as alongadas à esquerda. No nosso caso, como desconhecemos a distribuição real das populações, o valor desse parâmetro terá de ser estimado em função dos elementos da amostra com base no seguinte estimador (vide ref.13).

$$\sqrt{b_1} = \frac{m_3}{(m_2)^{3/2}} ,$$

onde

$$m_3 = \frac{\sum_{i=1}^{n} x_i^3}{n} - 3 \cdot \frac{\sum_{i=1}^{n} x_i^2}{n} \frac{\sum_{i=1}^{n} x_i}{n} + 2 \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^3$$

$$m_2 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n} ,$$

b_1 é o estimador de β_1,

x_i são os elementos da amostra,

$\bar{x}$ é a média da amostra,

n número de elementos da amostra.

O coeficiente de curtose serve como medida de achatamento da forma da distribuição. O valor de tal parâmetro também terá de ser estimado pelos mesmos motivos, tomando-se como base o seguinte estimador:

$$b_2 = \frac{m_4}{(m_2)^2},$$

onde

$$m_4 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^4}{n},$$

b_2 é o estimador de β_2.

As estimativas para esses dois parâmetros das populações em consideração foram obtidos por meio do SPSS e estão reunidos na tabela 5.4. Observe-se que já incluímos o valor do quadrado da estimativa do coeficiente de assimetria obtido de cada amostra, pois este é que será usado na carta de E.S. Pearson.

Na figura 5.7 apresentamos a carta de E.S.Pearson já preenchida com os pontos obtidos plotando-se os valores das estimativas do par (b_1, b_2) de cada amostra. A amostra a que se refere cada ponto pode ser identificado pela sua numeração que corresponde ao código adotado no modelo.

Observe-se que todos os pontos se localizam na região de aderência impossível aos modelos de distribuição apre-

CÓDIGO	ATIVIDADE A QUE SE REFERE A AMOSTRA	COEFICIENTE DE ASSIMETRIA	(COEFICIENTE DE ASSIMETRIA)^2	COEFICIENTE DE CURTOSE
9	Outras atividades	1,150	1,323	0,910
10	Desen.programa "batch" de atualização	0,997	0,994	0,945
12	Desen.programa "batch" de relatório	1,364	1,860	1,501
14	Desen.programa "on-line" com SCOA	1,081	1,169	0,979
16	Desen.programa "on-line" sem SCOA	0,931	0,867	0,860
18	Desen.progr.em outra linguagem (não COBOL)	0,709	0,503	-0,269
11	Manut.programa "batch" de atualização	1,859	3,456	4,131
13	Manut.programa "batch" de relatório	1,956	3,826	3,678
15	Manut.programa "on-line" com SCOA	1,246	1,553	1,217
17	Manut.programa "on-line" sem SCOA	1,600	2,560	2,087
19	Manut.progr.em outra linguagem (não COBOL)	1,650	2,723	2,165

Tabela 5.4 - Estimativas dos coeficientes de assimetria e de curtose obtidas de cada amostra - e.p.a. -

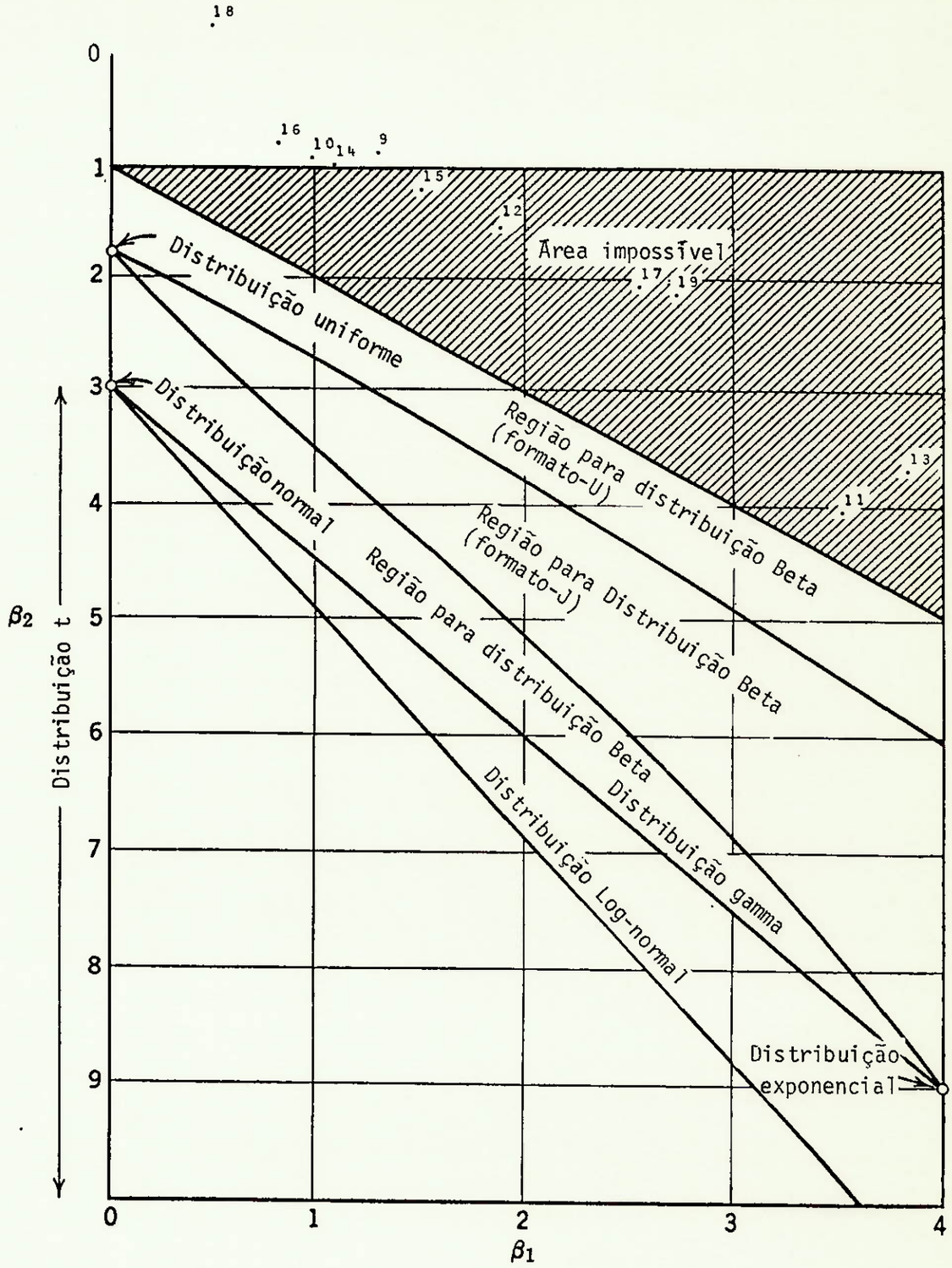


Figura 5.7 - Carta de E.S.Pearson, preenchida com os pontos obtidos das amostras

- adaptado da ref.13 -

sentados ou até mesmo fora da carta, estando nitidamente distantes de qualquer dos modelos. Assim, aproveitaremos os dados das próprias amostras para alimentarmos o modelo de simulação, mais especificamente, a distribuição das freqüências acumuladas (vide Anexo I e figura 4.11).

B. Distribuição do Tempo de Duração das Férias

Para se investigar essa distribuição foi necessário consultar os programadores e obter informações verbais devido à inexistência de dados organizados a respeito.

Constatou-se que os programadores não costumam parcelar o cumprimento de férias, ou seja, cumprem-nos de uma vez só. Quanto à sua duração, escolhem entre 20, 25 ou 30 dias corridos com probabilidades praticamente iguais. Assim sendo, o problema da especificação do tempo de duração das férias acaba-se resumindo na determinação de uma função descontínua (vide ref. 10).

Considerando o calendário e o horário de trabalho normal do Setor de Informática e Sistemas de Informação, obtemos o seguinte fator de conversão médio de dias corridos para horas úteis:

$$\frac{\left(\frac{\text{dias úteis}}{\text{ano}}\right) \cdot \left(\frac{9 \text{ horas úteis}}{\text{dia útil}}\right)}{\left(\frac{\text{dias corridos}}{\text{ano}}\right)} = \frac{244{,}9}{365} = 6{,}016 \ \frac{\text{horas úteis}}{\text{dia corrido}}$$

Então, com base nesses dados, especificamos a função que é apresentada na tabela 5.5 com a qual alimentaremos o modelo. Essa tabela fornece o tempo de duração das férias em função do valor dos números randômicos entre 0 e 1 a serem gerados pelo GPSS (variável independente).

VARIÁVEL INDEPENDENTE x	TEMPO DE DURAÇÃO (horas úteis)
$0,000 \leq x < 0,333$	120,3
$0,333 < x < 0,666$	150,4
$0,666 < x \leq 1,000$	180,5

Tabela 5.5 - Função descontínua para a determinação do tempo de duração das férias
- e.p.a. -

C. Distribuição dos Outros Tempos Envolvidos no Atendimento de Serviços Produtivos

Vimos no subitem 5.2.4. que o modelo de atendimento desenvolvido requererá dados a respeito do tempo que normalmente se leva no atendimento de serviços de manutenção ou desenvolvimento de programas que não chegam a ser concluídas. Além disso, para o caso particular de desenvolvimento de programas novos, como o atendimento se divide em etapas, são necessários dados a respeito do tempo que se leve para a execução dos serviços de digitação e do tempo que se gasta em cada etapa de depuração ou de testes.

Na falta de qualquer tipo de dados a respeito e pelo excessivo tempo que se levaria para coletar amostras pessoalmente, acompanhando o trabalho da Programação (com possibilidade de atrapalhar suas atividades ou dos programadores sentirem-se incomodados e mudarem de comportamento), já que os programadores não contribuem com preenchimento de formulários ou questionários (desconfiança), não restou outra saída senão estimar os dados com base em informações verbais colhidas em consultas informais.

Apresentamos na tabela 5.6 os dados estimados para a média desses tempos e incluímos também os valores máximo e mínimo que se tem verificado normalmente, já que os palpites das pessoas consultadas foram apresentadas na forma $(\bar{t} \pm \Delta t)$ onde $\bar{t}$ corresponde à média e o Δt à semi-amplitude do intervalo de tempo estimado.

VARIÁVEL	VALOR MÍNIMO (horas úteis)	VALOR MÉDIO (horas úteis)	VALOR MÁXIMO (horas úteis)
Tempo de atendimento para serviços não concluíveis	1,0	4,0	7,0
Tempo necessário para a digitação do programa	4,5	9,0	13,5
Tempo normalmente levado numa etapa de depuração ou de testes	1,0	3,5	6,0

Tabela 5.6 - Estimativas para os outros tempos envolvidos no atendimento de serviços produtivos

- e.p.a. -

A distribuição real dessas variáveis também é desconhecida, mas levando-se em conta que as tarefas envolvidas não podem ser executadas num tempo tendendo a zero e muito menos demorar indefinidamente, podemos supor que ela seja truncada em ambas as extremidades. Assim sendo, na falta de melhores informações, adotaremos (alimentação do modelo) uma distribuição uniforme com intervalo definido pelos valores mínimo e máximo estimados, para cada uma dessas variáveis, assumindo a hipótese de não estarmos assim, fugindo em muito da realidade. Trata-se de uma saída razoável, já que sejam a e b os extremos de uma distribuição uniforme, então o seu valor esperado é dado pelo ponto médio do intervalo, ou seja:

$$E(t) = \frac{a + b}{2}$$

Conseguimos assim, pelo menos respeitar os valores médios coletados.

Ilustramos, a título de exemplo, a função de distribuição acumulada para o tempo de atendimento dos serviços não concluíveis (para os demais casos a figura é análoga) na figura 5.8.

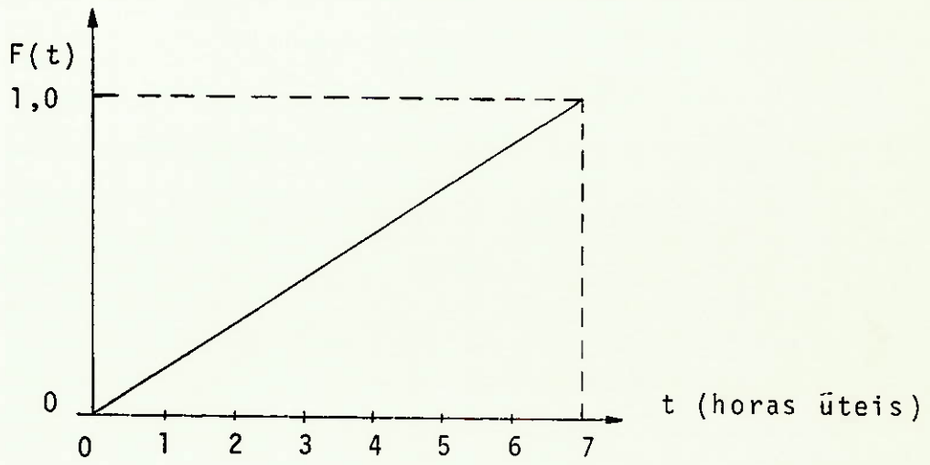


Figura 5.8 - Função de distribuição acumulada para o tempo de atendimento dos serviços não concluíveis
- e.p.a. -

D. Dados Complementares

O modelo necessitará de alguns dados complementares dos quais alguns serão fixos (se bem que facilmente reajustáveis) e outro, sujeito à alterações.

São dados complementares fixos o intervalo de tempo simulado a ser coberto, que definimos como sendo de 12 meses (1 ano), e o número de horas úteis a ser considerado por mês, que no caso é de 183 horas úteis.

O outro dado é o número de elementos (inteiro) do grupo médio de programação, ou seja, a capacidade do respecti

vo depósito de transações. Para a situação atual, esse valor é de 7 programadores.

Em virtude das dificuldades que encontramos para o levantamento de dados de tempos, não pudemos obter distribuições com rigorosa precisão. Entretanto, o programa desenvolvido é suficientemente flexível para poder ser facilmente adaptado, caso consigamos algum dia, obter dados mais precisos.

5.4. DESENVOLVIMENTO DE UM PROGRAMA PARA O MODELO

O programa foi desenvolvido em vários módulos, cada qual com funções bem específicas, quais sejam:

a) Gerador de programas em mãos na situação inicial;

b) Gerador de férias;

c) Gerador de outras atividades;

d) Gerador de desenvolvimentos de programa novo para cada um dos 5 tipos de programas;

e) Gerador de manutenções para cada um dos 5 tipos de programas;

f) Atendimento principal;

g) Finalização principal;

h) Atendimento e finalização das atividades que não serão concluídas;

i) Desenvolvimento de programas novos;

j) Conclusão de programas novos;

k) Coleta de dados extras.

O módulo em a é o responsável pelo estabelecimento das condições iniciais de simulação. Os módulos em b , c ,

d e e (num total de 12) geram as chegadas de atividades, e ao contrário do módulo em a que só funciona no instante inicial, alimentam intermitentemente o sistema simulado até o fim.

Os módulos apresentados em f , g , h , i e j foram programados com base no modelo de atendimento descrito no subitem 5.2.4. (vide figuras 5.4 e 5.5).

A integração de todos esses módulos é ilustrada pela figura 5.9.

Por último, temos o módulo para a coleta de dados extras que não está integrado com os demais. Este módulo acompanha a simulação coletando dados que nos interessam ao fim de cada mês e computando alguns dados globais ao término do intervalo de tempo simulado (1 ano). Esses dados serão utilizados em complementação com alguns dados coletados automaticamente pelo GPSS durante a simulação, para a emissão de relatórios personalizados juntamente com as listagens de saída de cada processamento. Esses relatórios serão apresentados posteriormente.

Para o GPSS, o tempo é uma grandeza inteira e portanto, durante a simulação, os valores gerados com base em distribuições contínuas, são truncados. Assim para não termos dúvidas quanto à validade dos resultados a serem obtidos, adotamos uma unidade de medida de tempo inteira que é dado por:

1 hora útil = 10 u.t.s.

1 u.t.s. = 0,1 hora útil = 6 minutos úteis,

onde

u.t.s. é a unidade de tempo simulado.

Conseguiremos assim, processar simulações envolvendo extensos intervalos de tempo simulado (1 ano) com os valores de tempos, sendo manipulados com uma precisão da ordem de 6 minutos. A unidade escolhida também é muito prática, pois

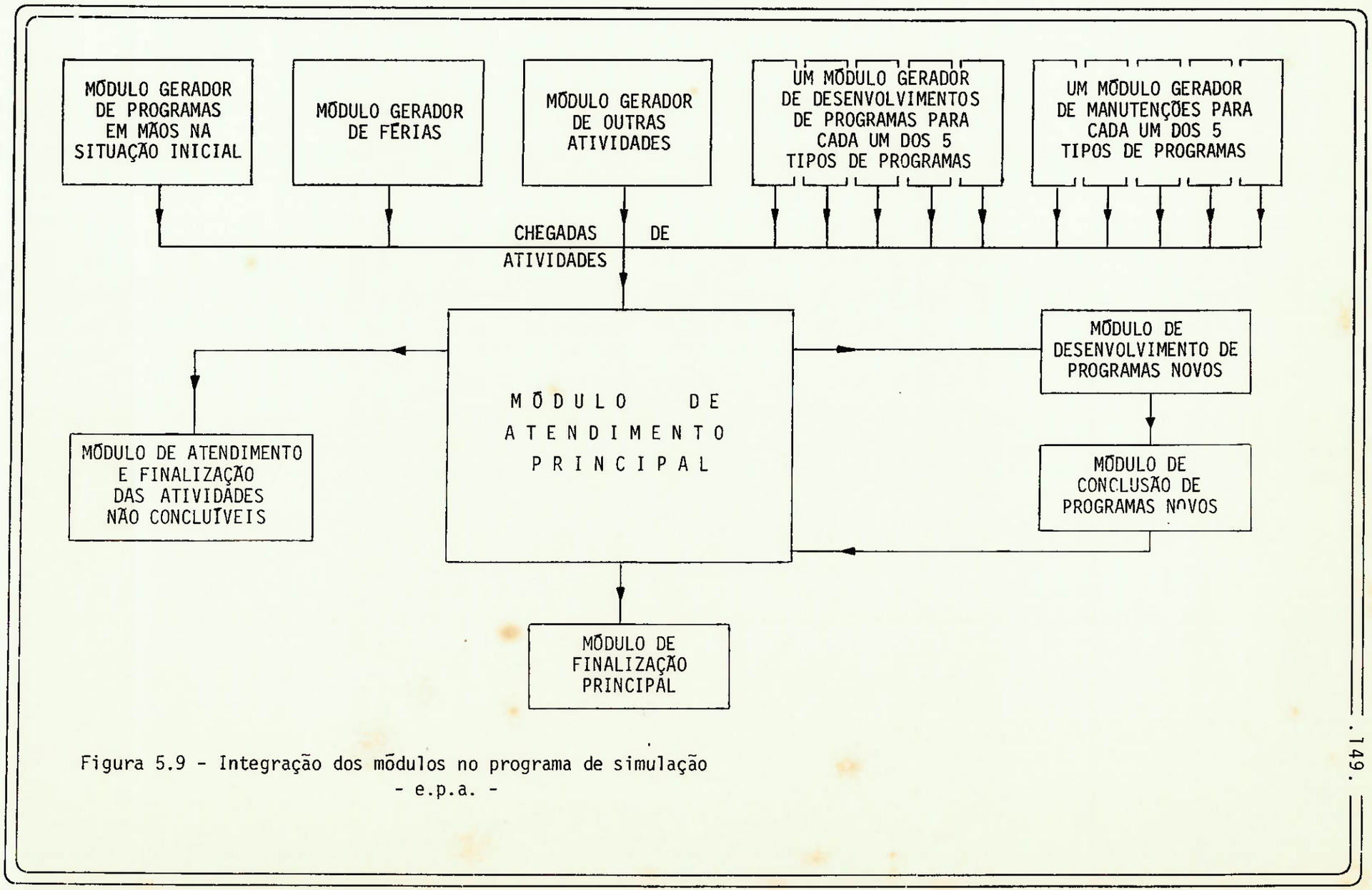


Figura 5.9 - Integração dos módulos no programa de simulação - e.p.a. -

requer apenas um zero a mais para transformar os valores obtidos em horas úteis.

Para se alcançar o objetivo do trabalho, ou seja, um parecer crítico quanto à viabilidade e a validade de se introduzir o Meta-Cobol no ambiente da Programação, foi necessário realizar uma série de simulações (diversas situações), sendo que em cada caso, evidentemente, tivemos de proceder a algumas alterações no programa original. Incluímos, a título de exemplo, no Anexo II, a listagem do programa utilizado para a simulação da situação atual, que foi fruto da etapa de calibração do modelo, e em cima do qual, procedemos a verificação da validade do modelo (ambas as etapas são discutidas no item seguinte). Observe-se que o programa está suficientemente bem documentado, de modo que o leitor poderá, facilmente, localizar o trecho que corresponde à cada módulo do modelo e a sua inter-ligação como foi apresentada na figura 5.9, além de poder visualizar como se processará o fluxo de transações (atividades da Programação).

Como o programa original é suficientemente flexível, as adaptações que lhe foram necessárias para o processamento das simulações do aproveitamento do Meta-Cobol na Programação, foram poucas e relativamente simples. Tais alterações são comentadas mais adiante.

5.5. VALIDAÇÃO DO MODELO

Neste item apresentaremos como verificamos a validade do modelo desenvolvido.

Vimos no subitem 5.2.5. que o porcentual p de manutenções que vai receber maior prioridade que os programas novos era desconhecido e que ficou de ser estimado experimentalmente, utilizando-se o próprio modelo, ou seja, na fase de calibração. Este percentual p é o único valor que ainda falta ser especificado para "fecharmos" o modelo. Ele foi adotado

como parâmetro de calibração porque influi diretamente na atividade de maior peso (manutenções) na carga de trabalho da Programação e porque conforme o seu valor, o fator

$$q = \frac{\text{média de programas em mãos}}{\text{média de programas liberados por mês}}$$

também varia. A não consideração desse parâmetro, implicaria em adotar critério de atendimento idêntico tanto para manutenções como para desenvolvimento de programas novos em fila (fila principal do modelo) e, assim sendo, essa relação tenderia a ter mesmo valor para ambos os tipos de serviço, o que não corresponderia à realidade (vide subitem 4.2.1.).

Experimentamos o programa simulando a situação atual (ano de 1984) com 4 valores diferentes de p, quais sejam 33,3% (1/3), 66,6% (2/3), 44,4% (4/9) e 55,6% (5/9). O que fizemos foi antes experimentar os dois primeiros valores (extremos) o que nos permitiu desconfiar que o valor real de p se encontrava entre os mesmos, e somente então, processar os experimentos com os demais valores.

Com base nos resultados obtidos (os relatórios que foram consultados constam no Anexo III), cujos dados estão sumarizados na tabela 5.7, levantamos a curva do fator q em função de p tanto para os programas novos como para os enviados para manutenção, as quais são mostradas na figura 5.10. Observe-se pelos resultados que o sistema simulado comportou-se aproximadamente, em regime, como ocorre na realidade, e independentemente do valor de p testado, o número de programas recebidos e o de entregues praticamente se equivaleram. Entretanto, além do fator q, o número médio de programas em mãos, também variou em função do valor de p. Isso é ilustrado na figura 5.11.

Uma vez levantados todos esses dados pudemos, então, partir para a estimação do valor mais adequado. Este seria aquele que permitisse que o programa reproduzisse com a máxima fidelidade a atual situação. Os valores reais das variáveis

DESCRIÇÃO DA VARIÁVEL	VALOR DA VARIÁVEL CONFORME p				VALOR REAL DA VARIÁVEL
	p=33,3%	p=44,4%	p=55,6%	p=66,6%	
número médio de programas recebidos por mês	63,6	66,5	69,2	61,7	64,17
número médio de programas liberados por mês	62,6	61,3	67,9	63,0	67,40
número médio de programas em mãos	77,0	71,5	55,7	35,1	50,27
número médio de programas novos em mãos	23,26	26,05	18,62	14,43	18,95
número médio de programas novos liberados por mês	12,92	13,50	12,67	11,83	13,58
fator q para programas novos	1,80	1,93	1,47	1,22	1,40
número médio de programas para manutenção em mãos	53,73	45,44	37,02	20,50	31,32
número médio de programas para manutenção liberados por mês	49,75	47,83	55,25	51,25	53,82
fator q para programas para manutenção	1,08	0,95	0,67	0,40	0,58

Tabela 5.7 - Resumo dos resultados obtidos nas simulações para calibração do modelo - e.p.a. -

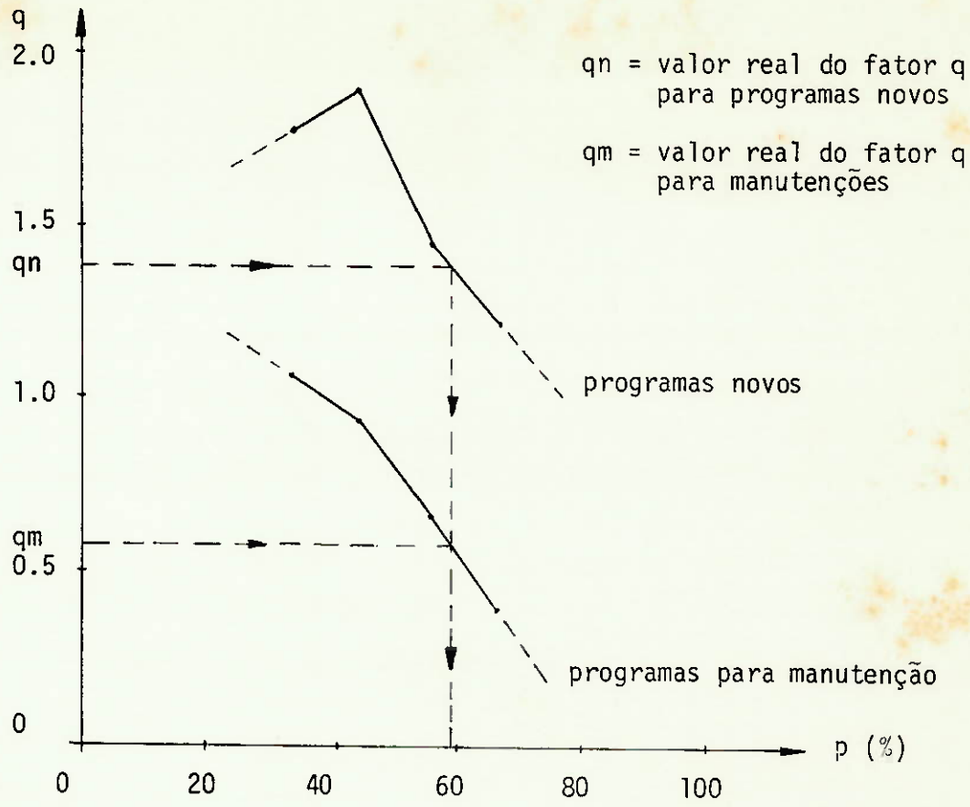


Figura 5.10 - Gráfico do fator q em função de p
- e.p.a. -

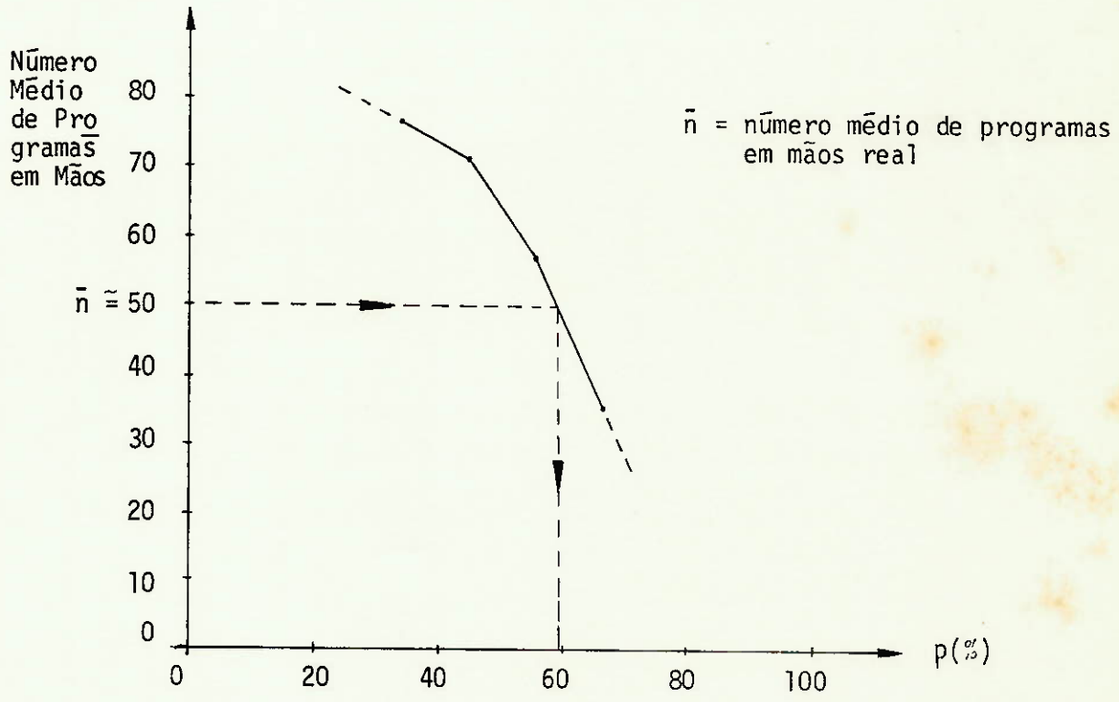


Figura 5.11 - Gráfico do número médio de programas em mãos em função de p
- e.p.a. -

consideradas para efeito de comparação (fator q para manutenções e programas novos, e o número médio de programas em mãos) foram retiradas das tabelas do Capítulo 4 e devidamente ajustadas para o nosso grupo médio de programação, estando incluídas na tabela 5.7. Como os dados coletados não abrangem todo o período simulado, estamos admitindo a hipótese de que as médias obtidas, sejam válidas para todo o ano de 1984, o que é razoável, pois as condições vigentes no período a que se referem os dados, podem ser extrapolados para todo o decorrer do ano de 1984. Não houve, e nem há previsão de nenhuma alteração conjuntural significativa que possa invalidar tal consideração. Esta hipótese aqui assumida, será também, tomada como base para todo o restante do trabalho. No gráfico das figuras 5.10 e 5.11, está indicado que tal valor estaria muito próximo de 60%. Adotamos então, o valor de 60% para p, admitindo ser esse um bom valor, apesar da simplicidade do método utilizado para a sua estimação (um único processamento para cada valor de p e utilização de gráficos) e consideraremos, assim, que o modelo já esteja devidamente calibrado. Processamos, então, outra simulação para a situação atual com esse valor de p e obtivemos os resultados apresentados no Anexo IV. Observe-se que os mesmos estão muito próximos dos dados apresentados no Capítulo 4.

Como a simulação também é uma técnica de experimentação, o ideal teria sido executar vários processamentos para cada valor de p e coletar amostras razoáveis para cada variável tomada como elemento de comparação. E somente então, baseando-se em técnicas estatísticas mais rigorosas (ex. testes não paramétricos) procedermos a estimação do valor de p final. Isso, evidentemente, seria muito trabalhoso e custoso, pois requereria um acréscimo considerável no consumo de computador e de tempo, e portanto, não foi aplicado no presente trabalho.

5.6. PLANEJAMENTO DAS SIMULAÇÕES DO USO DO META-COBOL NO AMBIENTE DA PROGRAMAÇÃO

Para se avaliar a viabilidade e a validade de se desenvolver o Meta-Cobol e introduzí-lo no ambiente da Programa

ção, foi necessário definir como o modelo de simulação seria aproveitado e planejar as etapas que seriam necessárias para isso.

Optamos, então, por basearmos o estudo de avaliação num método comparativo: simular a Programação continuando a trabalhar exatamente como na situação atual nos próximos anos, e simular o seu trabalho contando com o aproveitamento do Meta-Cobol. Conseguiríamos assim, alcançar os principais objetivos desse Capítulo quais sejam:

- verificar o impacto que se verificaria no ambiente da Programação caso se introduzisse o Meta-Cobol;

- avaliar o uso do Meta-Cobol;

- obter informações para podermos no Capítulo seguinte, procedermos o estudo de viabilidade econômica do projeto do Meta Cobol;

- realizar um estudo de análise de sensibilidade.

Planejamos então, a divisão desse estudo de avaliação conforme a seqüência abaixo:

a) Obter uma previsão da data de entrada em operação do Meta Cobol e do seu período de aproveitamento (caso viesse a ser aprovado e desenvolvido), ou seja, da duração de sua vida útil, de modo a podermos definir o horizonte de simulação a ser adotado.

b) Caracterizar o cenário mais provável para este horizonte de simulação.

c) Preparar os dados necessários para a alimentação do programa, com base no cenário levantado.

d) Proceder os ajustes necessários ao programa de simulação.

e) Elaborar critérios para a simulação das duas situações; projeção do futuro com e sem a introdução do Meta-Cobol, de modo a podermos comparar, consistentemente, os resultados a serem obtidos.

f) Simulação do futuro com o aproveitamento do Meta-Cobol.

g) Simulação do futuro sem o aproveitamento do Meta-Cobol.

h) Executar simulações complementares para se realizar um estudo de análise de sensibilidade da alternativa a ser avaliada (uso do Meta-Cobol).

i) Interpretar os resultados.

Desta seqüência, as etapas descritas em a , b , c e d são tratados nos subitens seguintes, e as etapas em e, f, g, h e i, são tratadas no item 5.8. fechando o presente Capítulo.

5.6.1. DEFINIÇÃO DO HORIZONTE DE SIMULAÇÃO

Para podermos definir o horizonte a ser abrangido pelas simulações do futuro, consultamos as pessoas que têm estado envolvidas no desenvolvimento do ante-projeto do Meta-Cobol.

Como foi descrito no Capítulo 3, o Meta-Cobol seria, caso viésse a se concretizar, uma ferramenta muito poderosa. Entretanto, evidentemente, não seria possível e nem aconselhável, desenvolver o Meta-Cobol e colocá-lo em operação dotado de todos os recursos que lhe foram atribuídos, desde a sua versão inicial. O objetivo, segundo os maiores responsáveis pelo seu ante-projeto, é o de se desenvolver o Meta-Cobol gradualmente, e ir implantando versões intermediárias, até chegar no final do processo, ao Meta-Cobol na sua configuração máxi

ma. Existem muitas vantagens em se adotar tal tipo de metodologia (tática), tais como, a possibilidade de se eliminar com maior facilidade as muitas irregularidades que costumam acompanhar qualquer versão original; a possibilidade de se verificar o uso da ferramenta e sentir a sua adequação às reais necessidades de seus usuários; a possibilidade de se chegar à versão final, adotando-se as soluções mais adequadas pelos aprimoramentos que podem ser incluídos no projeto, com as constatações experimentais a serem colhidas durante o acompanhamento do desempenho das versões iniciais, entre outras.

O desenvolvimento do Meta-Cobol seria delegado à Equipe para Pacotes Padronizados, assim sendo, a primeira versão poderia ser tratada num ambiente menor (restrito) constituído pelo grupo de programadores dessa equipe. As versões seguintes, bem mais poderosas, já passariam a ser usadas por toda a Programação.

Pela experiência acumulada no desenvolvimento de outras ferramentas próprias de "software", estima-se que, se o desenvolvimento do Meta-Cobol fosse delegado a um grupo encabeçado por 1 analista de sistemas, mais 2 programadores (todos já familiarizados com o seu ante-projeto e tendo participado de outros projetos de natureza semelhante), ficando sob a orientação do supervisor da Equipe para Pacotes Padronizados, a versão final poderia ser alcançada num prazo de um ano.

A dúvida maior é quanto ao início dos trabalhos de desenvolvimento. No entanto, devido à urgência em se chegar a uma decisão, quanto à alternativa a ser adotada para se enfrentar o problema tratado nesse trabalho (aumentar produtividade no desenvolvimento de programas "batch"), acredita-se que a avaliação final das alternativas possa ser realizada ainda durante o ano de 1984. Caso seja dado parecer favorável ao desenvolvimento e implantação do Meta-Cobol, o início dos trabalhos poderia se dar em janeiro de 1985.

Desta forma, a primeira versão a ser utilizada em caráter experimental e em ambiente limitado, seria liberado

ainda no primeiro semestre de 1985. No início do segundo semestre, o Meta-Cobol já numa segunda versão de maior potencial, seria lançado para uso geral na Programação, entrando a partir daí, em fase operacional. No decorrer do segundo semestre estariam previstos alguns melhoramentos, ampliando em muito a potencialidade do Meta-Cobol. Finalmente, no início de 1986, já estaria implantada a versão final no Meta-Cobol . Esta versão, evidentemente, não seria a definitiva, pois estaria sujeita às eventuais manutenções ou mesmo a alguns acréscimos de recursos. Entretanto, as alterações não seriam substanciais de modo a mudar em muito as suas características e potencialidades.

Quanto à sua vida útil, acredita-se que não seria muito longa. A introdução do Meta-Cobol é uma alternativa de solução a médio prazo. Como a evolução tecnológica tanto na área de "hardware" como de "software" tem se processado rapidamente, será inevitável o aparecimento de novos produtos que possam suplantar o Meta-Cobol e forçar o seu desuso pelo obsoletismo tecnológico. A expectativa é a de que sua vida se situe entre 5 e 6 anos, com incertezas razoáveis.

Neste trabalho admitiremos que a sua vida se encerre ao final de 1989, ou seja, 4 anos e meio após a sua entrada em fase operacional. É uma previsão mais pessimista (estamos estimando um pouco para menos) mas de qualquer modo, caso o Meta-Cobol se mostre viável nesse período, tornar-se-ia, melhor ainda caso se registrasse uma vida efetivamente mais longa, e se constituísse uma alternativa frágil (pouco viável), mesmo com algum prolongamento em sua vida útil, dificilmente essa situação poderia ser revertida.

Assim sendo, as simulações do futuro (projeções) foram limitadas aos anos de 1985 a 1989.

5.6.2. O CENÁRIO DE SIMULAÇÃO

Como as simulações serão processadas para um período do futuro, é necessário elaborar um cenário que se aproxime

ao máximo do que realmente poderá vir a ocorrer nos próximos anos. Quando se processa uma simulação de uma situação futura, os resultados obtidos são referentes ao particular cenário baseado no qual os dados de alimentação do programa foram levantados. Não seria adequado utilizar, indiscriminadamente, tais resultados para se projetar e avaliar qualquer situação. Os resultados não são genéricos. Para se avaliar uma outra situação, seria necessário reprocessar o programa com um outro conjunto de dados de entrada mais adequado ao caso.

No presente trabalho, como estamos interessados num estudo comparativo entre o comportamento do sistema a ser simulado, com e sem o aproveitamento do Meta-Cobol, para ambas as situações consideraremos o mesmo cenário básico (diferirá apenas pela presença ou ausência do Meta-Cobol no ambiente). As simulações poderiam ser processadas para uma diversidade de cenários mas, evidentemente, isso não foi possível de se realizar pela enorme demanda de trabalho e de tempo que isso iria requerer.

Para a elaboração do cenário de simulação, foi necessário consultar diversas pessoas das mais diversas áreas do Setor de Informática e Serviços de Teleprocessamento e além disso, para se poder detectar a existência de algum viés nas informações procurou-se colher a opinião de elementos externos a este ambiente, como os professores do Departamento de Engenharia de Produção da Escola Politécnica da U.S.P.. O resultado dessas investigações é o cenário descrito a seguir em linhas gerais.

a) a capacidade de produção (desenvolvimento) de programas novos deverá ser aumentado de 10%, cumulativamente, ao ano.

b) as atuais taxas de manutenção mensal (percentual sobre o número de programas implantados e em operação) deverá se manter estável.

c) a atual configuração de "software" básico (Sistema Operacional OS/VS1, monitor de TP CICS, método de acesso a arquivos VSAM, gerenciador de redes NCP) deverá ser conservada.

d) as atuais ferramentas de "software", tanto as compradas como as desenvolvidas continuarão a ser usadas pelo menos até o fim do horizonte de simulação.

e) procurar-se-á continuar dispendendo somente 20% do tempo útil com as outras atividades.

f) continuar-se-á a manter os programadores trabalhando com o mínimo de ociosidade, ou seja, forçando-os a trabalhar sob pressão (razoável número de programas em mãos).

g) manter-se-á a determinação de que cada programa novo seja compilado ou processado para testes somente uma vez ao dia.

As afirmativas dadas por a , b e e servirão de base para se projetar, respectivamente, as taxas de geração de desenvolvimento de programas novos, manutenções de programas e as chamadas outras atividades, para o período compreendido entre 1985 e 1989.

A afirmativa dada por c se baseia em dois aspectos. Primeiro é o pouco tempo que se passou desde a implantação da atual configuração de "software" básico. Isto ocorreu há cerca de 3 anos, quando também foi necessário realizar um grande trabalho de conversão dos sistemas que haviam sido anteriormente desenvolvidos no antigo ambiente e, evidentemente, não se deseja pelo menos nos próximos anos, repetir tal experiência que exige consideráveis investimentos e tempo improdutivo. Com a exceção do sistema operacional OS/VS1, todos os demais elementos básicos eram relativamente recentes no mercado brasileiro na ocasião. Hoje, todos os elementos que fazem parte da configuração de "software" básico estão plenamente consolidados no mercado, e ao que tudo indica, pelo menos aqui no Brasil, é razoável supor que possam continuar normalmente em uso nos próximos anos, sem enfrentar maiores problemas de obsolетismo tecnológico, já que tanto a IBM como as demais empresas multinacionais da área de "software" continuam lançando e desenvolvendo produtos compatíveis com tais elementos básicos e

também, porque a própria IBM se preocupa em aprimorar os seus produtos, lançando periodicamente versões melhoradas ou fornecendo recursos complementares. O segundo aspecto que fortalece esta afirmativa, é o fato das ferramentas próprias de "software" como o SCOA e o DB VSAM, terem sido desenvolvidas partindo-se da premissa de ter de se integrar com esta configuração básica[1]. Uma alteração nessa configuração requereria também consideráveis reformulações no projeto de tais ferramentas, e evidentemente, é um ônus a ser evitado.

Como o leitor já deve ter notado, a afirmativa dada por d está intimamente relacionada com a dada por c . Mantendo-se a atual configuração de "software" básico, poder-se-á manter também as atuais ferramentas de "software" utilizadas pelo pessoal de desenvolvimento de sistemas. Admitiremos assim, que, ao se manter tal situação, os atuais tempos efetivos de atendimento a serviços produtivos, continuarão a se distribuir como na situação atual. A exceção ficaria por conta do tempo efetivo para o desenvolvimento de programas "batch" de atualização e de relatório, que se modificariam (redução) no caso da introdução do Meta-Cobol. Além do Meta-Cobol, não há previsão de se implantar nenhuma outra ferramenta de "software" que possa alterar significativamente os tempos atualmente registrados.

A afirmativa f será fundamental para se verificar os resultados de cada simulação e determinar o tamanho do sistema a ser simulado (número de programadores) para cada ano. Determinando-se qual seria esse tamanho adequado, estaremos, indiretamente, dimensionando o contingente de toda a Programação, e desta forma, poderemos obter os dados básicos necessários para o estudo de viabilidade econômica do Meta-Cobol.

---

[1] O mesmo ocorreria com o desenvolvimento do Meta-Cobol, caso venha a ser aprovado.

Evidentemente, os critérios de atendimento às atividades não sofrerão qualquer alteração, pois são intrínsecos à natureza de cada atividade e não dependem do tamanho do sistema e muito menos do tempo cronológico. Isso, somando-se a afirmativa dada por g permitirá conservar a escala de prioridades elaborada para o programa original.

Um outro aspecto, embora não incluído diretamente no cenário, que merece ser comentado é a influência das mudanças na configuração de "hardware" do ambiente. Esse tipo de mudança pode ser assimilado tranquilamente pelo pessoal de desenvolvimento de sistemas, assim como recentemente um computador IBM/370 do C.P.D. foi substituído por outro da linha 4341 de forma transparente aos analistas e programadores. Isso decorre da política dos fabricantes em desenvolver equipamentos novos compatíveis com os modelos mais recentes, para que as dificuldades de instalação e readaptação não constituam obstáculos à sua venda.

5.6.3. PREPARAÇÃO DOS DADOS PARA A SIMULAÇÃO DO FUTURO

Cada simulação abrangerá o período de um ano. Por limitações do programa não será possível considerar que as atividades da Programação crescerão, gradativamente aos poucos, ou mesmo, sujeitas às oscilações periódicas. Cada ano, será simulado ininterruptamente, com base em dados médios e o sistema (grupo médio) deverá se comportar aproximadamente em regime. Assim, as variáveis crescentes do sistema a ser simulado, sejam determinísticos (ex. número de programadores) ou estocásticos (ex. número de programas novos liberados por mês), deverão aumentar por "patamares" (aos saltos) com o passar dos anos, como é ilustrado na figura 5.12. Isso, no entanto, não prejudicará o trabalho de avaliação, pois estamos interessados no desempenho da Programação a longo prazo, e para isso, os dados médios que obteremos para cada ano serão mais do que suficientes.

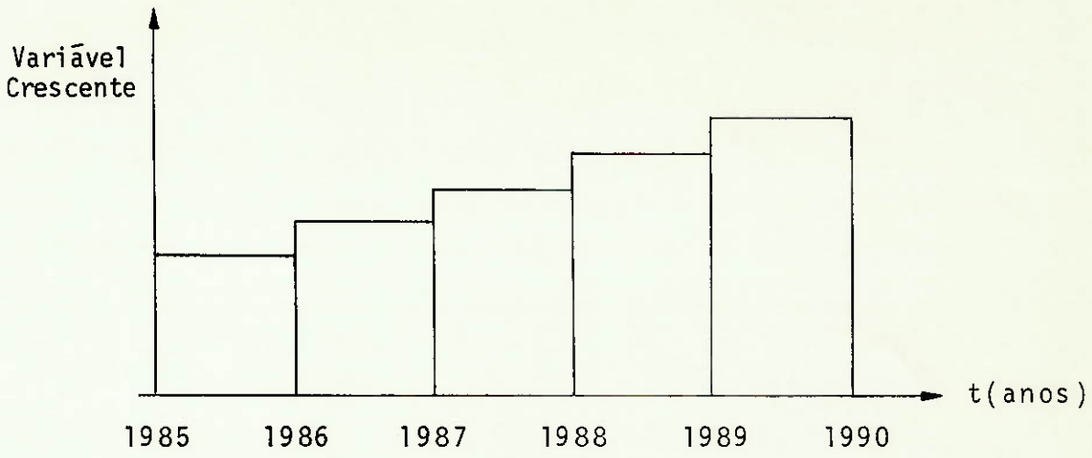


Figura 5.12 - Gráfico das variáveis crescentes do sistema - e.p.a. -

Com base no cenário de simulação levantado, identificamos os dados que precisariam ser reajustados e aqueles que poderiam ser mantidos exatamente como na situação atual. As distribuições dos tempos efetivos de atendimento, com as exceções daquelas que se referem ao desenvolvimento de programas "batch" para a simulação do futuro, considerando o uso do Meta-Cobol, deverão ser mantidas. Os dados que determinarão a situação inicial da simulação, os que determinarão a geração de transações (atividades) e o número de programadores do sistema, terão de ser reajustados para cada ano do horizonte de simulação. Apresentamos a seguir os dados projetados para cada ano.

A. Dados para a Geração de Programas Novos

A direção do Setor de Informática e Sistemas de Teleprocessamento espera que a Programação aumente a sua capacidade de desenvolvimento de programas novos a uma taxa de 10% ao ano, cumulativamente.

Admitindo-se que essa capacidade aumente igualmente para o desenvolvimento de qualquer tipo de programa, tomamos como base os dados do número médio de programas liberados

por mês e por tipo constantes na tabela 4.4 elaboramos a tabela 5.8. Esta tabela fornece para cada ano do horizonte de simulação, a expectativa da quantidade total de programas novos por tipo que deverá ser produzido por toda a Programação. Observe-se que se admitimos a mesma taxa de crescimento para todos os tipos de programas, o perfil (distribuição relativa) dos programas a serem produzidos manter-se-á constante.

Com os dados da tabela 5.8 pudemos então calcular para o grupo médio a ser simulado, a taxa média de chegadas por tipo de programa ($\lambda_{i,t}$) e o tempo médio entre chegadas por tipo de programa ($\theta_{i,t}$) para cada ano. Esses dados são apresentados respectivamente pelas tabelas 5.9 e 5.10. Novamente, admitimos que essas chegadas se dêem à uma taxa constante ao longo de cada ano e que se trata de eventos independentes, de modo que possamos representá-las por meio de uma distribuição de Poisson. O tempo médio entre chegadas para um determinado tipo de programa i durante o ano t foi obtido pelas expressões:

$$\theta_{i,t} = \left(\frac{1}{\lambda_{i,t}}\right) \cdot r \qquad \text{horas úteis,}$$

e

$$\lambda_{i,t} = \frac{NPT_{i,t}}{N_1 \cdot N_2} \qquad \frac{\text{programas}}{\text{mês}},$$

onde

$r = 183$ = fator de conversão de mês em horas úteis,

$NPT_{i,t}$ = número estimado de programas do tipo i a ser produzido pela Programação no ano t,

$N_1 = 12$ = número de meses por ano,

$\frac{1}{N_2} = \frac{1}{6}$ = escala de redução da Programação para o grupo médio.

TIPO DE PROGRAMA \ ANO	1984	1985	1986	1987	1988	1989
"batch" de atualização	277	305	335	369	406	446
"batch" de relatório	275	302	333	366	403	443
"on-line" com SCOA	192	211	232	256	281	309
"on-line" sem SCOA	203	223	246	270	297	327
em outras linguagens (não COBOL)	31	34	38	41	45	50
TOTAL	978	1.075	1.184	1.302	1.432	1.575

Tabela 5.8 - Projeção do número de programas novos a ser produzidos por ano e por tipo (para a Programação)
- e.p.a. -

TIPO DE PROGRAMA \ ANO	1985	1986	1987	1988	1989
"batch" de atualização	4,24	4,65	5,13	5,64	6,19
"batch" de relatório	4,19	4,63	5,08	5,60	6,15
"on-line" com SCOA	2,93	3,22	3,56	3,90	4,29
"on-line" sem SCOA	3,10	3,42	3,75	4,13	4,54
em outras linguagens (não COBOL)	0,43	0,47	0,57	0,63	0,69
TOTAL	14,93	16,43	18,08	19,89	21,88

Tabela 5.9 - Projeção da taxa média mensal de chegadas de programas por tipo para cada ano (para o grupo médio)
- e.p.a. -

TIPO DE PROGRAMA \ ANO	1985	1986	1987	1988	1989
"batch" de atualização	43,3	39,3	35,8	32,5	29,6
"batch" de relatório	43,5	39,6	36,0	32,7	29,7
"on-line" com SCOA	62,4	56,7	51,5	46,9	42,6
"on-line" sem SCOA	59,0	53,6	48,8	44,3	40,3
em outras linguagens (não COBOL)	386,4	351,2	319,3	290,3	263,9

Tabela 5.10 - Tempo médio entre chegadas em horas úteis para cada tipo de programa e por ano (para o grupo médio) - e.p.a. -

Observe-se que como procuraremos simular o sistema de modo que se comporte aproximadamente em regime (como deve ocorrer na realidade), estimamos as chegadas com base na capacidade de produção da Programação. Isto na prática ocorre pelo próprio controle que as equipes de análise realizam sobre a carga de trabalho dos programadores, procurando mantê-los sem ociosidade e equilibrar a "vazão" de entrada de programas com a "vazão" de saída. Se os analistas não praticassem esse controle, e passassem a enviar mais programas independentemente da capacidade de produção dos programadores, a fila de programas tenderia a crescer e degradar a qualidade e dos serviços e os prazos de atendimento.

B. Dados para a Geração de Manutenções

As manutenções são provocadas pelas alterações que ocorrem no ambiente em que os Sistemas de Informação são utilizadas. As condições do ambiente na época em que os Sistemas de Informação são desenvolvidos não prevalecem para sempre. Sempre que ocorre algum evento capaz de prejudicar o

aproveitamento do sistema instalado, faz-se necessário executar manutenções para se prolongar a sua vida útil.

Os fatores que podem motivar a necessidade de se executar manutenções são os mais diversos e, praticamente, fogem de nosso controle. Nesse trabalho, assumiremos uma hipótese de que a taxa de manutenções seja um percentual constante, dentro do horizonte de simulação, do número de programas instalados e em operação. Com base nela executaremos a projeção do volume de manutenções a ser realizado em cada ano. Essa hipótese, tomando-se como base o período de 1 ano pode ser representada por:

$$\left(\begin{array}{c}\text{taxa de manutenção}\\ \text{anual média}\end{array}\right) = \frac{\left(\begin{array}{c}\text{número de programas}\\ \text{enviados para manutenção}\end{array}\right)}{\left(\begin{array}{c}\text{número de programas instalados}\\ \text{e em operação no início do ano}\end{array}\right)} \%$$

Como estamos interessados em obter dados para uma situação de regime para cada ano a ser simulado, adotaremos uma taxa média. Não usaremos no denominador o número de programas médio no ano, pois precisamos considerar o fato de que logo após a entrada em operação os Sistemas de Informação apresentam baixíssima probabilidade de que venham a necessitar de manutenções, pelo menos nos seus seis primeiros meses (estimativas internas do Setor). Desta forma, faz-se necessário considerar um certo efeito de "retardamento". Assumimos então que, se defasássemos em menos meio ano, o intervalo de tempo a ser tomado como base para o cálculo do número médio de programas instalados e em operação, estaríamos considerando com razoável precisão tal efeito.

Assim, se admitirmos que seja razoável supor que o total de programas em produção (instalados e em operação) aumente linearmente ao longo de cada ano, vemos pela figura 5.13, que o denominador para o cálculo da taxa média de manutenção pode ser aproximado pelo número de programas em produção no início do ano a que se refere, dando o exemplo do ano de 1984.

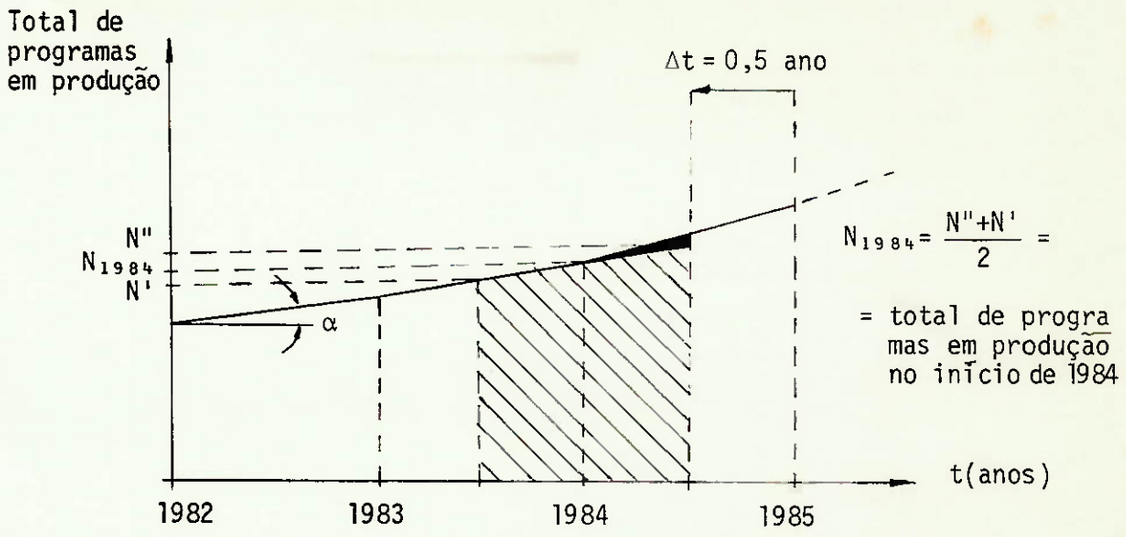


Figura 5.13 - Evolução do total de programas em produção ao longo dos anos

-e.p.a. -

Na realidade, como a inclinação da curva (α) aumenta de ano para ano (aumento da capacidade de produção), cometemos assim, um pequeno erro, que na figura é apresentado pela área em escuro. Entretanto, trata-se de uma diferença muito pequena e será desprezada.

Para se calcular essa taxa de manutenção, com base nos dados disponíveis (ano de 1984) e posteriormente utilizá-la para a projeção do número de programas a receberem manutenção em cada ano do período de simulação, foi necessário colher informações para se ter uma idéia de como o total de programas em produção aumenta no tempo. Essas informações juntamente com alguns dados estimados pelo pessoal do Setor de Informática e Serviços de Teleprocessamento são dadas a seguir:

a) Além dos programadores, os analistas também produzem programas que são instalados e entram em produção. Estima-se que dos programas desenvolvidos mensalmente pelos analis

tas, uma média de 60 (sendo 15 do tipo "batch" e 45 "on-line") são efetivamente implantados.

b) Como já foi mencionado, cerca de 5% dos serviços produtivos da Programação, não são liberados com conclusão. Isso significa que dos programas novos desenvolvidos pelos programadores, 95% são efetivamente implantados.

c) Dos programas "batch" desenvolvidos, cerca de 10% destinam se à reposição de outros que já estão implantados (isso ocorre porque ainda existem programas "batch" muito antigos). Quando se coloca em produção esses programas, a respectiva versão mais antiga é substituída e retirada da biblioteca. Assim dos programas "batch" que são implantados, 90% contribuem para um real acréscimo no total de programas em produção.

d) Os programas "on-line" em produção, ao contrário dos "batches", são todos relativamente novos, pois passaram a ser implantados há cerca de 3 anos. Portanto, os programas "on-line" desenvolvidos para reposição, praticamente inexistem. Pode se considerar que a totalidade dos programas "on-line" que são produzidos, contribuem para um real acréscimo.

Verificamos, então, que, no fim do mês de setembro de 1984, estavam em produção 6.785 programas, sendo 4.244 do tipo "batch" e 2.541 "on-line"(1). Considerando-se, então, as informações dadas por a a d e os dados de produção de programas novos da Programação (vide tabela 4.3), estimamos o total de programas dos tipos "batch" e "on-line" que estavam em produção no início de 1984 e o número de manutenções de programas de ambos os tipos a serem executados durante o mesmo ano. Com esses valores obtivemos as seguintes taxas médias de manutenção:

---

(1) Isso foi feito consultando-se o cadastro das bibliotecas (arquivos em disco) onde estão armazenados os programas de produção.

$$\left(\begin{array}{c}\text{taxa média de manutenção}\\ \text{anual para programas "batch"}\end{array}\right) = \frac{1.519}{3.759} = 40,41\% \ ,$$

$$\left(\begin{array}{c}\text{taxa média de manutenção}\\ \text{anual para programas "on-line"}\end{array}\right) = \frac{2.012}{1.844} = 109,11\% \ .$$

Observe-se que dos programas "batch" cerca de 4 programas em cada 10, requisitam algum tipo de manutenção do pessoal da Programação no espaço de um ano. Essa média sobe para mais de 10 (tem programas que exigem mais de uma manutenção por ano) para os programas "on-line".

Para se projetar o número de manutenções a serem realizadas em cada ano do horizonte de simulação, foi necessário antes estimar o total de programas em produção no início desses anos com base nas informações dadas em a a d . Os resultados obtidos constam na tabela 5.11. O número de programas a ser instalado em cada ano (acréscimo real) foi obtido consultando-se as projeções de produção de programas apresentadas na tabela 5.8 e supondo que a produção que cabe aos analistas também cresça 10% ao ano, acumulativamente. No número de programas desenvolvidos pela Programação já estão descontados os 5% dos que não chegam a ser concluídos. Observe-se que o acréscimo de programas "batch" não corresponde ao total de programas a serem instalados, pois já descontamos os 10% que são destinados à reposição. Consideramos que dos programas desenvolvidos em outras linguagens metade seja do tipo "batch" e a outra metade "on-line".

Com base, então, nesses totais de programas em produção e nas taxas médias de manutenção elaboramos a tabela 5.12 onde é apresentado o número de manutenções a serem executados em cada ano.

Para se definir como essas manutenções iriam-se distribuir supomos simplesmente que:

ANO	NÚMERO DE PROGRAMAS INSTALADOS E EM OPERAÇÃO NO INÍCIO DO ANO CONFORME O TIPO		NÚMERO DE PROGRAMAS "BATCH" A SER INSTALADO DURANTE O ANO CONFORME A ORIGEM		ACRÉSCIMO TOTAL DE PROGRAMAS "BATCH" NO ANO	NÚMERO DE PROGRAMAS "ON-LINE" A SER INSTALADO DURANTE O ANO CONFORME A ORIGEM		ACRÉSCIMO TOTAL DE PROGRAMAS "ON-LINE" NO ANO
	"Batch"	"On-Line"	Programação	Análise		Programação	Análise	
1984	3.759	1.844	539	180	647	390	540	930
1985	4.406	2.774	593	198	712	428	594	1.022
1986	5.118	3.796	653	218	784	472	653	1.125
1987	5.902	4.921	718	240	862	519	719	1.238
1988	6.764	6.159	790	264	949	570	791	1.361
1989	7.713	7.520	868	290	1.042	870	1.158	1.498

Tabela 5.11 - Tabela utilizada para se estimar o número de programas em produção no início de cada ano

- e.p.a. -

ANO	NÚMERO DE MANUTENÇÕES A SEREM EXECUTADAS CONFORME O TIPO DE PROGRAMA	
	"Batch"	"On-Line"
1985	1.780	3.027
1986	2.068	4.142
1987	2.385	5.369
1988	2.734	6.720
1989	3.116	8.205

Tabela 5.12 - Projeção do número de manutenções a serem executadas

- e.p.a. -

- os programas "batch" enviados para manutenção conservariam a proporção de 44% para programas de atualização e 56% para os de relatório (vide tabela 4.3).

- os programas "on-line" enviados para manutenção conservariam a proporção de 20% para programas desenvolvidos com SCOA e 80% sem SCOA (vide tabela 4.3).

- na dificuldade de se reconhecer entre os programas em produção aqueles que foram desenvolvidos em outras linguagens, admitimos que o número de manutenções com esse tipo de programa seria 2,86% (dado da situação atual) do total de programas instalados no início de cada ano (sua participação é quase insignificante).

Partindo dessas suposições elaboramos a distribuição dos serviços de manutenção como é mostrado na tabela 5.13.

ANO / TIPO DE PROGRAMA	1985	1986	1987	1988	1989
"batch" de atualização	783	910	1.049	1.203	1.371
"batch" de relatório	997	1.158	1.336	1.531	1.745
"on-line" com SCOA	608	832	1.078	1.349	1.648
"on-line" sem SCOA	2.419	3.310	4.291	5.371	6.557
em outras linguagens (não COBOL)	205	225	309	369	435
TOTAL	5.012	6.465	8.063	9.823	11.756

Tabela 5.13 - Projeção dos serviços de manutenção a serem executados por tipo e por ano
- e.p.a. -

Então, analogamente ao procedimento utilizado para se obter os dados para a geração de programas novos, assumimos que as chegadas de manutenções possam ser representadas por uma distribuição de Poisson e efetivamos os ajustes necessários para a obtenção do tempo médio entre chegadas para o grupo médio de programação.

As projeções da taxa média de chegadas e do tempo médio entre chegadas para o grupo médio constam, respectivamente, nas tabelas 5.14 e 5.15.

C. Dados para a geração de férias e outras atividades

Esses dados são função do número de programadores, ou seja, do tamanho do grupo médio. Não sabíamos, a priori,

TIPO DE PROGRAMA \ ANO	1985	1986	1987	1988	1989
"batch" de atualização	10,88	12,64	14,57	16,71	19,04
"batch" de relatório	13,85	16,08	18,56	21,26	24,24
"on-line" com SCOA	8,44	11,56	14,97	18,74	22,89
"on-line" sem SCOA	33,60	45,97	59,60	74,60	91,07
em outras linguagens (não COBOL)	2,85	3,54	4,29	5,13	6,04
TOTAL	69,62	89,79	111,99	136,43	163,28

Tabela 5.14 - Projeção da taxa média mensal de chegadas de manutenções por tipo para cada ano (para o grupo médio)
- e.p.a. -

TIPO DE PROGRAMA \ ANO	1985	1986	1987	1988	1989
"batch" de atualização	16,8	14,5	12,6	11,0	9,6
"batch" de relatório	13,2	11,4	9,9	8,6	7,6
"on-line" com SCOA	21,7	15,8	12,2	9,8	8,0
"on-line" sem SCOA	5,4	4,0	3,1	2,5	2,0
em outras linguagens (não COBOL)	64,3	51,7	42,6	35,7	30,3

Tabela 5.15 - Tempo médio entre chegadas de manutenções em horas úteis por tipo de programa para cada ano (para o grupo médio)
- e.p.a. -

qual deveria ser o número ideal de programadores a ser alocado ao grupo médio em cada ano do horizonte de simulação. A obtenção deste dado é um dos objetivos da simulação, pois no final pretendemos avaliar em quanto o Meta-Cobol poderia aliviar o efetivo da Programação, caso viesse a ser introduzido.

O tamanho ideal do grupo foi, então, procurado experimentalmente por tentativas. Em cada experimentação (simulação com determinado número de programadores num determinado ano) o tempo médio entre as chegadas de férias e o tempo médio entre as chegadas das outras atividades, foram consultados na tabela 5.16. Os dados dessa tabela também supõem que as chegadas possam ser representadas por uma distribuição de Poisson. Os tempos médios entre chegadas de outras atividades foram obtidos tomando-se como base a atual taxa média de chegadas de 7,6 outras atividades por mês e por programador.

NÚMERO DE PROGRAMADORES NO GRUPO MÉDIO	TEMPO MÉDIO ENTRE AS CHEGADAS DE OUTRAS ATIVIDADES (horas úteis)	TEMPO MÉDIO ENTRE AS CHEGADAS DE FÉRIAS (horas úteis)
7	34	3.137
8	30	2.745
9	27	2.440
10	24	2.196
11	22	1.996
12	20	1.830
13	18	1.689
14	17	1.569
15	16	1.464
16	15	1.372
17	14	1.292
18	13	1.220
19	13	1.156
20	12	1.098

Tabela 5.16 - Dados para as gerações de outras atividades e de férias conforme o tamanho da equipe
- e.p.a. -

D. Dados para o estabelecimento das condições iniciais

A simulação de cada ano deverá ser iniciada já com um certo número de programas em mãos dos programadores. Baseando-nos no cenário descrito, consideraremos que a Programação continuando a trabalhar sob pressão e com o mesmo esquema de atendimento, deverá manter os atuais valores do fator q (vide item 5.5.) que é de 1,40 para os programas novos e 0,58 para os programas destinados à manutenção. Assim, com base na projeção da média de programas a serem atendidos por mês (vide tabelas 5.9 e 5.14) obtivemos a média de programas em mãos para manutenção e para desenvolvimento, como mostra a tabela 5.17, lembrando que:

$$(\text{Média de programas em mãos}) = q \cdot (\text{média de programas atendidos por mês})$$

Como estamos interessados na situação de regime de cada ano, iniciaremos as simulações com esses valores médios. Como o início de um ano corresponde ao fim do ano imediatamente anterior, consideraremos que os programas em mãos na situação inicial, se distribuam entre os vários tipos, conforme o

ANO	FATOR q		MÉDIA DE PROGRAMAS A SEREM ATENDIDOS MENSALMENTE		MÉDIA DE PROGRAMAS EM MÃOS	
	Programa Novo	Manutenção	Programa Novo	Manutenção	Programa Novo	Manutenção
1985	1,40	0,58	14,9	69,6	21	40
1986	1,40	0,58	16,4	89,8	23	52
1987	1,40	0,58	18,1	112,0	25	65
1988	1,40	0,58	19,9	136,4	28	79
1989	1,40	0,58	21,9	163,3	31	95

Tabela 5.17 - Média de programas novos e manutenções em mãos para cada ano

- e.p.a. -

perfil dos programas atendidos nesse ano anterior. Como vimos, para os programas novos, esse perfil continuará a ser o mesmo da atual situação, pois a capacidade de produção deverá ser aumentada uniformente. Já os programas de manutenção, devido às suposições assumidas na projeção do número de serviços de manutenção a serem realizados, a cada ano se distribuirá de forma diferente. É o que mostra a tabela 5.18, elaborada com base nos dados das tabelas 4.3, 5.8 e 5.14.

TIPO DE SERVIÇO E ANO / TIPO DE PROGRAMA	PROGRAMA NOVO	PROGRAMA PARA MANUTENÇÃO				
	1985 a 1989	1985	1986	1987	1988	1989
"batch" de atualização	28,3	18,1	15,6	14,1	13,0	12,2
"batch" de relatório	28,2	23,1	19,9	17,9	16,6	15,6
"on-line" com SCOA	19,6	11,0	12,1	12,9	13,4	13,7
"on-line" sem SCOA	20,7	43,5	48,3	51,2	53,2	54,7
em outras linguagens (não COBOL)	3,2	4,3	4,1	3,9	3,8	3,8

Tabela 5.18 - Distribuição relativa (em %) dos programas novos e para manutenção em mãos na situação inicial
- e.p.a. -

E. Tempo efetivo para o desenvolvimento de programas "batch" usando Meta-Cobol

O Meta-Cobol, caso venha a ser implantado, deverá realmente constituir uma poderosa ferramenta de "software" e possibilitar que os programas "batch" sejam desenvolvidos em prazos mais curtos. Como, no entanto, ainda não passa de um ante-projeto, essa redução que se espera com o seu aproveita

mento teve de ser estimado subjetivamente. Para isso contamos com as estimativas na base do "sentimento" do pessoal envolvido no seu ante-projeto, pois como são os idealizadores dessa ferramenta, são aqueles mais indicados para emitir opiniões a seu respeito.

As pessoas consultadas foram unânimes em afirmar que o ganho de produtividade a se esperar do Meta-Cobol dependerá das características do programa a ser desenvolvido. Supondo que, em geral, quanto maior a complexidade do programa, maior seja o tempo necessário para se desenvolvê-lo, as pessoas consultadas dividiram os programas "batch" em 3 categorias, quais sejam fácil, médio e difícil (segundo critérios próprios), e, com base nas distribuições de freqüência do tempo efetivo de desenvolvimento de programas "batch" de atualização e de relatório (vide Anexo I), associaram à cada uma delas um conjunto de classes de freqüência. Com isso, estimaram o mínimo e o máximo tempo efetivo (intervalo de tempo) que se levaria para desenvolver os programas de cada categoria. Feito isso , para cada categoria estimaram a redução que se espera com o aproveitamento do Meta-Cobol. As estimativas feitas pelo supervisor que tem orientado o desenvolvimento do ante-projeto, para o caso dos programas "batch" de atualização, são apresentadas como exemplo na tabela 5.19. Calculando-se a média das estimativas constantes nesta tabela, com base nos valores de freqüência relativa da amostra, encontramos o tempo efetivo médio estimado para esse caso que é de 7,91 horas úteis. Como na situação atual a média é de 41,43 horas úteis, o Meta-Cobol proporcionaria segundo esta estimativa, uma redução da ordem de 81%, ou seja, em média, os programadores passariam a dispender somente 19% do tempo que se gastaria normalmente. Calculamos as médias das estimativas feitas pelas demais pessoas consultadas e incluindo as feitas pelo supervisor, verificamos que tanto para os programas de atualização como os de relatório, passariam a levar em média, entre 16 e 25% do tempo atualmente dispendido para o seu desenvolvimento.

CATEGORIA DOS PROGRAMAS (grau de complexidade)	TEMPO EFETIVO DE DESENVOLVIMENTO (horas úteis)	FREQÜÊNCIA RELATIVA (%)	REDUÇÃO ESTIMADA COM O META-COBOL	TEMPO EFETIVO ESTIMADO COM O META-COBOL (horas úteis)
FÁCIL	9	9,5	$\frac{1}{10}$	0,9
	18	15,9	$\frac{1}{10}$	1,8
	27	14,3	$\frac{1}{10}$	2,7
	36	14,3	$\frac{1}{10}$	3,6
MÉDIO	45	14,3	$\frac{1}{5}$	9,0
	54	12,7	$\frac{1}{5}$	10,8
	63	6,3	$\frac{1}{5}$	12,6
	72	3,2	$\frac{1}{5}$	14,4
	81	3,2	$\frac{1}{5}$	16,2
DIFÍCIL	90	1,6	$\frac{1}{3}$	30,0
	99	1,6	$\frac{1}{3}$	33,0
	108	1,6	$\frac{1}{3}$	36,0
	117	1,6	$\frac{1}{3}$	39,0

Tabela 5.19 - Estimativas para a redução nos tempos efetivos de desenvolvimento para programas "batch" de atualização
- e.p.a. -

A consideração da variação dessa redução conforme a complexidade do programa, seria muito trabalhosa de se embutir no programa. Adotamos, então, o valor de 80% como o percentual de redução para o tempo efetivo de desenvolvimento de programas "batch" em geral. A generalização desse percentual facilita em muito a adaptação do programa de simulação, e, embora, implique em desconsiderar o efeito do grau de complexidade do programa sobre a aplicabilidade do Meta-Cobol, respeita ao menos, a média das estimativas, o que é razoável quando se trata de simulação envolvendo extensos intervalos de tempos e se pretende obter resultados globais a longo prazo.

Este procedimento será adotado para a simulação dos anos de 1986 a 1989. O ano de 1985 constituirá uma exceção, pois como vimos no subitem 5.6.1. a versão final do Meta-Cobol só poderia entrar em operação a partir de 1986. As versões que seriam entregues durante o segundo semestre de 1985 ao uso na Programação, não proporcionariam tantos ganhos como a versão final, mas seria mais poderosa que a primeira versão a ser testada em ambiente menor. As pessoas consultadas estimaram que em média, conseguiria-se reduções da ordem de 60% no tempo efetivo de desenvolvimento com essas versões.

Para se simular então, o ano de 1985, poder-se-ia dividí-lo em duas etapas (2 semestres) sendo que no primeiro, manter-se-ia os atuais tempos efetivos de desenvolvimento de programas "batch" e no segundo, considerar-se-ia o uso do Meta-Cobol possibilitando que os tempos efetivos de desenvolvimento de programas "batch" se reduzissem à 40% do que se levaria normalmente. No entanto, optamos por considerar esta particularidade do ano de 1985 de maneira mais simples. Recorremos ao artifício de se simulá-lo ininterruptamente, considerando uma situação intermediária, ou seja, fazendo com que os programas "batch" passassem a ser desenvolvidas com tempos efetivos 30% menores (média entre redução de 0% no primeiro semestre e 60% no segundo semestre). Esperamos assim, obter resultados globais médios muito mais facilmente.

5.6.4. AJUSTES NO PROGRAMA DE SIMULAÇÃO

Os ajustes que se fizeram necessários para as simulações do futuro são basicamente de 3 tipos, quais sejam:

A. Alimentação de Dados Referentes a Cada Ano que Independem do Tamanho do Grupo

Corresponde à alimentação dos dados para a geração de programas novos, geração de manutenções e estabelecimento das condições iniciais, que para um determinado ano, independentemente da situação a ser simulada (com ou sem o Meta-Cobol)e do número de programadores no grupo médio, são sempre os mesmos.

B. Alimentação de Dados que Dependem Apenas do Número de Programadores

Corresponde à alimentação dos dados para a geração de outras atividades e de férias que é função apenas do número de programadores. Esses ajustes são indispensáveis para se determinar em cada ano e para cada situação simulada, o tamanho adequado do grupo médio.

C. Alteração na Lógica do Programa para Prever Desenvolvimento de Programas "batch" com o Meta-Cobol

Para a simulação do futuro contando com o aproveitamento do Meta-Cobol a lógica do modelo recebeu alguns passos adicionais de modo que as transações referentes ao desenvolvimento de programas "batch", passassem a contar com as reduções estimadas no subitem 5.6.3. no seu tempo efetivo de atendimento.

O programa utilizado para a simulação do ano de 1985 contando com a introdução do Meta-Cobol, e que serviu também de base para as simulações dos anos de 1986 e 1987, segue a título de exemplo no Anexo V . Para que a coleta dos resultados fosse agilizada, aproveitamos neste trabalho um interessante recurso do GPSS que é a possibilidade de se redefinir dinamicamente, instruções ou declarações de um programa, durante o seu próprio processamento. Após a simulação especificada pelo programa original, o GPSS verifica se há comandos de redefinição do programa e cartões de controle para a continuação da simulação na seqüência. Caso positivo, ele trata de reconfigurar o programa e processar na seqüência mais uma simulação. O programa que consta no Anexo V , por exemplo, foi preparado para simular consecutivamente os anos de 1985, 1986 e 1987, e fornecer os respectivos resultados. Só não foi possível simular todos os 5 anos na seqüência, por necessitar de alocação excessiva de memória de computador. Os anos de 1988 e 1989 foram simulados juntos por meio de um outro programa que é análogo ao que consta no Anexo V .

A mesma técnica foi adotada para a simulação do futuro sem contar com o Meta-Cobol, utilizando o mesmo programa utilizado na validação do modelo (vide Anexo II) como base.

5.7. PROCESSAMENTO DAS SIMULAÇÕES E INTERPRETAÇÃO DOS RESULTADOS

Neste estudo, fizemos simulações com 3 finalidades distintas quais sejam:

a) Calibração e validação do modelo.

b) Avaliação do impacto da introdução do Meta-Cobol.

c) Análise de sensibilidade dos resultados referentes ao impacto da introdução do Meta-Cobol.

No item 5.5, já tratamos das simulações que se fizeram necessárias para se alcançar a finalidade dada em a. As simulações que se referem ao item b, são as que mais nos interessam, pois além de proporcionarem uma idéia do impacto (repercussões) que se verificaria no ambiente da Programação com a introdução do Meta-Cobol, fornecem dados fundamentais para o estudo de viabilidade econômica do desenvolvimento dessa ferramenta. O processamento dessas simulações é tratado no subitem 5.7.1. e os seus resultados, interpretados no subitem 5.7.2.

Finalmente, nos subitens 5.7.3. e 5.7.4., tratamos, respectivamente, das simulações processadas com o objetivo dado por c, e dos seus resultados, que são indispensáveis para se ter uma idéia dos limites de validade dos resultados obtidos das simulações para a avaliação do impacto da introdução do Meta-Cobol.

5.7.1. PROCESSAMENTO DAS SIMULAÇÕES PARA A AVALIAÇÃO DO IMPACTO DA INTRODUÇÃO DO META-COBOL

Trataremos agora dos processamentos que foram realizados para as simulações do futuro com o objetivo de se avaliar as modificações advindas do uso do Meta-Cobol.

Dadas as previsões de crescimento de programas novos e manutenções para os próximos anos (horizonte de vida previsto para o Meta-Cobol), optamos por simular a cada ano vindouro deste horizonte, os trabalhos do grupo médio de programação, para avaliar quantos programadores seriam necessários neste grupo se se mantivesse os atuais tempos de atendimento das atividades e as ferramentas de apoio disponíveis (continuar sem o "Meta-Cobol") ou se se introduzisse o Meta-Cobol.

Elaboramos, então, alguns critérios que definissem o tamanho (número de programadores) adequado ao grupo em ambas as situações a cada ano simulado. Estes critérios são:

a) Média de programas liberados por mês: O sistema simulado, tal como o real, deve se manter aproximadamente em regime. Assim sendo, a média de liberação de programas deve se equilibrar com a média de recepção de programas (chegadas de programas), ou seja, a relação (tomando-se 1 mês como base)

$$\frac{\text{(média de programas recebidos)}}{\text{(média de programas liberados)}}$$

deverá tender a 1.

b) Parcela do tempo útil ocioso: Vimos no cenário que se pretende manter os programadores trabalhando sob pressão. Assim sendo, a ociosidade deve ser mínima, tendendo a um porcentual nulo do tempo útil corrido, como ocorre na situação atual.

c) Média de programas em mãos: Esta variável está associada à anterior. Em decorrência também da tática de se manter os programadores trabalhando sob pressão, o número médio de programas em mãos deve se manter relativamente alto, durante o período de simulação. Adotamos como o valor de referência para cada ano (vide tabela 5.20), o total de programas em mãos do grupo médio na situação inicial pois, como vimos (vide subitem 5.6.3.), estes valores foram estimados

ANO	MÉDIA DE PROGRAMAS EM MÃOS
1985	61
1986	75
1987	90
1988	107
1989	126

Tabela 5.20 - Média de programas em mãos estimada para cada ano
- e.p.a. -

para uma situação de regime e respeitando os valores do fator q (quociente da média de programas em mãos pela média de programas liberados) que se têm verificado na prática. A média de programas em mãos é a soma dos valores da média de programas para desenvolvimento e da média de programas para manutenção, em mãos (vide subitem 5.6.3.).

d) Dispêndio do tempo útil em outras atividades: A verificação do dispêndio de tempo útil em outras atividades, ou melhor, do aproveitamento do tempo útil corrido, excluindo o tempo gasto em férias e o perdido em ociosidade, entre as atividades produtivas e as chamadas outras atividades (vide subitem 2.2.1.) pode ser feita através do quociente

$$p = \frac{(\text{tempo útil gasto em outras atividades})}{(\text{tempo útil gasto no desenvolvimento de programas}) + (\text{tempo útil gasto em manutenções de programas}) + (\text{tempo útil gasto em outras atividades})}$$

Segundo os dados obtidos através das Folhas de Acompanhamento da Programação este quociente tem girado em torno de 20% e, conforme o cenário, tal nível de aproveitamento deverá ser mantido.

Convém observar que a ociosidade será medida em função do tempo útil corrido total, mas o dispêndio de tempo em outras atividades não. Esse dispêndio, quando medido também em função do tempo útil corrido total, deveria se situar abaixo dos 20% que se registraria para o quociente p.

Não adotamos margens de desvios toleráveis (desvio percentual em relação aos valores reais esperados) para a verificação desses resultados, pois nada impede que o número de programadores adequado para o grupo médio em cada ano, seja um valor não inteiro. E o modelo, da forma como foi elaborado, não tem condições de considerar "capacidade" não inteira (número fracionário de "postos de atendimento"). Então, mesmo que adotássemos as mais criteriosas margens de desvios possíveis, provavelmente, não seria possível obter para nenhum caso (simulação de uma das situações, para um determinado ano, com um determinado número de programadores) os resultados desejados já que é muito mais provável que o número ideal não seja inteiro.

Entretanto, o fato de que o número mais adequado não possa ser inteiro, não poderia ser desprezado já que, por exemplo, 0,5 programador para o grupo médio, respeitando a escala de 1 para 6 em relação ao conjunto total formado pelos grupos da Programação, representaria para este um total de 3 programadores o que, evidentemente, não é tão insignificante.

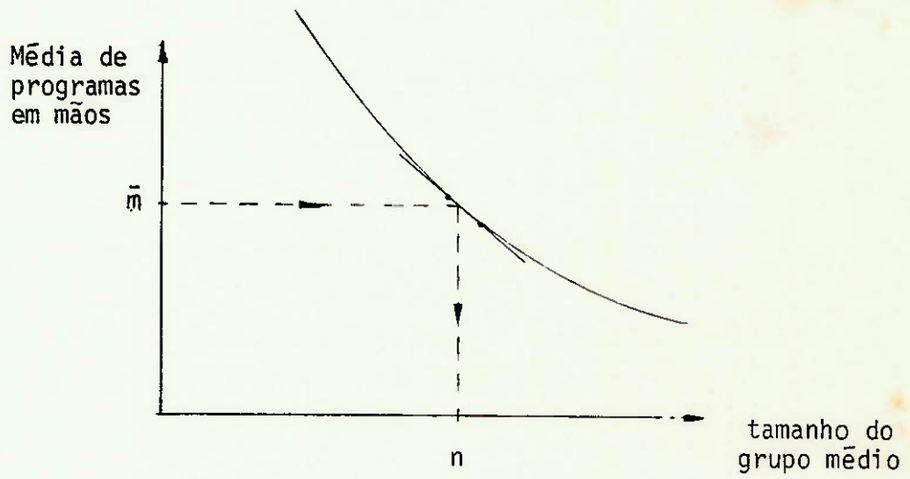
Para se contornar este problema, adotamos o seguinte procedimento:

a) Processamos simulações, para cada ano e cada situação, experimentando números diferentes de programadores para o grupo médio, até obter 2 tamanhos que proporcionassem resultados próximos aos que foram enumerados (a, b, c e d do subitem 5.7.1.). O número de programadores adequado para o grupo médio, não pre

cisa estar, necessariamente, entre os valores correspondentes aos dos resultados escolhidos.

b) Estimamos o número adequado de programadores para cada ano de cada situação, supondo que a variável "média de programas em mãos" fosse uma função do tamanho do grupo médio. Assim, a partir dos relatórios das simulações seria possível determinar dois pontos da curva dessa função. Então, admitindo ser razoável aproximar o trecho nas imediações (inclusive extrapolando o intervalo) desses pontos por uma reta, obtivemos o número de programadores adequado com base na média real de programas em mãos estimada para o ano em questão. A figura 5.14 ilustra este passo do procedimento.

Gráfico da média de programas em mãos em função do tamanho do grupo médio para um determinado ano de uma situação



$\bar{m}$ = média real estimada

n = número de programadores adequado

Figura 5.14 - Ilustração da técnica utilizada para a estimação do número de programadores adequado

- e.p.a. -

A média de programas em mãos, foi escolhida como o parâmetro a servir de base para a estimação do número adequado de programadores, pois é um dado fornecido diretamente pelos programas e porque dentre as variáveis envolvidas em a, b, c e d do subitem 5.7.1. é aquela que com maior evidência sofre influência da variação do tamanho do grupo (aumenta quando o número de programadores é estimado para menos, e diminui quando este é superestimado), oferecendo ainda, um valor de referência (média real estimada) objetivo.

Apresentamos na tabela 5.21 os tamanhos que foram experimentados para cada ano de cada situação, cujos resultados serão apresentados mais adiante. Para cada um desses casos, executamos apenas uma única simulação, e portanto, obtivemos apenas uma coleção de resultados. O ideal seria obter amostras em vez de pontos isolados para cada caso. Mas, evidentemente, isso não seria exeqüível dado o grande número de casos experimentados (vide tabela 5.21). Entretanto, como adotamos um extenso intervalo de tempo simulado, acreditamos que os resultados obtidos estejam próximos da média e, portanto, que seja razoável aproveitá-los na estimação do tamanho adequado do grupo.

SITUAÇÃO / ANO	FUTURO SEM O META-COBOL	FUTURO COM O META-COBOL
1985	8 e 9	8 e 9
1986	9 e 10	8 e 9
1987	12 e 13	9 e 10
1988	14 e 15	12 e 13
1989	16 e 17	13 e 14

Tabela 5.21 - Combinações de número de programador, situação e ano experimentados

- e.p.a. -

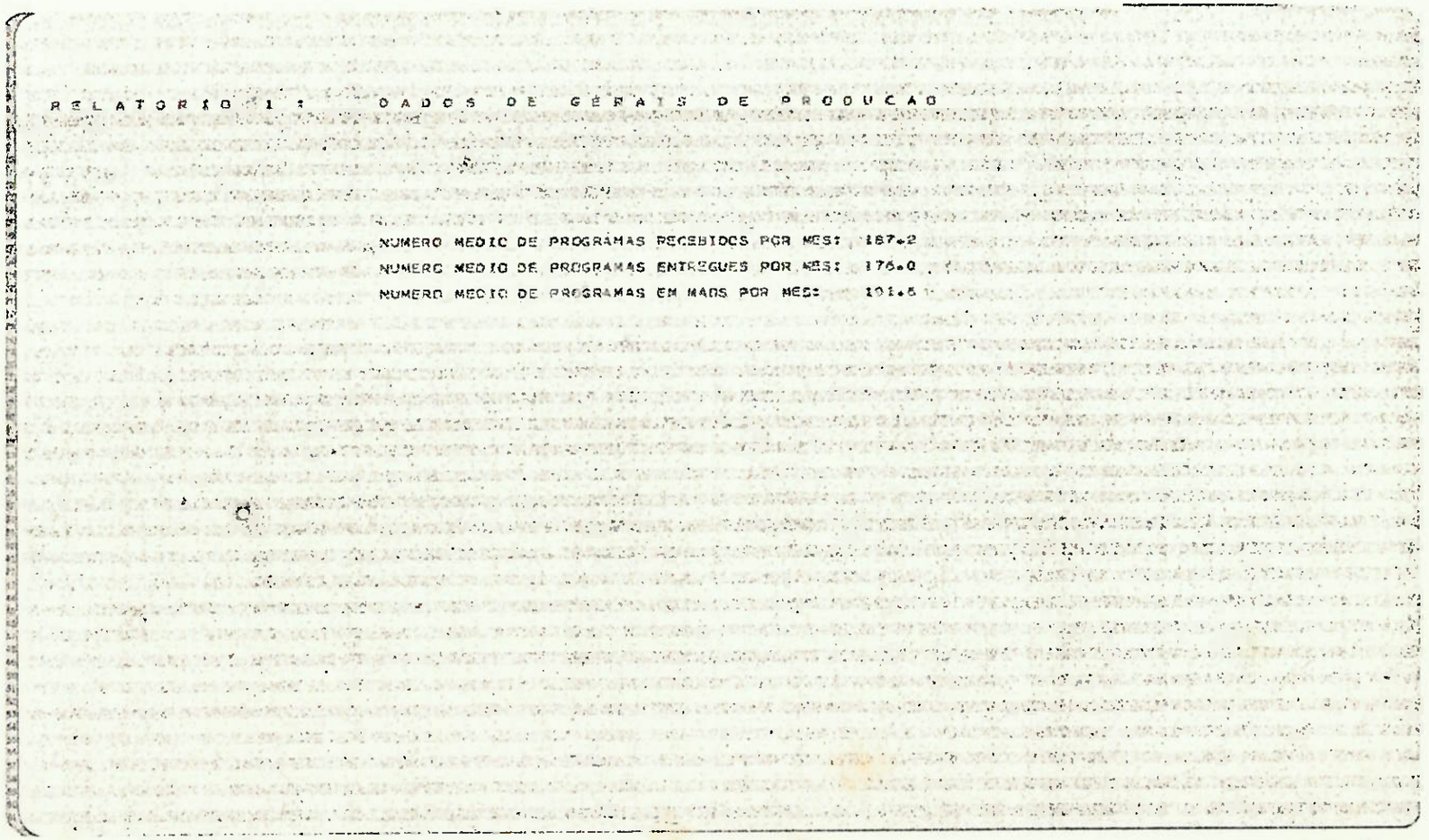
5.7.2. RESULTADOS PARA A AVALIAÇÃO DO IMPACTO DA INTRODUÇÃO DO META-COBOL

Apresentamos nas tabelas 5.22 e 5.23 (futuro com e sem o Meta-Cobol), os dados globais de desempenho do grupo médio, obtidos da simulação com os dois tamanhos que serviram de base para a estimação do número de programadores adequado para cada ano de cada situação. Observe-se que os resultados obtidos estão muito próximos das condições enumeradas no subitem 5.7.1.

Pela consistência dos resultados globais apresentados nas tabelas 5.22 e 5.23, pode-se dizer que o modelo está ajustado às condições futuras determinadas pelo cenário elaborado, e que portanto, os resultados obtidos das simulações são suficientemente confiáveis para prosseguirmos o estudo.

Seguem nas páginas seguintes, os principais relatórios com dados globais, retirados de uma das 10 listagens (dois tamanhos para cada ano) obtidas do processamento das simulações do futuro com o Meta-Cobol. Estes relatórios referem-se ao desempenho do grupo médio em 1989 com 13 programadores. Para os demais anos desta situação, escolhemos os principais relatórios de um dos tamanhos simulados e preparamos o Anexo VI . Das listagens obtidas da simulação do futuro sem o Meta-Cobol, também separamos os principais relatórios de um dos tamanhos experimentados em cada ano e preparamos o Anexo VII. Os relatórios originais obtidos para todos esses casos, bem como para os casos não incluídos nos Anexos VI e VII, estão todos organizados no volume em anexo com as listagens originais.

Antes de seguirmos para a análise dos resultados obtidos, verificamos rapidamente se os módulos de geração de transações funcionaram conforme as especificações dos tempos médios entre chegadas levantadas no subitem 5.6.3., e se pelo menos os tempos efetivos de atendimento para o desenvolvimento de programas "batch" com o Meta-Cobol também estavam próximos das suas especificações. Isso foi feito para nos certificarmos de que todos os dados de entrada do modelo, haviam sido



RELATORIO 1 : DADOS DE GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 187.2
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 176.0
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 101.5

RELATÓRIO COM OS DADOS GERAIS DA SIMULAÇÃO DO ANO DE 1989, CONTANDO COM O META-COBOL PARA O GRUPO MÉDIO COM 13 PROGRAMADORES

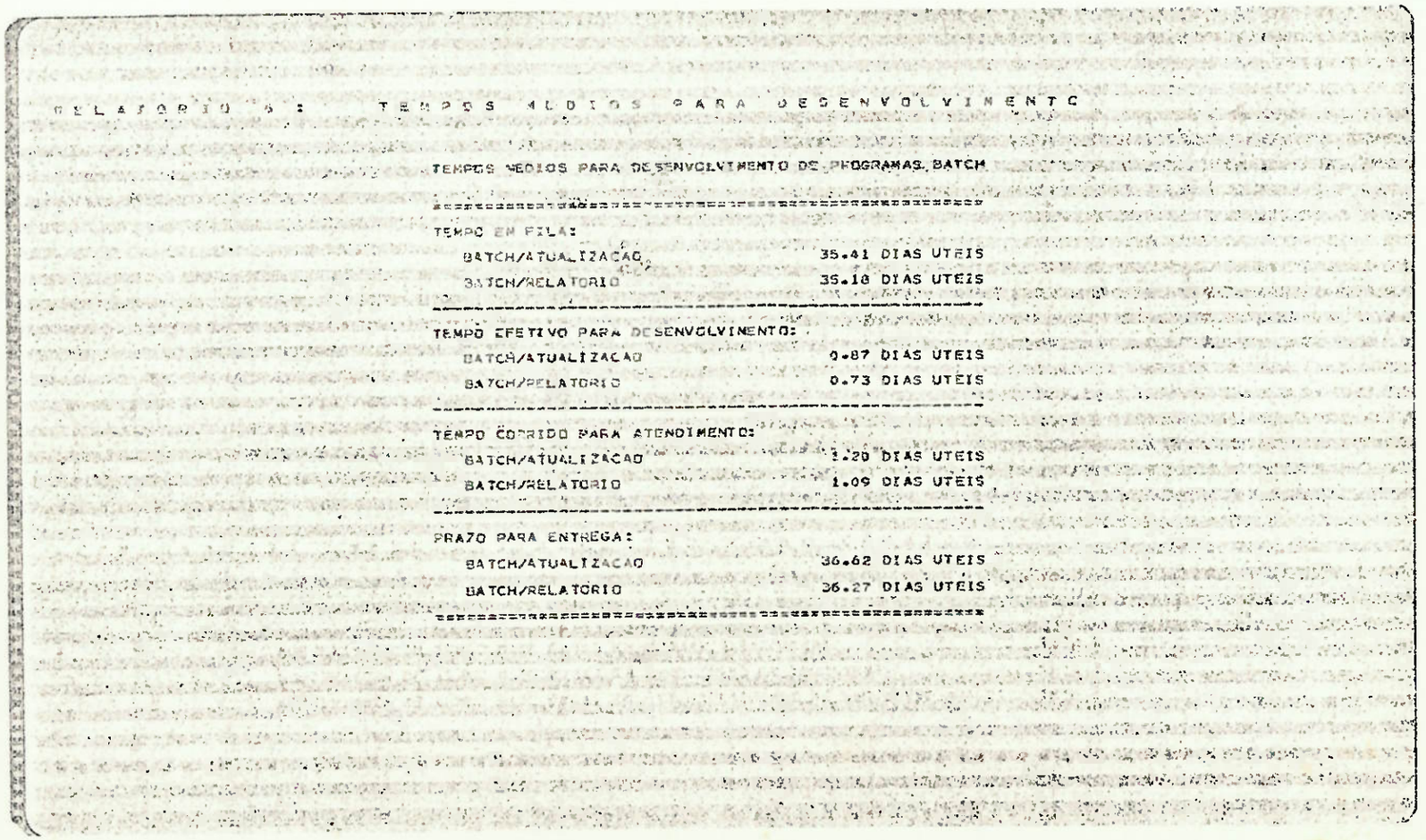
RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	14.9 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	65.5 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.5 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	13.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	60.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	18.1 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.4 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	0.6 %

RELATÓRIOS COM DADOS DE APROVEITAMENTO DO TEMPO ENTRE AS DIVERSAS ATIVIDADES PARA A SIMULAÇÃO DO ANO DE 1989, CONTANDO COM O META-COBOL PARA O GRUPO MÉDIO COM 13 PROGRAMADORES



RELATORIO A : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	35.41 DIAS UTEIS
BATCH/RELATORIO	35.18 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	0.87 DIAS UTEIS
BATCH/RELATORIO	0.73 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	1.20 DIAS UTEIS
BATCH/RELATORIO	1.09 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	36.62 DIAS UTEIS
BATCH/RELATORIO	36.27 DIAS UTEIS

RELATÓRIO COM OS TEMPOS ENVOLVIDOS NO DESENVOLVIMENTO DE PROGRAMAS "BATCH" PARA O ANO DE 1989, CONTANDO COM O META-COBOL PARA O GRUPO MÉDIO COM 13 PROGRAMADORES

ANO E TAMANHO EXPERIMENTADOS / VARIÁVEL	1985		1986		1987		1988		1989	
	8	9	8	9	9	10	12	13	13	14
ociosidade (% do tempo útil corrido total)	0,6	1,0	1,2	5,8	0,6	0,8	0,7	0,5	0,6	0,7
dispêndio de tempo em outras atividades (p%)	18,8	20,1	18,4	18,9	18,3	19,6	19,7	19,0	19,5	18,6
média de programas recebidos (programas/mês)	87,4	94,9	108,5	104,8	134,3	128,7	164,0	157,9	187,2	183,0
média de programas liberados (programas/mês)	86,6	98,6	105,0	109,7	121,6	127,0	153,7	165,4	176,0	186,1
$\left(\frac{\text{média de prog.recebidos}}{\text{média de prog.liberados}}\right)$	1,009	0,962	1,033	0,955	1,104	1,013	1,067	0,955	1,064	0,983
média de programas em mãos	79,9	29,9	102,1	24,8	168,3	77,5	165,6	71,4	191,6	69,5

Tabela 5.22 - Dados globais obtidos das simulações do futuro com o Meta-Cobol - e.p.a. -

ANO E TAMANHO EXPERIMENTADOS / VARIÁVEL	1985		1986		1987		1988		1989	
	8	9	9	10	12	13	14	15	16	17
ociosidade (% do tempo útil corrido total)	0,8	9,6	1,3	2,1	1,5	1,4	1,4	0,5	0,7	0,8
dispêndio de tempo em outras atividades (p%)	18,8	18,9	18,5	19,0	18,3	18,6	19,1	19,2	18,4	19,4
média de programas recebidos (programas/mês)	85,5	82,4	110,6	103,1	133,8	129,3	153,8	161,4	183,7	189,6
média de programas liberados (programas/mês)	82,1	86,0	95,0	105,1	130,1	135,8	154,5	163,7	183,7	190,0
$\left(\frac{\text{média de prog.recebidos}}{\text{média de prog.liberados}}\right)$	1,041	0,958	1,164	0,981	1,028	0,952	0,995	0,986	1,000	0,998
média de programas em mãos	90,8	20,8	191,5	37,6	112,4	32,0	101,9	86,0	137,7	126,7

Tabela 5.23 - Dados globais obtidos das simulações do futuro sem o Meta-Cobol
- e.p.a. -

devidamente alimentados ao programa, nas simulações realizadas. A tabela 5.24 apresenta a média dos valores obtidos com base nos resultados dos processamentos citados na tabela 5.23, os valores médios reais esperados e a discrepância percentual entre esses valores. Evidentemente, como o modelo é do tipo estocástico, os resultados não são exatamente o esperado, mas as discrepâncias observadas são razoavelmente baixas. Se cada simulação tivesse durado um tempo simulado muito maior, naturalmente, as discrepâncias tenderiam a zero.

Relembrando que no nosso caso, o período de tempo simulado está limitado a 1 ano, e observando-se os dados da tabela 5.24 acreditamos que contamos com evidências suficientes de que os dados estimados no subitem 5.6.3. foram devidamente alimentados no programa e que o comportamento do sistema durante a simulação foi fiel à essas especificações.

O único resultado, um pouco mais longe, obtido dessas simulações é o percentual do tempo útil corrido gasto em férias que em média foi de 5,46% (os valores considerados na média são encontrados nos Relatórios de Aproveitamento do Tempo Corrido nas listagens em anexo). Se os programas tivéssem cumprido à risca as especificações de gerações de férias, esse percentual seria de 6,8%, portanto em média, tivemos uma discrepância de 19,7% a menos, nas simulações. Isso ocorreu porque o intervalo de simulação de 1 ano é extenso em relação à maioria dos tempos médios entre chegadas, mas não em relação às chegadas de férias (a diferença entre a ordem de grandeza desses valores não é muito grande). Se o intervalo de simulação fosse mais extenso, sem dúvida, o percentual obtido experimentalmente seria mais próximo dos valores esperados.

Apresentamos a seguir, as principais repercussões que se verificariam, caso o Meta-Cobol viésse a ser introduzido na Programação. A constatação desses efeitos só foi possível, graças à metodologia de estudo comparativo das duas situações futuras, que adotamos.

ANO	MÉDIA MENSAL DO TOTAL DE PROGRAMAS GERADOS			TEMPO EFETIVO EM HORAS ÚTEIS PARA O DESENVOLVIMENTO DE PROG. "BATCH" DE ATUALIZAÇÃO			TEMPO EFETIVO EM HORAS ÚTEIS PARA O DESENVOLVIMENTO DE PROG. "BATCH" DE RELATÓRIO		
	Média dos Resultados	Média Real Esperada	Discrepância (%)	Média dos Resultados	Média Real Esperada	Discrepância (%)	Média dos Resultados	Média Real Esperada	Discrepância (%)
1985	87,6	84,6	3,5	26,6	29,0	8,1	21,9	22,5	2,6
1986	106,8	106,2	0,6	8,1	8,3	1,9	6,8	6,4	5,5
1987	131,5	130,1	1,1	7,5	8,3	10,0	6,3	6,4	1,6
1988	159,3	156,3	1,9	8,1	8,3	3,0	6,4	6,4	0
1989	185,9	185,2	0,4	7,6	8,3	8,9	6,5	6,4	1,3

Tabela 5.24 - Média dos valores obtidos nas simulações em comparação com os valores médios reais esperados
- e.p.a. -

A. Redução do Efetivo Necessário à Programação nos Próximos Anos

Com base no procedimento descrito no subitem 5.7.1. calculamos qual seria o número de programadores adequado à Programação para o período compreendido entre 1985 e 1989. Esse número é 6 vezes maior ao que seria necessário ao grupo médio e se trata de uma média. A Programação não teria, necessariamente, de crescer aos saltos no início de cada ano para atingir essas médias, mas conforme as circunstâncias poderia começar cada ano com um efetivo menor que a média projetada e terminar com um efetivo maior, desde que se respeitasse a média do ano.

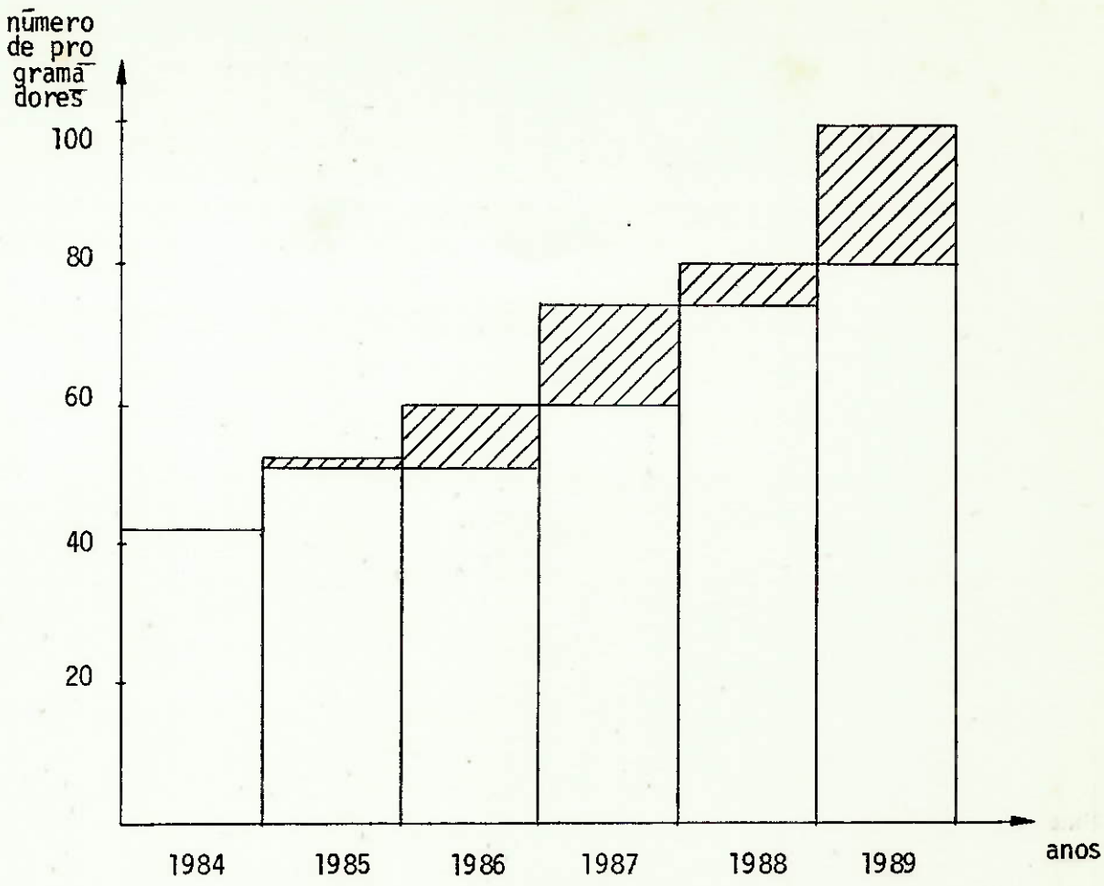
A tabela 5.25 apresenta os valores utilizados para se realizar esse dimensionamento do pessoal necessário à Programação e o número adequado de programadores para cada ano de cada situação. Observe-se que com a introdução do Meta-Cobol verificaria-se um significativo impacto em termos de redução do efetivo necessário ao conjunto formado pelos grupos da Programação. O Meta-Cobol não chegaria a reduzir o efetivo atual, mas possibilitaria ao Setor de Informática e Serviços de Teleprocessamento, contratar menos programadores nos próximos anos do que seria necessário se se continuar a desenvolver, convencionalmente, os programas "batch".

Observe-se pelos valores encontrados que, com a introdução do Meta-Cobol, a partir de 1986 o efetivo que seria necessário à Programação seria, aproximadamente, aquele que seria necessário um ano antes, a se permanecer sem o mesmo. Isto implica em dizer que, com o Meta-Cobol, conseguir-se-ia postergar de um ano (a partir de 1986) a projeção do número de programadores necessário para os próximos anos a se continuar desenvolvendo convencionalmente os programas "batch".

Esta tendência é apresentada pela figura 5.15 onde a área hachurada corresponde ao alívio que seria proporcionado pelo Meta-Cobol. No ano de 1985, o impacto de uma eventual introdução do Meta-Cobol não seria considerável, já que isso

ANO	SITUAÇÃO	MENOR DOS TAMANHOS TESTADOS E ESCOLHIDOS	MÉDIA DE PROGRAMAS EM MÃOS PARA O MENOR TAMANHO	MAIOR DOS TAMANHOS TESTADOS E ESCOLHIDOS	MÉDIA DE PROGRAMAS EM MÃOS PARA O MAIOR TAMANHO	MÉDIA REAL DE PROGRAMAS EM MÃOS ESTIMADO	TAMANHO ADEQUADO PARA O GRUPO MÉDIO	EFETIVO ADEQUADO PARA A PROGRAMAÇÃO
1985	Sem o Meta-Cobol	8	90,8	9	20,8	61	8,43	51
	Com o Meta-Cobol	8	79,9	9	29,9		8,38	50
1986	Sem o Meta-Cobol	9	191,5	10	37,6	75	9,76	59
	Com o Meta-Cobol	8	102,1	9	24,8		8,35	50
1987	Sem o Meta-Cobol	12	112,4	13	32,0	90	12,28	74
	Com o Meta-Cobol	9	168,3	10	77,5		9,86	59
1988	Sem o Meta-Cobol	14	101,9	15	86,0	107	13,68	82
	Com o Meta-Cobol	12	165,6	13	71,4		12,58	75
1989	Sem o Meta-Cobol	16	137,7	17	126,7	126	17,06	102
	Com o Meta-Cobol	13	191,6	14	69,5		13,54	81

Tabela 5.25 - Valores utilizados para a determinação do efetivo adequado à Programação
- e.p.a. -



diferença entre o número de programadores necessários sem a introdução do Meta-Cobol e com a introdução do Meta-Cobol

Figura 5.15 - Projeção do número de programadores necessário para o período de 1985 a 1989 a se continuar sem o Meta-Cobol
- e.p.a. -

só seria efetivado no segundo semestre desse ano e as primeiras versões não contariam com toda a potencialidade da versão final. Mas já em 1986, com o lançamento da versão final do Meta-Cobol, conseguir-se-ia passar o ano mantendo o mesmo efetivo do ano 1985 para atender uma quantidade 10% maior de programas novos e 29% maior de programas para manutenção.

Em 1989, de acordo com os dados projetados no subitem 5.6.3. a Programação teria de atender em média a cada mês 131,3 programas novos e 979,7 programas para manutenção num total de 1111,0 programas. Como atualmente essa média tem sido de 404,4 programas, a capacidade de atendimento teria de aumentar a uma taxa média de 22,4%, acumulativamente, ao ano. Segundo os resultados das simulações, a se continuar desenvolvendo convencionalmente os programas "batch", para suprir a Programação dessa capacidade de atendimento, o seu efetivo teria de crescer a uma taxa média de 19,4% ao ano. Com o Meta Cobol, essa taxa média de crescimento cairia a 14,0% ao ano.

Tudo isso, evidentemente, seria conseqüência de um real ganho de produtividade dos programadores. Atualmente, os tempos necessários para o desenvolvimento de programas "batch" são muito longos. Reduzindo-os drasticamente através do uso do Meta-Cobol, os programadores passariam a contar com maior disponibilidade de tempo para desenvolver mais programas ou executar mais manutenções (essa disponibilidade não seria aproveitada em outras atividades ou em ociosidade, frente aos níveis de aproveitamento do tempo estabelecidos). Assim na prática, passariam a atender e liberar maior número de programas na unidade de tempo. Isso é o que veremos a seguir.

B. Aumento Real de Produtividade

Atualmente, os 42 programadores têm liberado uma média de 404,4 programas por mês, o que dá uma produção média individual de 9,6 programas por mês. Com base nos relatórios da média de programas novos e para manutenção liberados por mês, obtivemos os dados médios de produção individual em cada ano de cada situação (média dos resultados obtidos para os dois tamanhos escolhidos), os quais são apresentados na tabela 5.26.

SITUAÇÃO / ANO	SEM O META-COBOL	COM O META-COBOL	DIFERENÇA PERCENTUAL
1985	9,9	10,9	10,1
1986	10,5	12,7	21,0
1987	10,6	13,1	23,6
1988	11,0	12,8	16,4
1989	11,3	14,0	23,9

Tabela 5.26 - Produção média individual (programas liberados/mês)
- e.p.a. -

Numa primeira impressão, o leitor poderá estranhar o fato de que, independentemente do uso do Meta-Cobol, o desempenho individual dos programadores deverá "melhorar". Isso como vimos no subitem 4.2.1. decorre do fato da medida

$$\frac{\text{programas liberados}}{\text{mês . programador}}$$

não ser adequada para se medir a produtividade dos programadores por não considerar o tempo efetivo que se leva para atender cada programa. Como o leitor já deve ter notado, segundo as projeções feitas (vide tabelas 5.8 e 5.13), apesar de ter sido fixado o objetivo de se aumentar a capacidade de produção de programas novos a uma taxa real de 10% ao ano, a se persistir as atuais taxas de manutenção, com o crescimento do número de programas em produção, a incidência de manutenções deverá aumentar a taxas ainda maiores. Hoje, os programas destinados à manutenção correspondem a cerca de 80% do total de programas manipulados pela Programação. Segundo as estimativas, esse percentual chegaria a cerca de 88% em 1989.

Assim concluímos que, mesmo sem o Meta-Cobol, os programadores passariam realmente a liberar maior quantidade de programas na unidade de tempo, já que o percentual de manutenções aumentaria ainda mais, e os tempos que normalmente se gastam em tais atividades são significativamente menores que os dispendidos em desenvolvimento. Entretanto, observa-se que com o Meta-Cobol, os programadores aumentariam ainda mais esta média de produção individual. Em cada ano do período simulado, a diferença entre esses dois valores, se deverá na realidade, a um real aumento de produtividade. Segundo os dados da tabela 5.26 ainda em 1985, se registraria um aumento real de produtividade em torno de 10%. Tirando-se a média para os 4 anos seguintes, temos que a versão final do Meta-Cobol, proporcionaria um aumento real da ordem de 21,2%.

Esses resultados, assim como os apresentados em A, chegam a ser surpreendentes levando-se em conta que do total de programas manipulados, apenas uma pequena parcela (cada vez menor com o passar dos anos) será constituído de programas novos e destes cerca de 56% é que caberá efetivamente aos programas do tipo "batch".

C. Redução dos Prazos de Desenvolvimento de Programas "batch"

Adotando procedimento análogo ao utilizado na determinação do tamanho adequado, calculamos para cada ano de cada situação, a partir dos valores obtidos do prazo médio para o desenvolvimento de programas "batch" (média aritmética dos prazos obtidos para os programas "batch" de atualização e de relatório, já que as suas quantidades deverão ser bem equilibradas) para cada tamanho experimentado, o prazo que corresponderia ao número adequado de programadores de cada caso (atravessando uma reta pelos pontos obtidos). Os valores utilizados na determinação dos prazos foram retirados dos relatórios das simulações (vide Anexos VI e VII e as listagens no volume em anexo). As tabelas 5.27 e 5.28 apresentam, respectivamente, os resultados obtidos nas simulações do futuro com e sem a introdução do Meta-Cobol.

ANO	MENOR DOS TAMANHOS TESTADOS E ESCOLHIDOS	PRAZO MÉDIO PARA A ENTREGA DE PROG. "BATCH" P/ O MENOR TAMANHO	MAIOR DOS TAMANHOS TESTADOS E ESCOLHIDOS	PRAZO MÉDIO PARA A ENTREGA DE PROG. "BATCH" P/ O MAIOR TAMANHO	TAMANHO ADEQUADO DETERMINADO	PRAZO MÉDIO PARA A ENTREGA DE PROG. "BATCH" P/ O TAMANHO ADEQUADO
1985	8	33,2	9	13,1	8,38	25,6
1986	8	30,8	9	8,7	8,35	23,1
1987	9	41,6	10	21,9	9,86	24,7
1988	12	33,3	13	17,4	12,58	24,1
1989	13	36,4	14	13,9	13,54	24,3

Tabela 5.27 - Prazos médios em dias úteis com a introdução do Meta-Cobol - e.p.a. -

ANO	MENOR DOS TAMANHOS TESTADOS E ESCOLHIDOS	PRAZO MÉDIO PARA A ENTREGA DE PROG. "BATCH" P/ O MENOR TAMANHO	MAIOR DOS TAMANHOS TESTADOS E ESCOLHIDOS	PRAZO MÉDIO PARA A ENTREGA DE PROG. "BATCH" P/ O MAIOR TAMANHO	TAMANHO ADEQUADO DETERMINADO	PRAZO MÉDIO PARA A ENTREGA DE PROG. "BATCH" P/ O TAMANHO ADEQUADO
1985	8	40,8	9	13,1	8,43	28,9
1986	9	68,3	10	16,5	9,76	28,9
1987	12	33,9	13	14,1	12,28	28,4
1988	14	29,1	15	23,6	13,68	30,9
1989	16	33,2	17	30,7	17,06	30,6

Tabela 5.28 - Prazos médios em dias úteis sem a introdução do Meta-Cobol.

- e.p.a. -

SEM O META-COBOL

TABLE TBINT
ENTRIES IN TABLE 120 — MEAN ARGUMENT 4.266 — STANDARD DEVIATION 2.148 — SUM OF ARGUMENTS 512.000 — NON-WEIGHTED

UPPER LIMIT	OBSERVED FREQUENCY	PER CENT OF TOTAL	CUMULATIVE PERCENTAGE	CUMULATIVE REMAINDER	MULTIPLE OF MEAN	DEVIATION FROM MEAN
1	0	.00	.0	100.0	.234	-1.528
2	24	12.99	19.9	80.0	.468	-1.055
3	25	20.83	40.8	59.1	.703	-.589
4	30	25.00	65.8	34.1	.937	-.124
5	17	14.16	79.9	20.0	1.171	.341
6	10	8.33	88.3	11.6	1.406	.806
7	6	4.99	93.3	6.6	1.640	1.272
8	1	.83	94.1	5.8	1.875	1.737
9	1	.83	94.9	5.0	2.109	2.203
10	3	2.49	97.4	2.5	2.343	2.668
11	1	.83	98.3	1.6	2.578	3.134
12	2	1.66	100.0	.0	2.812	3.599

REMAINING FREQUENCIES ARE ALL ZERO

COM O META-COBOL

TABLE TBINT
ENTRIES IN TABLE 141 — MEAN ARGUMENT 3.262 — STANDARD DEVIATION 1.347 — SUM OF ARGUMENTS 460.000 — NON-WEIGHTED

UPPER LIMIT	OBSERVED FREQUENCY	PER CENT OF TOTAL	CUMULATIVE PERCENTAGE	CUMULATIVE REMAINDER	MULTIPLE OF MEAN	DEVIATION FROM MEAN
1	0	.00	.0	100.0	.306	-1.673
2	45	31.91	31.9	68.0	.613	-.936
3	53	37.58	69.5	30.4	.919	-.194
4	24	17.02	86.5	13.4	1.226	.547
5	7	4.96	91.5	8.5	1.532	1.289
6	6	4.25	95.7	4.2	1.839	2.031
7	4	2.83	98.5	1.4	2.145	2.773
8	2	1.41	100.0	.0	2.452	3.515

REMAINING FREQUENCIES ARE ALL ZERO

Figura 5.16 - Distribuição do número de etapas de depurações por programa com e sem o uso do Meta-Cobol

O prazo de entrega de um programa novo é a soma do tempo em que ele fica parado na fila da Programação com o tempo corrido de atendimento, sendo que este é que seria mais diretamente afetado pela introdução do Meta-Cobol. Com a introdução dessa ferramenta, os programas "batch" sendo desenvolvidos em menor tempo efetivo, o seu tempo corrido de atendimento também diminuiria e, consequentemente, ocorreria uma redução dos prazos de liberação.

Observe-se pelas tabelas 5.27 e 5.28 que em cada ano os prazos médios seriam sistematicamente menores com a introdução do Meta-Cobol. Em média, haveria uma redução da ordem de 17% nos prazos medidos em dias úteis, o que, evidentemente, constituiria mais um benefício a ser proporcionado pelo Meta-Cobol.

D. Diminuição das Etapas de Depurações

Na figura 5.16 apresentamos comparativamente dois relatórios especiais (um para cada situação) que foram obtidos apenas para se ter uma idéia do número de depurações (compilações e processamentos para testes) que normalmente é necessário para se desenvolver um programa "batch", utilizando as técnicas atuais — convencionais — (pois o Setor de Informática e Serviços de Teleprocessamento não possui nenhum dado formal a respeito) e o que passaria a ser necessário com a introdução do Meta-Cobol, com base nas estimativas de tempos feitas pelos seus idealizadores (vide subitem 5.6.3.).

Segundo os resultados obtidos, para o desenvolvimento com técnicas convencionais seriam necessários cerca de 4,3 etapas de depuração por programa e cerca de 3,3 etapas com o uso do Meta-Cobol. O primeiro resultado, segundo os programadores e analistas procurados para analisá-lo, é bem representativo do que ocorre realmente. O segundo resultado é 23% menor, e espera-se realmente que esse valor esteja próximo do que se verificaria na prática. As razões para isso são muitas; dentre as quais, as principais seriam, a drástica redução (completa eliminação em alguns casos) das perdas de compi

lação por erros de sintaxe, a minimização da possibilidade de se cometer erros de lógica e a eliminação das perdas de processamentos de testes, por não inicialização dos campos com zeros ou espaços ("brancos"). O único fator que poderia prejudicar esse número de depurações necessárias, seria a não familiaridade dos programadores com o uso do Meta-Cobol como ferramenta de desenvolvimento. Mas isso pode ser resolvido com treinamento e, tem-se verificado que a assimilação do uso de linguagens de altíssimo nível pode ser feita em questões de dias, pois o seu aprendizado é muito rápido (vide ref. 3). Assim sendo, acreditamos que tal benefício deverá realmente ocorrer caso o Meta-Cobol seja introduzido.

E. Aumento da Parcela do Tempo Útil a ser Dispendido em Serviços de Manutenção

Vimos que o crescimento da incidência de manutenções se daria, a taxas anuais maiores que as almejadas para se aumentar a capacidade de produção de programas novos. Assim, independentemente da introdução do Meta-Cobol, os programadores passariam a dispender mais tempo ainda absorvidos em serviços de manutenção. Atualmente, se considerarmos apenas o tempo útil corrido dispendido em atividades produtivas e com as chamadas outras atividades (excluindo portanto, tempo útil gasto em férias e perdido em ociosidade), a Programação tem passado 38% desse total de tempo em desenvolvimento de programas novos e 43,4% em manutenções. Se considerarmos o total de tempo útil corrido, esses percentuais seriam respectivamente de 34,9% e 39,9% (vide resultados da calibração no Anexo IV).

Evidentemente, com o avanço das quantidades de programas para manutenção a cada ano, a tendência é de se alterar esse perfil. No caso da introdução do Meta-Cobol, as alterações que se verificariam, naturalmente, seriam muito mais acentuadas. A tabela 5.29 fornece os percentuais médios obtidos (média dos resultados dos dois tamanhos experimentados e escolhidos em cada ano de cada situação) para ambas as situa

SITUAÇÃO	SEM A INTRODUÇÃO DO META-COBOL		COM A INTRODUÇÃO DO META-COBOL	
Descrição da Parcela de Tempo / Ano	Parcela do Tempo Útil Dispendido em Desenvolvimento (%)	Parcela do Tempo Útil Dispendido em Manutenções (%)	Parcela do Tempo Útil Dispendido em Desenvolvimento (%)	Parcela do Tempo Útil Dispendido em Manutenções (%)
1985	31,5	49,6	28,8	51,6
1986	29,3	51,9	20,4	60,9
1987	30,0	51,0	16,9	64,1
1988	28,0	52,8	15,7	64,9
1989	24,5	56,5	15,9	65,0

Tabela 5.29 - Distribuição do tempo útil corrido, excluindo férias e ociosidade, entre desenvolvimento e manutenção de programas

- e.p.a. -

ções nos próximos anos, tomando-se o total de tempo útil corrido (excluindo-se férias e ociosidade) como base.

Observe-se que no primeiro ano de uso do Meta-Cobol, a distribuição do tempo ainda seria semelhante caso continuasse a se desenvolver convencionalmente os programas "batch". Entretanto, a partir da entrada em operação da versão final do Meta-Cobol, essa distribuição seria alterada significativamente. A parcela do tempo dispendido em manutenções se distanciaria cada vez mais da parcela dispendida em desenvolvimento, alcançando em 1989, 65% desse total de tempo útil, num crescimento da ordem de 50% em relação aos atuais 43,4%. Caso não contássemos com o Meta-Cobol, esse crescimento limitar-se-ia a 30%.

Independentemente da introdução do Meta-Cobol, esses dados servem de alerta para a direção do Setor de Informática e Serviços de Teleprocessamento, pois mantendo-se esta tendência a Programação passaria a dedicar cada vez maior parcela de tempo com serviços de manutenção. Talvez seja o caso de se tomar medidas eficazes para se minimizar as taxas de incidência de manutenção e/ou determinar uma taxa de crescimento da capacidade de produção de programas novos ainda maior, de modo a se evitar esta tendência.

Este problema, torna-se ainda mais delicado se considerarmos que, psicologicamente para os programadores, a execução de serviços de manutenção é muito mais desgastante e menos recompensadora (em termos de enxergar os frutos do trabalho) do que o de desenvolvimento de programas novos.

É necessário procurar minimizar a diferença entre essas parcelas de modo a se evitar que os programadores passem a se sentir desmotivados ao trabalho. O tratamento dessa questão, é particularmente mais importante ainda no caso de se usar o Meta-Cobol, pois como vimos, o desequilíbrio entre essas parcelas seria ainda maior neste caso.

5.7.3. PROCESSAMENTO DE SIMULAÇÕES PARA ANÁLISE DE SENSIBILIDADE

O objetivo da realização dessas simulações, é realizar a avaliação de sensibilidade do desempenho do sistema simulado a variações do ganho de produtividade proporcionável (redução do tempo efetivo para o desenvolvimento de programas "batch") pelo Meta-Cobol. A partir desses resultados, pretendemos avaliar até que ponto os resultados proporcionados pelas simulações descritas no subitem anterior, manter-se-iam razoavelmente válidos já que existe um risco não desprezível, de o Meta-Cobol não vir a proporcionar os resultados esperados pelos seus idealizadores. Por outro lado, estamos também interessados no que ocorreria, se o Meta-Cobol chegasse a superar as atuais expectativas.

Segundo as estimativas feitas pelos idealizadores do Meta-Cobol para os ganhos de produtividade esperados, a média dos tempos efetivos de desenvolvimento de programas "batch" passariam a ser no mínimo, de 16% da média atual, e no máximo, de 25%. Adotamos então, os valores de 10% (otimista) e de 30% (pessimista) para testarmos em simulações a sensibilidade dos resultados alcançados em relação ao ganho estimado: a média de tempo de desenvolvimento baixaria para 20% da atual média.

Por problemas de limitação de tempo para o desenvolvimento do trabalho e limitação de consumo de tempo de computador, restringimos a análise de sensibilidade em um ano intermediário do período de vida útil previsto para o Meta-Cobol (1985 a 1989), acreditando que os resultados obtidos poderão ser generalizados para os demais casos. O particular caso adotado para esta análise de sensibilidade, é o de 13 programadores para 1988.

Esperamos assim levantar curvas de variáveis representativas do desempenho do sistema em função desse percentual de ganho de produtividade do Meta-Cobol e analisar a sensibilidade dos resultados.

5.7.4. RESULTADOS DA ANÁLISE DE SENSIBILIDADE

Como planejado, realizamos simulações para o ano de 1988 com outros índices de ganhos de produtividade a se esperar do Meta-Cobol, procurando verificar a sensibilidade a essas variações do desempenho do particular grupo médio constituído de 13 programadores (em 1988). Além disso, experimentamos mais dois tamanhos de grupo médio (15 e 16) para a simulação do ano de 1989 com a introdução do Meta-Cobol. Assim, juntando os resultados obtidos nesses processamentos com os que já haviam sido obtidos (13 e 14) para a estimação do tamanho adequado, pudemos realizar também, um estudo para se analisar a sensibilidade do grupo médio a variações do seu tamanho. Os resultados obtidos são apresentados a seguir.

A. Análise de Sensibilidade Quanto a Variações do Ganho de Produtividade Proporcionável pelo Meta-Cobol

Os valores experimentados nesta análise são aqueles que foram estimados como o pior (30%) e o melhor (10%) casos possíveis quanto aos ganhos que o Meta-Cobol poderia proporcionar. Assim, como no caso padrão de uma redução média para 20% dos atuais tempos efetivos de desenvolvimento, pelos resultados globais obtidos na simulação desses casos extremos (os principais relatórios referentes às mesmas constam no Anexo VIII, percebe-se que o grupo médio se comportou de maneira aproximada, conforme às condições impostas pelo cenário, pois a média de liberação de programas se equilibrou com a média de chegadas, a ociosidade foi muito pequena e o valor do quociente p (aproveitamento do tempo útil em outras atividades) beirou os 20% de referência. Apenas o número médio de programas em mãos, que pelas estimativas deveria ser de cerca de 107, não foi satisfatoriamente alcançado.

Como o valor dessa variável, variou de forma bem clara conforme a redução dos tempos efetivos experimentados, acreditamos que a mesma, além de ser uma função do número de programadores em um determinado ano de cada situação, é também uma função do ganho de produtividade a ser proporcionado pelo

Meta-Cobol, para um determinado ano. Isso pode ser verificado pelos resultados globais obtidos que constam na tabela 5.30.

VARIÁVEL \ PERCENTUAL DOS TEMPOS EFETIVOS ATUAIS DA REDUÇÃO TESTADA	10%	20%	30%
ociosidade (% do total de tempo útil corrido)	2,8	0,5	0,7
dispêndio de tempo em outras atividades (p%)	20,1	19,0	19,5
média de programas recebidos (programas/mês)	158,5	157,9	162,3
média de programas liberados (programas/mês)	166,8	165,4	164,3
$\left(\frac{\text{média de prog.recebidos}}{\text{média de prog.liberados}}\right)$	0,950	0,955	0,988
média de programas em mãos	46,4	71,4	85,2

Tabela 5.30 - Dados globais das simulações processadas para o ano de 1988 considerando o aproveitamento do Meta-Cobol por um grupo médio de 13 programadores - e.p.a. -

Observe-se que, evidentemente, com ganhos maiores os programadores passariam a contar com disponibilidade de tempo ainda maior para atender a outros programas, aliviando-se a fila de programas e, consequentemente, diminuindo a média de programas em mãos. A figura 5.17 mostra graficamente o comportamento dessa variável em função da redução proporcionada pelo Meta-Cobol.

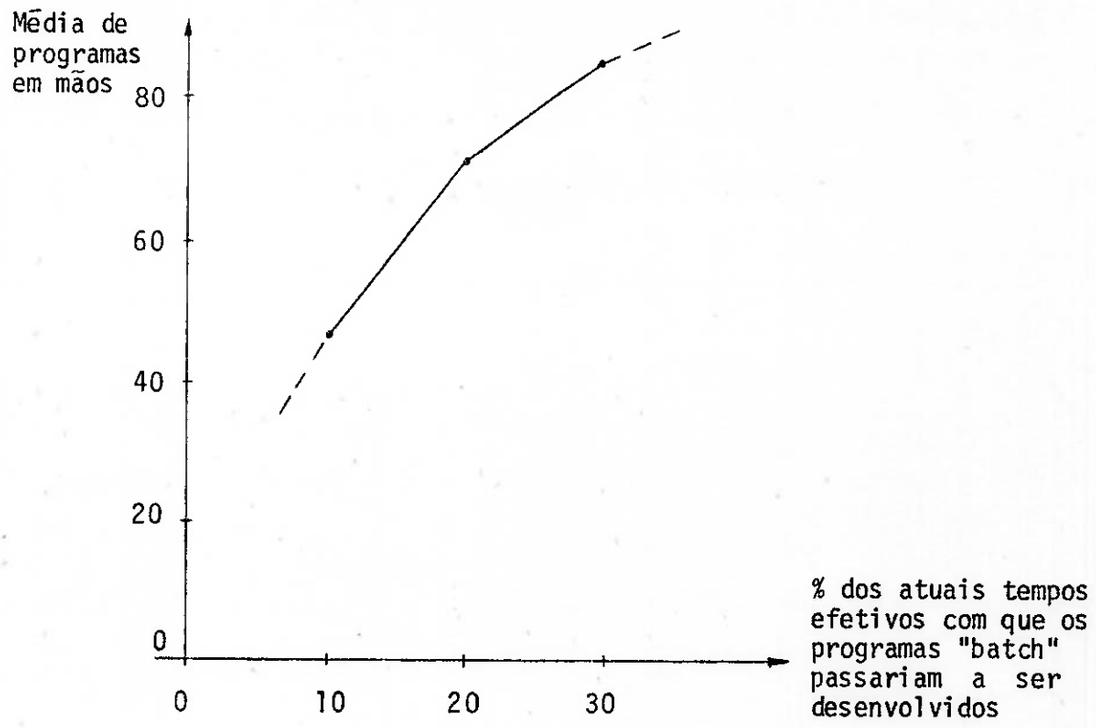


Figura 5.17 - Gráfico da média de programas em mãos em função do ganho de produtividade proporcionável pelo Meta-Cobol para um grupo médio de 13 programadores em 1988

- e.p.a. -

Conforme essa variação, os prazos para a entrega dos programas também sofreriam alterações já que, mantendo-se um número médio de programas em mãos maior o comprimento médio da fila também aumenta, e consequentemente, os prazos ficam mais dilatados. Caso a média de programas em mãos reduzisse, ocorreria justamente o inverso. Nas simulações da redução dos tempos efetivos a 10% e 30% dos tempos atuais, obtivemos respectivamente, o prazo médio de 9,48 dias úteis e 17,62 dias úteis. Com esses resultados mais o prazo que havíamos obtido para a redução a 20% (17,42 dias úteis) elaboramos o gráfico da figura 5.18.

Naturalmente, os benefícios ou os prejuízos advindos dessa redução ou dilatação dos prazos de entrega, como conse

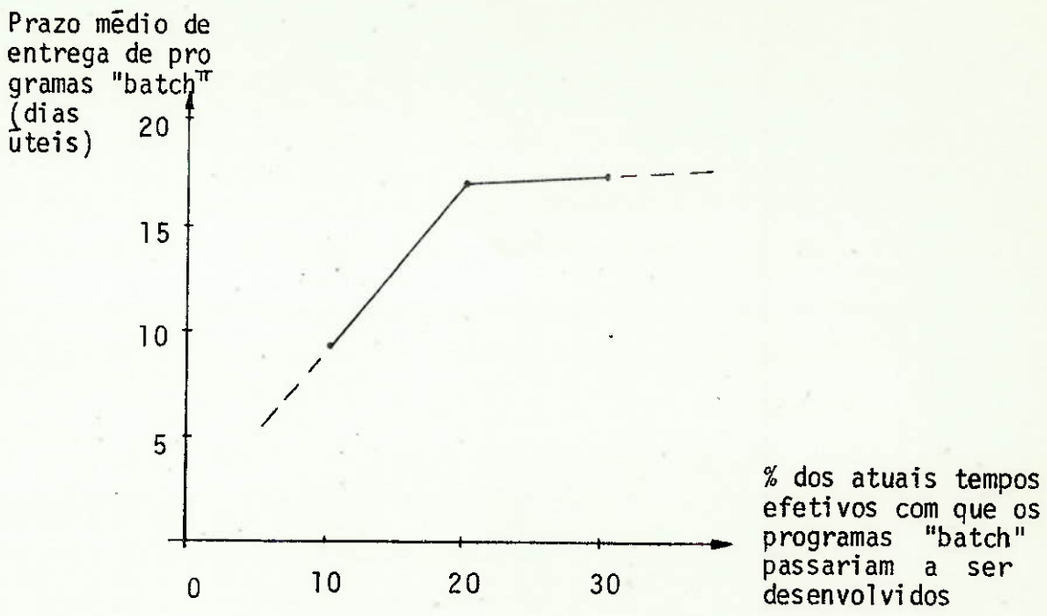


Figura 5.18 - Gráfico do prazo médio de entrega de programas "batch" em função do ganho de produtividade proporcionável pelo Meta-Cobol para um grupo médio de 13 programadores em 1988

- e.p.a. -

quência da variação do número médio de programas em mãos, não se limitariam ao âmbito do desenvolvimento de programas "batch", pelo contrário, podem ser generalizados ao atendimento de todos os tipos de serviços.

De qualquer forma, segundo os resultados obtidos e sumarizados na tabela 5.30 temos que, mesmo que ocorresse o pior caso em termos de ganho de produtividade (redução a 30% dos atuais tempos), o desempenho do grupo médio e, consequentemente, da Programação como um todo, não seria em muito prejudicado, levando em conta a suposição de que esses resultados (baseados no particular caso analisado da simulação de 1988 com 13 programadores) possam ser generalizados para os demais casos.

O único fator que mereceria maior atenção é a variação que se verificaria na média de programas em mãos. Entre

tanto, a sua consequência que mais nos interessa, no caso, a variação dos prazos médios de entrega, não seria em muito afetado, se os ganhos reais viéssem a ser menores que o esperado, e mesmo que atingisse o pior caso possível como podemos ver pela figura 5.18 para o caso de entregas de programas "batch". Acreditamos que tal tendência se verificaria de modo análogo também para a liberação de outros tipos de serviços.

Por outro lado, como podemos ver pelos gráficos apresentados (figuras 5.17 e 5.18), a inclinação da curva no trecho compreendido pelos valores menores do que 20% é mais acentuada. Desta forma, se os ganhos viéssem a ser superiores, provavelmente, os benefícios que adviriam disso seriam comparativamente maiores do que as perdas que se verificariam no caso oposto. Talvez, como conseqüência disso, tivemos na simulação com redução a 10% dos atuais tempos efetivos, uma ociosidade de 2,8% do tempo útil corrido que é significativamente maior que o obtido nas demais simulações (vide tabela 5.30).

Concluindo, acreditamos que contamos com evidências suficientes para supor que o desempenho global da Programação, deverá se manter pouco sensível à variações dos ganhos de produtividade dentro dos limites críticos, e que portanto, os resultados apresentados no subitem 5.8.1. deverão permanecer consistentes e válidos.

B. Análise de Sensibilidade Quanto a Variações do Tamanho do Grupo Médio

Determinamos qual seria o tamanho adequado para o grupo médio com base na média de programas em mãos estimada para o futuro segundo o cenário elaborado. Mas evidentemente, estes números poderão não ser seguidos à risca.

Resolvemos então, analisar a sensibilidade do sistema a variações do seu tamanho e adotamos a simulação do ano de 1989 contando com a introdução do Meta-Cobol, para servir de base ao estudo. Realizamos então, simulações adicionais com

os tamanhos de 15 e 16 programadores, procurando verificar apenas o que ocorreria em caso de superdimensionamento do efetivo (o tamanho adequado para o grupo para esta situação, nesse ano é de 13,54 programadores), pois descartamos a hipótese de subdimensionamento do efetivo, já que isto acarretaria uma degradação do desempenho dos programadores e incapacidade de atender plenamente à demanda de programas projetada. Os resultados globais de todas as simulações incluídas nesta análise constam na tabela 5.31 e foram retirados os relatórios apresentados no Anexo IX.

Observe-se que qualquer que seja o tamanho (dentre os simulados) do grupo médio, os resultados obtidos estão muito próximos das condições determinadas pelo cenário, com a exceção, novamente, da média de programas em mãos e, consequentemente do prazo médio para a entrega de programas "batch".

Temos que, à medida que se intensifica o superdimensionamento do efetivo, como é de se esperar, o índice de ociosidade aumenta (não significativamente, pois enquanto houver programas em fila, haverá serviços a fazer) e a média de liberação de programas suplanta a média de recepções, como sinal de que o número de programas em mãos estaria sendo desafogado. E com esse desafogo, o prazo médio para a entrega de programas "batch" também se reduz significativamente (este resultado pode ser generalizado à liberação de outros tipos de serviços) como mostra a figura 5.19, embora o tempo corrido para o atendimento permaneça estável em torno de 1,2 dias úteis, já que o mesmo, independe do tamanho do grupo pois, como vimos, uma vez iniciado o desenvolvimento de um programa, essa atividade ganha altíssima prioridade de modo que possa ser concluída o mais breve possível.

A curva da figura 5.19, teoricamente, é limitado inferiormente por uma assíntota horizontal na altura de 1,2 dias úteis, pois no caso ideal, com a redução a zero do tempo em fila, o prazo de entrega tenderia ao próprio tempo corrido médio para o atendimento do programa.

VARIÁVEL \ NÚMERO DE PROGRAMADORES TESTADO	13	14	15	16
ociosidade (% do total de tempo útil corrido)	0,6	0,7	1,0	3,0
dispêndio de tempo em outras atividades (p%)	18,1	18,6	19,3	20,0
média de programas recebidos (programas/mês)	187,2	183,0	185,3	190,5
média de programas liberados (programas/mês)	176,0	186,1	195,0	199,5
$\left(\frac{\text{média de programas recebidos}}{\text{média de programas liberados}}\right)$	1,064	0,983	0,950	0,955
média de programas em mãos	191,6	69,5	60,3	33,5
tempo médio corrido para atendimento de programas "batch" (dias úteis)	1,15	1,04	1,28	1,22
prazo médio para entrega de programas "batch" (dias úteis)	36,45	13,89	11,78	6,36

Tabela 5.31 - Dados globais obtidos das simulações para a análise de sensibilidade do desempenho do sistema em relação à variação do seu tamanho

- e.p.a. -

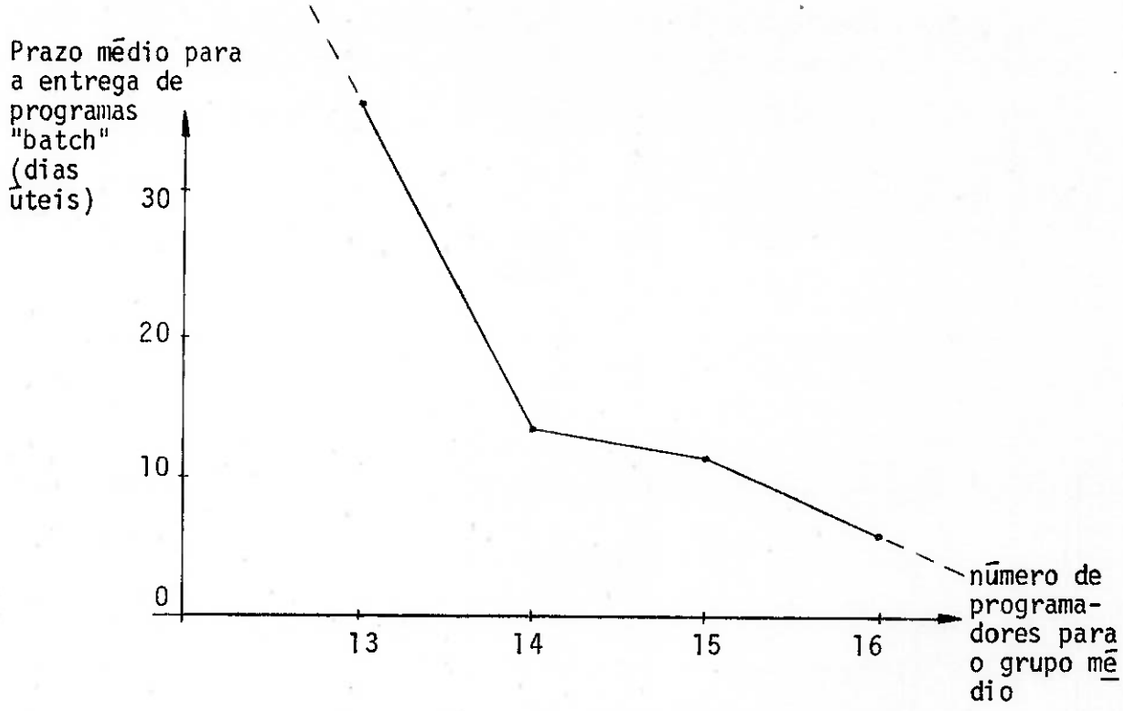


Figura 5.19 - Gráfico do prazo médio para a entrega de programas "batch" em função do tamanho do grupo
- e.p.a. -

Estimando o prazo médio que se verificaria para o tamanho adequado por interpolação linear (vide tabela 5.27), obtivemos o prazo médio de 24,3 dias úteis que é 21% menor que o prazo que se verificaria ao se continuar sem o Meta-Cobol.

Adotando-se um efetivo maior ainda, vemos que a direção poderia conquistar prazos mais reduzidos ainda. Trata-se sem dúvida de mais um problema de se determinar a situação de melhor relação benefício/custo. Os custos, no caso, seriam o adicional de salários e encargos para se manter um efetivo superdimensionado. Os benefícios, no caso, são tipicamente não mensuráveis e intangíveis, sendo constituídos basicamente dos efeitos da redução dos prazos médios de entrega (não somente para os programas novos do tipo "batch").

Para que o leitor tenha uma idéia da importância desses prazos apresentamos o gráfico da figura 5.20 cujos pontos foram determinados, obtendo-se das Folhas de Acompanhamento do Programador (vide exemplo na figura 4.10) o prazo que o ana

lista havia estipulado para o desenvolvimento de programas "batch" a partir do instante em que os mesmos saíram da fila (como já foi dito, o tempo de espera não é registrado, mas logo que o programa sai da fila, o programador comunica o início do seu atendimento ao analista e este combina o prazo para a liberação) e o prazo de que o programador efetivamente necessitou.

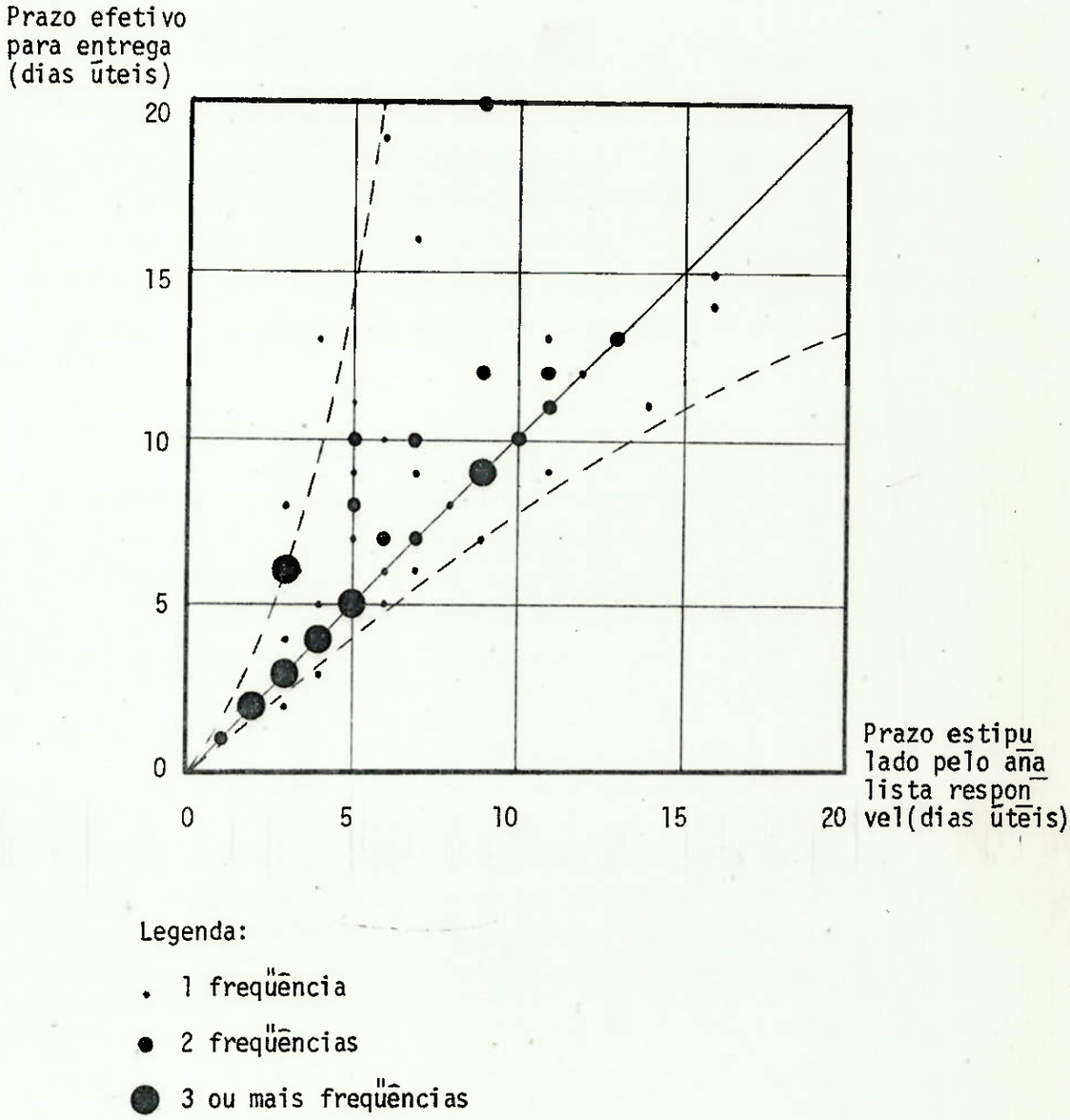


Figura 5.20 - Gráfico comparativo dos prazos realizados contra os prazos estipulados

- e.p.a. -

Observe-se que embora frequentemente os programadores consigam liberar o programa no dia combinado, a incidência de atrasos é muito maior que a de antecipações nos prazos estipulados.

Além disso, observe-se que os desvios do prazo realizado pelos programadores em relação ao prazo estipulado pelos analistas, tendem a ser muito maiores no caso de atrasos do que de antecipações. Outra importante constatação é a de que quanto menores os prazos estipulados, os analistas ficam sujeitos a menores desvios relativos. Tudo isso é típico do ser humano, porque a curto prazo, conseguimos fazer boas previsões e cronogramas, mas a médio e longo prazo, isso fica cada vez mais difícil.

Disso tudo, concluimos que a importância de se conseguir prazos menores de entrega, reside no fato de que com isso os analistas passariam a contar com estimativas de prazos mais confiáveis e, consequentemente, poderiam melhorar a qualidade do planejamento e cronograma para o desenvolvimento ou manutenções de seus Sistemas de Informação.

O Meta-Cobol já propiciaria uma pequena redução desses prazos e já elaboramos a projeção do efetivo adequado de programadores, mas fica aberta a opção de se adotar efetivos maiores com base no gráfico da figura 5.19, de modo a se alcançar prazos mais satisfatórios a um custo aceitável.

Da figura 5.19, também podemos concluir que realmente, não seria aconselhável ariscar um subdimensionamento do efetivo. Observe-se que para efetivos menores que o adequado (13,54) a inclinação da curva é bastante acentuada, sugerindo que em caso de subdimensionamento, os prazos de entregas degradariam drasticamente, piorando a qualidade do atendimento.

6. VIABILIDADE ECONÔMICA DO META-COBOL

Procuraremos agora, avaliar a viabilidade em termos econômicos do desenvolvimento do Meta-Cobol. Basearemos este estudo numa análise de custos e benefícios e adotaremos algumas simplificações já que há muitas variáveis envolvidas de difícil quantificação e o nosso intento não é o de se justificar minuciosamente a viabilidade econômica do desenvolvimento e operação do Meta-Cobol, mas sim, apenas dar uma idéia ao leitor dos valores monetários envolvidos.

No item 6.1. levantaremos os principais custos envolvidos e no item 6.2., os principais benefícios diretos esperados. Os valores encontrados nestes itens serão confrontados no item 6.3. de maneira a se avaliar o retorno dos investimentos que seriam necessários.

6.1. LEVANTAMENTO DE CUSTOS

Para se mensurar os principais custos envolvidos, tomaremos como base os valores unitários apresentados pela tabela 6.1 que foram fornecidos pela gerência do Setor de Informática e Serviços de Teleprocessamento, em julho de 1984. Os valores estão devidamente corrigidos de Cr$ da época para ORTN que é a moeda deflacionada com a qual desenvolveremos este estudo, de modo a contornar o problema da inflação do período.

DESCRIÇÃO DO CUSTO	Cr$/HORA EM JULHO DE 1984	ORTN/HORA
Remuneração e encargos de analistas	40.888,00	3,085
Remuneração e encargos de programadores	22.726,00	1,715
Remuneração e encargos de digitadores	16.156,00	1,219
Uso de computador	1.072.059,00	80,882

Tabela 6.1 - Valores dos custos unitários de recursos - e.p.a. -

Em julho de 1984, 1 ORTN era equivalente a Cr$ 13.254,67.

Os custos envolvidos são basicamente de dois tipos; custos de desenvolvimento e implantação, e custos de operação, os quais são apresentados a seguir.

A. Custos de Desenvolvimento e Implantação

Como vimos, caso se decida pela aprovação do Meta Cobol, o seu desenvolvimento já poderia ser iniciado a partir de janeiro de 1985 e duraria 1 ano, até a implantação da versão definitiva ao final do referido ano.

Espera-se cumprir tal cronograma com a alocação de um analista em regime de tempo integral (tomaremos como base um mês médio de 183 horas úteis), e mais dois programadores, mas estes no entanto, poderiam continuar a assumir serviços de manutenção de outros sistemas, trabalhando assim, em regime de tempo parcial para o Meta-Cobol. Estimaremos que os dois programadores trabalharão 70% do seu tempo produtivo no projeto do Meta-Cobol.

Além destes recursos, seria necessário ainda, o trabalho de digitação para se armazenar os programas a serem desenvolvidos em memória em discos. O pré-compilador que seria desenvolvido para gerar programas fonte em COBOL, a partir dos programas originais codificados na linguagem do Meta-Cobol, seria basicamente uma ferramenta de "software" constituída de um programa principal para gerenciar todo o pré-processamento apoiado em cerca de 11 outras rotinas "periféricas". Consideraremos então que, o trabalho de digitação que seria necessário seria, aproximadamente, equivalente ao de se digitar 12 programas de médio porte.

O pré-compilador do Meta-Cobol seria tipicamente uma aplicação do tipo "batch". Fizemos então, um levantamento do número médio de linhas para programas do tipo "batch" consultando várias listagens e encontramos a média de 1.535 linhas "fonte". Assim sendo, o Meta-Cobol requereria um total de

cerca de 18.500 linhas de programas fonte. As digitadoras do Setor de Informática e Serviços de Teleprocessamento apresentam uma velocidade média de digitação de 450 linhas/hora, donde chegamos ao total de cerca de 40 horas de digitação para o desenvolvimento do Meta-Cobol.

Finalizando, temos de estimar o custo do uso de computador que seria necessário para se realizar compilações e testes do programa gerenciador, e de todas as rotinas por ele acessadas. Com base nas mesmas listagens consultadas para se levantar o tamanho médio dos programas "batch", verificamos que a soma do seu tempo de compilação e "link-edição" é em média de 19,1 segundos. Normalmente em testes leva-se menos tempo do que isso, mas generalizaremos essa média para todos os processamentos que serão necessários (estamos assumindo um caso pior). Como se trata do desenvolvimento de "software" que foge um pouco dos padrões normais reservaremos 10 processamentos entre depurações e testes, o que é mais do que o dobro do que se têm verificado usualmente. Assim sendo, chegamos ao total de cerca de 2.300 segundos, ou cerca de 0,64 hora para a alocação de uso de computador.

Levantados todos estes recursos, elaboramos a tabela 6.2 com os custos de desenvolvimento envolvidos. Observe se que já calculamos o total de dispêndios para todo o ano de 1985, pois neste trabalho faremos uma avaliação simplificada, adotando períodos de um ano e fazendo com que os valores totais anuais incidam no meio dos respectivos anos (média do regime de competência) já que na realidade os custos incorrem mensalmente.

Observe-se que não incluímos os custos de utilização de pastas, formulários, listagens de computador e outros recursos necessários para os trabalhos de desenvolvimento, além dos custos de treinamento e preparação de manuais, pois os mesmos constituem uma parcela insignificante em relação ao total dos valores envolvidos. Os custos de mão-de-obra indireta, como salários e encargos dos supervisores e gerentes, já estão embutidos dentro dos custos unitários de mão-de-obra, e foram portanto, indiretamente incluídos no cálculo.

RECURSO	TOTAL (hora	CUSTO UNITÁRIO (ORTN/hora)	CUSTO TOTAL (ORTN)
Analista	2.196,0	3,085	6.774,7
Programadores	3.074,4	1,715	5.272,6
Digitadoras	40	1,219	48,8
Computador	0,64	80,882	51,8
	Total dos custos de desenvolvimento e implantação		12.147,9

Tabela 6.2 - Custos dos recursos necessários ao desenvolvimento e implantação do Meta-Cobol
- e.p.a. -

B. Custos de Operação

Com a colocação do Meta-Cobol em operação, o Setor de Informática e Serviços de Teleprocessamento passará a arcar basicamente com dois custos adicionais: custos de processamento do pré-compilador para a geração de programas fonte e custos para a manutenção do "software" a ser implantado.

Teoricamente, desde que o Meta-Cobol seja bem aproveitado, o programador necessitaria utilizar o pré-compilador apenas uma vez em cada desenvolvimento, gerando o programa fonte desejado em COBOL, para logo em seguida, submetê-lo a uma compilação e "link-edição". Caso se constate erros, as alterações necessárias poderiam ser efetuadas diretamente sobre o próprio programa fonte (sem recorrer novamente ao pré-compilador) e posteriormente realizar os testes do programa. Entretanto, nesse estudo, assumiremos o pior caso, ou seja, que o programador realizará as depurações não sobre o programa gerado, mas sim, sobre o programa original em Meta-Cobol, o que implica em assumir que em toda etapa de depuração do programa, o pré-compilador voltará a ser acionado.

Quanto ao número médio de compilações a ser realizado, embora tudo indique que com o Meta-Cobol ele possa diminuir, assumiremos novamente o pior caso, ou seja, que se mantenha na atual média que é de cerca de 3 por programa.

Quanto à etapa de testes do programa, embora espere-se que o Meta-Cobol venha a contribuir em evitar perdas de processamentos para testes por erros de lógica (vide Capítulo 3), admitiremos novamente o pior caso, ou seja, consideraremos que cada programa continuará a ser testado cerca de 1,3 vezes, e assim sendo, os custos referentes a esta etapa deverão permanecer estáveis[1]. Segundo os idealizadores do Meta-Cobol, em média os processamentos do pré-compilador (geração de programas fonte do tipo "batch") acarretaria o uso de um tempo de computador (CPU) um pouco maior que o tempo requerido pelo gerador de programas "on-line" SCOA, para se efetuar tarefa análoga com os programas para aplicações "on-line". O SCOA tem consumido em média 15,3 segundos de tempo de computador para se gerar programas "on-line" com uma média de 2.190 linhas (que é significativamente maior que o número médio de linhas verificado para os programas "batch". Assumiremos, então que (procurando novamente superestimar o valor) em média, o pré-compilador do Meta-Cobol leve 30 segundos de uso de computador (se fizéssemos uma analogia com o SCOA, por uma "regra de três" essa média seria de 10,7 segundos).

Com base então, em todos esses valores assumidos, e na projeção do número de programas "batch" a ser desenvolvido durante o período de operação do Meta-Cobol, elaboramos a tabela 6.3 com os custos de consumo de computador para o processamento do pré-compilador do Meta-Cobol em cada ano.

---

[1] Estamos também, a rigor, assumindo a hipótese de que os programas gerados a partir do pré-compilador terão desempenho muito próximo aos dos programas que seriam obtidos segundo as técnicas convencionais.

ANO	NÚMERO DE PROGRAMAS "BATCH" A SER DESENVOLVIDO COM O "META-COBOL"	NÚMERO DE VEZES QUE O "PRÉ-COMPILADOR" SERÁ UTILIZADO	TEMPO TOTAL DE CONSUMO DE COMPUTADOR (horas)	CUSTO DO USO DE COMPUTADOR PARA PRÉ-COMPILAÇÃO (ORTN)
1985	394	1.182	9,850	796,7
1986	886	2.658	22,150	1.791,5
1987	975	2.925	24,375	1.971,5
1988	1.073	3.219	26,825	2.169,7
1989	1.179	3.537	29,475	2.384,0

Tabela 6.3 - Custos de utilização do computador para geração de programas fonte com o uso do pré-compilador do Meta-Cobol
- e.p.a. -

No total de programas a ser desenvolvido, já está incluída a participação daqueles que deverão ser produzidos pelos analistas (vide tabela 5.11), e em 1985, já levamos em consideração que o Meta-Cobol só entrará em fase operacional (para uso irrestrito a todos) no início do segundo semestre.

No tocante aos recursos necessários para a manutenção, acredita-se que alocando um analista (de preferência o mesmo que coordenar o trabalho de desenvolvimento e implantação) em regime de tempo parcial, alocando cerca de 20% do seu tempo produtivo durante todo o período de vida do Meta-Cobol, já será suficiente. Esse analista, evidentemente, necessitará de recursos de utilização de computador para executar os serviços de manutenção. Reservaremos então, 5% do valor total de custo de uso de computador para desenvolvimento e im

plantação em cada ano entre 1986 e 1989, que é a média que se tem verificado para a manutenção das demais ferramentas próprias de "software" já instaladas no ambiente. As manutenções das primeiras versões a serem implantadas durante o segundo semestre, são perfeitamente exequíveis pelo próprio pessoal de desenvolvimento. Assim sendo, entre 1986 e 1987, incorreria anualmente os custos manutenção dados pela tabela 6.4.

RECURSO	TOTAL (hora)	CUSTO UNITÁRIO (ORTN/hora)	CUSTO TOTAL (ORTN)
Analista	439,2	3,085	1.354,9
Computador	0,032	80,882	2,6
	Total dos custos anuais de manutenção		1.357,5

Tabela 6.4 - Custos anuais de manutenção para o período entre 1986 e 1989

- e.p.a. -

6.2. LEVANTAMENTO DOS BENEFÍCIOS

Como vimos, o Meta-Cobol proporcionaria muitos benefícios, que poderiam ser resumidos conforme a relação que vem a seguir:

a) Redução do efetivo necessário à Programação (aumento de produtividade dos programadores).

b) Redução dos trabalhos de digitação.

c) Agilização do trabalho de depuração dos programas em desenvolvimento (diminuição das perdas de compilação ou processamento para testes por erros no programa fonte).

d) Redução dos prazos de desenvolvimento de programas "batch".

e) Facilidade de assimilação e uso.

f) Todos os fatores considerados na fase de avaliação de alternativas (vide tabela 2.4) que levaram o Meta-Cobol a ser considerado como a alternativa mais pertinente ao problema e ao ambiente (integração com as demais ferramentas próprias, acúmulo de "know-how" e prestígio, aproveitamento da linguagem COBOL e de seu compilador, etc.).

Destes, os benefícios dados em d, e e f são tipicamente intangíveis e não serão incluídos neste estudo. Servirão apenas como fatores de ponderação para a decisão final da direção do Setor de Informática e Serviços de Teleprocessamento. O benefício dado em c é quantificável, mas como optamos por considerar um caso crítico (nenhuma melhora em termos de depuração), não incluiremos aqui as economias que adviriam do mesmo.

Assim sendo, consideraremos para fins do presente estudo, apenas as economias diretas que se verificariam dos benefícios dados em a e b.

A tabela 6.5 apresenta as economias que se verificariam, já a partir de 1985 com a redução do efetivo necessário à Programação. Convém lembrar ao leitor, que tais economias não seriam geradas via demissão de parte do pessoal, mas sim, pela contratação de menos programadores do que seria necessário caso se continuasse a desenvolver convencionalmente os programas "batch". Aproveitamos as próprias projeções do efetivo adequado à Programação com base nos resultados da simulação (vide tabela 5.25) para se chegar ao valor das economias monetárias a serem geradas. Evidentemente, com a redução do efetivo haveria menos necessidade de ampliação de área e mobiliário, além de menores gastos com materiais de consumo. Entretanto, por simplificação, não consideraremos os respectivos valores, adotando novamente o pior caso.

ANO	DIMINUIÇÃO NO EFETIVO NECESSÁRIO À PROGRAMAÇÃO (nº de programadores)	TOTAL DE HORAS DE PROGRAMADOR ECONOMIZADO (horas)	CUSTO DO TOTAL DAS HORAS DE PROGRAMADOR ECONOMIZADO (ORTN)
1985	1	2.196	3.766,1
1986	9	19.764	33.895,3
1987	15	32.940	56.492,1
1988	7	15.372	26.363,0
1989	21	46.116	79.088,9

Tabela 6.5 - Economias anuais com a redução do efetivo de programadores
- e.p.a. -

Finalmente, teremos as economias a serem geradas pela redução do trabalho de digitação. Admitiremos que os programas de fácil e de média complexidade se tornarão tão curtos, que se tornará prática mais conveniente que os próprios programadores se encarregarem de sua digitação. Consideraremos que apenas para os programas de alta complexidade, ainda valerá a pena solicitar os trabalhos de Digitação. Tais programas, segundo as estimativas feitas pelos idealizadores do Meta-Cobol com base nos histogramas de tempo efetivo de desenvolvimento de programas "batch" (vide subitem 5.6.3.), correspondem apenas a 5% dos programas desenvolvidos. Então, supondo que o perfil dos programas "batch" a serem desenvolvidos em relação ao seu grau de complexidade não se altere nos próximos anos, calculamos os valores das economias diretas a serem proporcionadas com a eliminação da requisição dos serviços

de digitação, para 95% dos programas "batch" a serem desenvolvidos entre 1986 e 1989. Os valores obtidos constam na tabela 6.6.

ANO	PROGRAMAS "BATCH" DE BAIXA E MÉDIA COMPLEXIDADE	TAMANHO TOTAL EM LINHAS DOS PROGRAMAS "BATCH" DE BAIXA E MÉDIA COMPLEXIDADE	TOTAL DE HORAS DE DIGITADORAS QUE NÃO SERÁ MAIS REQUISITADO	CUSTO DO TOTAL DE HORAS DE DIGITADORAS ECONOMIZADO (ORTN)
1986	842	1.292.470	2.872,2	3.501,2
1987	926	1.421.410	3.158,7	3.850,5
1988	1.019	1.564.165	3.476,0	4.237,2
1989	1.120	1.719.200	3.820,4	4.657,1

Tabela 6.6 - Economias anuais com a redução das requisições de serviços de digitação

- e.p.a. -

Observe-se que não incluímos o ano de 1985, pois assumimos que esse benefício só passaria a se verificar a partir da implantação da versão final do Meta-Cobol. Para se estimar o total de linhas dos programas que não seriam mais enviados à digitação, adotamos novamente a média de 1.535 linhas por programa (estamos superestimando, já que esses programas não incluem os de alta complexidade).

6.3. CONFRONTO DOS BENEFÍCIOS COM OS CUSTOS

A figura 6.1 resume todos custos que deverão incorrer com o desenvolvimento, implantação e operação do Meta-Cobol, já considerando o total dos custos anuais para cada período.

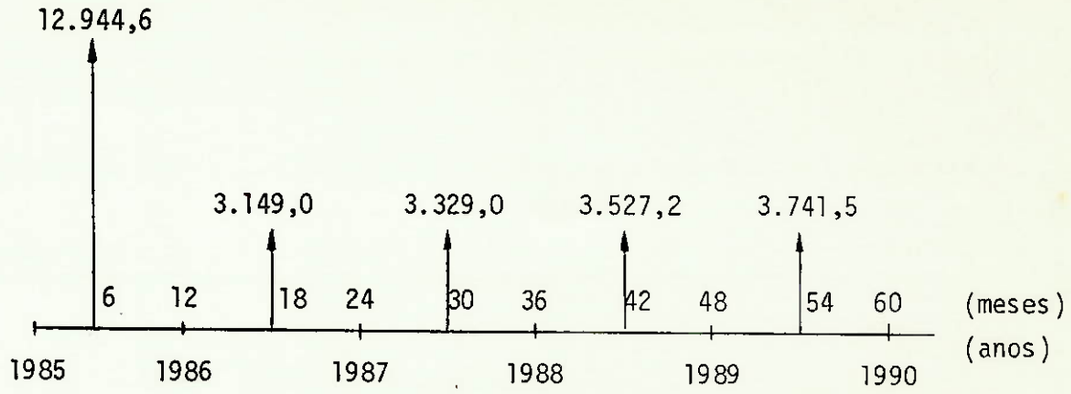


Figura 6.1 - Fluxo dos dispêndios anuais em ORTN
- e.p.a. -

A figura 6.2 resume todas as economias diretas que deverão se verificar com a introdução do Meta-Cobol.

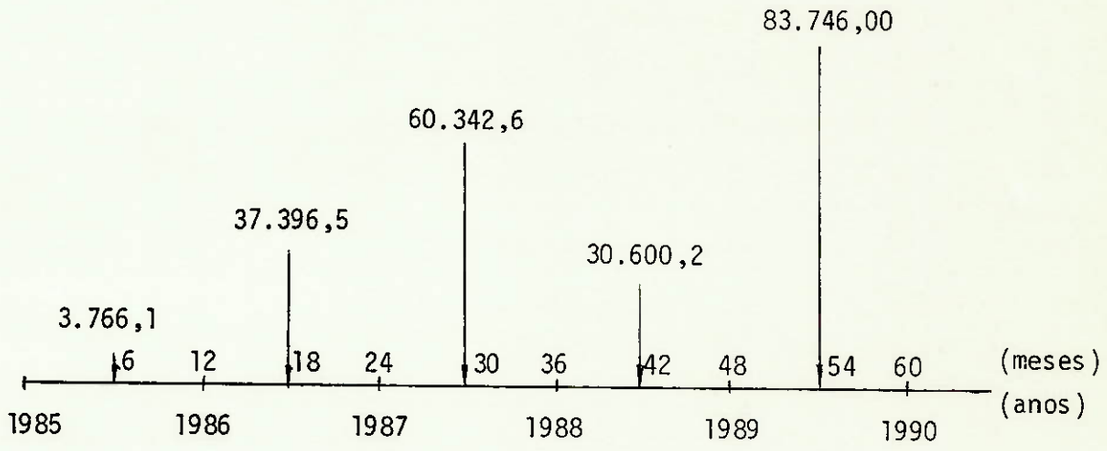


Figura 6.2 - Fluxo das economias anuais em ORTN
- e.p.a. -

Adotando-se então a taxa de atratividade mínima da empresa de 8,0% ao ano (0,643% ao mês) transportamos todos os valores para o início de 1985 ("data zero") e obtivemos os seguintes resultados:

Valor total dos dispêndios na data 1/1/85 = (23.348,6) ORTN

Valor total das economias na data 1/1/85 = 169.330,8 ORTN

Valor líquido dos fluxos na data 1/1/85 = 145.982,2 ORTN

Donde concluímos que o Meta-Cobol, do ponto de vista econômico, é uma ferramenta extremamente interessante, pois as economias que adviriam de sua introdução suplantam com grande margem de diferença os dispêndios que incorreriam. O resultado obtido, transportado para novembro de 1984 e convertido em cruzeiros da época (1 ORTN = Cr$ 20.118,71) equivale à quantia de Cr$ 2.899.542.035,00 o que justificaria plenamente o desenvolvimento do Meta-Cobol.

Para que o leitor tenha uma melhor idéia do que isso representa, diríamos que essa quantia seria equivalente ao custo de um mês de um efetivo de 84 programadores. Trata-se realmente, de uma considerável economia.

Em particular existem ainda, mais dois fatores que poderão fazer com que o valor do resultado real desse empreendimento venha a ser maior que o calculado, quais sejam:

a) Sempre que tivemos de assumir simplificações adotamos os piores vasos.

b) O período de vida que adotamos para o Meta-Cobol no presente trabalho, é inferior ao que os seus idealizadores vislumbram e que o Setor de Informática e Serviços de Teleprocessamento pretende conseguir.

Há ainda um outro aspecto a considerar. Observe-se que pela sua natureza, os custos de desenvolvimento, implantação e manutenção do Meta-Cobol independem do efetivo da Programação e da produção de programas "batch". O mesmo no entanto, não ocorre com os benefícios. As economias que seriam geradas são função da taxa de crescimento das atividades do

Setor de Informática e Serviços de Teleprocessamento para os próximos anos. Se a direção, na realidade, ambicionasse uma expansão ainda maior, a produção e os efetivos envolvidos seriam também maiores, e o Meta-Cobol, por sua vez, proporcionaria também economias maiores. Com isso queremos mostrar que existe a possibilidade do Meta-Cobol viabilizar taxas de expansão antes não cogitadas pela direção e que agora poderiam passar a ser consideradas.

Assim concluímos este Capítulo acrescentando mais um bom resultado para o Meta-Cobol e que, sem dúvida, será de grande valor para que a direção do Setor de Informática e Serviços de Teleprocessamento chegue à decisão final quanto ao aproveitamento ou não dessa ferramenta.

7. CONCLUSÕES

Pode-se dizer que os resultados concretos de todo o estudo desenvolvido e apresentado no presente trabalho, são basicamente de dois tipos: os resultados do estudo de avaliação do Meta-Cobol e o modelo de simulação desenvolvido.

As conclusões quanto a cada um desses dois tipos, são apresentadas nos itens seguintes.

7.1. QUANTO AO ESTUDO DE AVALIAÇÃO DO META-COBOL

Alcançamos todos os objetivos de avaliação do Meta Cobol (conforme Capítulos 5 e 6). Acreditamos serem os resultados obtidos consistentes, principalmente pelo ajustamento do modelo de simulação ao sistema real (vide resultados da calibração), e pelos resultados obtidos na Análise de Sensibilidade.

Sempre que deparamos com incertezas, assumimos as piores hipóteses, e mesmo assim, o Meta-Cobol mostrou ser uma poderosa ferramenta, a ponto de provocar consideráveis impactos ao ambiente (em caso de uma eventual implantação) intervindo diretamente sobre uma pequena parcela do montante de programas enviados à Programação. Observamos que os impactos que se verificariam são amplamente favoráveis (vide interpretação dos resultados no item 5.7.2.) e que economicamente, o projeto do Meta-Cobol, também é plenamente justificável (conforme Capítulo 6).

Concluímos então, que em se tratando de uma alternativa destinada à melhoria de eficiência da Programação, e em particular, da atividade de desenvolvimento de programas "batch", o Meta-Cobol apresenta enorme potencial em proporcionar resultados positivos. Se a decisão de implantação ou não, dependesse apenas desses ganhos e da sua viabilidade econômica, o Meta-Cobol apresenta amplas possibilidades de vir a ser aprovado. Mas na realidade, é necessário ainda examinar os resultados dos estudos desenvolvidos pela equipe encarregada de in

vestigar a sua viabilidade técnica, bem como analisar as outras implicações que estariam por trás da implantação dessa ferramenta, tais como o processo de implantação, sistema de treinamento, normas para o seu uso (disciplina) e sistema de manutenção, entre outras, que fogem do escopo do presente trabalho, mas que não podem, de forma alguma, serem ignoradas.

Vimos no Capítulo 1, que o objetivo de se melhorar o desempenho da Programação faz parte, na realidade, de um contexto muito mais amplo, que é o de se promover uma melhoria de qualidade das atividades de desenvolvimento de Sistemas de Informação como um todo. Isso só seria conquistado gradativamente com o tempo, mediante adoção de um conjunto de conceitos, modelos, métodos, ferramentas de trabalho, normas organizacionais, etc.. Como os estudos a seu respeito já estão em andamento, apresentaremos a seguir algumas constatações que poderão servir de subsídios aos mesmos.

a) "O Meta-Cobol proporcionaria uma redução dos prazos médios de entrega, mas isso seria desprezível em relação ao tempo de desenvolvimento total dos Sistemas de Informações".

A versão final do Meta-Cobol proporcionaria reduções dos prazos médios de entrega de programas novos do tipo "batch" à cerca de 24% dos prazos atualmente verificados (vide subitem 5.7.2. mas este benefício é desprezível em relação aos prazos totais necessários para o desenvolvimento de Sistemas de Informação. A tabela 7.1, transcrita da referência 14 (segundo alguns supervisores e analistas solicitados a avaliar os seus dados, pode-se considerar que os mesmos podem ser normalmente estendidos ao ambiente de desenvolvimento do Setor de Informática e Serviços de Teleprocessamento) ilustra que em média se leva apenas cerca de 20% do prazo de desenvolvimento e implantação total, na fase de codificação. Assim, conclui-se que o Meta-Cobol, não reduziria significativamente o tempo que normalmente se leva em desenvolvimento de sistemas.

Conclusão análoga, pode ser tirada com relação aos custos totais para o desenvolvimento de Sistemas de Informação

com base nos dados de distribuição de custos dados pela mesma tabela.

FASE	PARCELA DO PRAZO TOTAL (%)	PARCELA DO CUSTO TOTAL (%)
Levantamento de requisitos	10,0	8,0
Projeto preliminar	15,0	8,5
Projeto detalhado	15,0	13,5
Codificação	20,0	20,0
Testes	40,0	50,0

Tabela 7.1 - Distribuição do custo e do prazo de desenvolvimento de Sistemas de Informação
- transcrito da ref.14 -

b) "O Meta-Cobol é uma poderosa ferramenta, a ser destinada somente a programadores com razoável conhecimento de COBOL".

Vimos que o Meta-Cobol facilitaria em muito o desenvolvimento de programas "batch", mas que não eliminaria a necessidade de um conhecimento prévio da linguagem COBOL, pois as funções do programa não cobertas pelos meta-comandos, teriam de ser codificadas com comandos normais do COBOL, e porque o programa fonte gerado continuaria na linguagem COBOL. Mas por outro lado, aliviaria os programadores de terem conhecimentos mais específicos de outras ferramentas disponíveis no ambiente, e exigiria pouco treinamento para a assimilação do seu uso.

c) "Convém fazer um estudo quanto a possibilidade de aproveitar efetivamente o Meta-Cobol em trabalhos de manutenção".

Vimos que a taxa de aumento da capacidade de produção de programas novos, planejada para o próximo quinquênio (10% reais, ao ano), não será o suficiente para se manter, ao menos, a atual proporção de programas novos e programas para manutenção, pois a taxa de incidência destes deverá aumentar muito mais (em média 26% reais ao ano).

Hoje, as atividades de manutenção, já são aquelas que exigem maior parcela do total de tempo útil da Programação (40%) e com o ritmo de expansão esperado, elas tornar-se iam ainda mais predominantes o que, do ponto de vista psicológico, é muito prejudicial ao ambiente.

Convém, portanto, aos idealizadores do Meta-Cobol , pensar seriamente no aproveitamento desta ferramenta também para fins de manutenção. Provavelmente, seria inviável torná lo aproveitável para qualquer tipo de manutenção, mas parece nos razoável, pelo menos adaptar o seu projeto ou desenvolver recursos extras, de modo a se prever pelo menos a possibilidade de utilizá-lo na manutenção dos próprios programas que seriam gerados pelo mesmo.

Entretanto, ressaltamos que agilizar os trabalhos de manutenção com um eventual aproveitamento do Meta-Cobol, corresponderia a uma maneira de se minimizar os efeitos (consequências) das crescentes taxas de manutenções. Acreditamos que talvez seja mais sensato atacar o problema, intervindo diretamente sobre uma de suas principais causas (origens) que seriam as falhas e omissões cometidas durante a fase de análise e de projeto dos Sistemas de Informação (entre as outras causas, estão a alteração das condições ambientais ou conjunturais e o efeito da obsolescência, que praticamente não são controláveis).

7.2. QUANTO AO MODELO DESENVOLVIDO

Vimos que, grande parte do estudo desenvolvido, concentrou-se na análise e entendimento de toda a dinâmica da Programação, no levantamento de todas as suas funções (embora o Meta-Cobol afete apenas algumas particulares atividades) e na construção de um modelo relativamente complexo, detalhado, e representativo do sistema real. Acreditamos, que este modelo constitui na realidade, o resultado mais importante do presente trabalho (até mais do que os resultados de avaliação do Meta-Cobol obtidos, com base nos relatórios de simulação emitidos pelo respectivo programa).

Para chegar a essa conclusão nos baseamos nos seguintes aspectos.

a) "os resultados obtidos nas simulações processadas durante o desenvolvimento deste trabalho, são válidos para o particular cenário elaborado. Não é possível generalizá-los. Entretanto, isso não é problema, pois o modelo é flexível o suficiente, para possibilitar (sem nenhuma ou com apenas algumas poucas adaptações) a simulação de outros cenários futuros completamente distintos, ou a simulação de pequenas variantes de cada cenário".

Assim sendo, o modelo pode também ser aproveitado para se verificar o desempenho da Programação, sob as mais diversas condições como por exemplo: "o que aconteceria se introduzíssemos uma ferramenta para agilização dos trabalhos de manutenção?", "qual seria o efeito de um possível prolongamento da vida útil do Meta-Cobol?", "o que ocorreria se os trabalhos de desenvolvimento de programas novos também passassem a ser executados ininterruptamente?", "é razoável adotar um expediente de trabalho de 8 horas úteis mantendo-se o mesmo efetivo?", etc.

b) "pelas potencialidades do modelo apresentadas em a, concluímos que o mesmo pode ser considerado como um poderoso ins

trumento de investigações (realização de experimentos por meio de simulações, com possibilidade de obtenção de resultados à curto prazo e a um custo relativamente baixo, sem ter de se recorrer à experiências reais) e de auxílio ao planejamento (dimensionamento do efetivo, levantamento da necessidade de recursos) à disposição da direção e gerência do Setor de Informática e Serviços de Teleprocessamento. A sua grande vantagem reside no fato de poder substituir a necessidade de uma intervenção real ou de realizar experiências muito onerosas, ou até mesmo impossíveis na realidade."

c) "o modelo desenvolvido envolve, basicamente, o sistema da Programação, mas pode servir de base para o desenvolvimento de um modelo com uma amplitude maior (evolução gradual) que envolva também o pessoal da análise, os recursos de "hardware", e assim em diante[1], constituindo uma ferramenta de análise ainda mais poderosa".

d) "aproveitando os mesmos programas utilizados durante o presente trabalho é ainda possível obter uma série de outras informações quanto ao comportamento do sistema simulado, pois os mesmos emitem diversos outros relatórios que por serem pouco relevantes ao estudo, não foram aqui abordadas, mas que, durante a fase de desenvolvimento e de testes do modelo foram, absolutamente, indispensáveis (o modelo foi se sofisticando progressivamente, graças a um constante processo de "feed-back" proporcionado por essas informações)".

---

[1] Evidentemente, existe um nível de complexidade (abrangência) do modelo, acima do qual os custos envolvidos tornam-se inaceitáveis. O(s) modelador(es) desse novo modelo teriam de se definir pelo nível de complexidade a se alcançar ponderando, racionalmente, os custos e os benefícios envolvidos.

A simulação é uma técnica poderosa e flexível, mas muitas vezes não chega a ser adotada por exigir um trabalho longo e exaustivo, quando desenvolvida seriamente do começo ao fim. Segundo Gershefski (vide ref. 8), normalmente no desenvolvimento de um bom trabalho de simulação, se dispende 25% do tempo total para se formular o problema, 25% para se coletar e analisar os dados, 40% para se desenvolver o modelo e o respectivo programa, e 10% para se colocar o modelo à disposição de seus interessados. Observe-se então, que se de agora em diante, pessoas interessadas em se realizar estudos a respeito da Programação, perceberem, ainda na fase inicial de formulação do problema, que o modelo aqui desenvolvido pode ser total, ou em grande parte, aproveitado para o seu caso, estaremos-lhes proporcionando um grande benefício, mediante significativa redução do prazo que lhes seria necessário.

APÊNDICE A

PRINCIPAIS FERRAMENTAS DE "SOFTWARE" PRÓPRIAS DO AMBIENTE

A.1. Dicionário de Dados

Comumente corresponde apenas a um "depósito" de dados que descreve a natureza de cada dado usado nos Sistemas de Informação. Os dados do Dicionário de Dados, desenvolvido internamente, está instalado em unidades de memória secundária e pode ser acessado através de transações "on-line", além de incorporar uma série de recursos extras como o gerenciamento e controle das informações sobre os recursos e meios utilizados na administração do ambiente de processamento de dados, cadastramento de informações sobre os usuários, alocação de recursos, etc.

A.2. Gerador de Programas Fonte On-Line (SCOA)

É um sistema que recebe especificações do analista ou programador via transações "on-line" e que gera programas fonte em linguagem COBOL por meio de um programa chamado de SCOA. Dispensa os programadores de se preocuparem com a estrutura de dados já cadastrados no Dicionário de Dados.

Entre outros programas auxiliares, temos o STLA para a criação de tela (imagem do terminal de video) através da qual o programa se relacionará com o usuário e o SCOC para inserir lógicas complementares codificados em linguagem COBOL no programa fonte (o gerador não é genérico).

A.3. Sistema "Global"

Ferramenta que oferece uma série de facilidades(apoio) ao trabalho de desenvolvimento de aplicações, tais como o Di

cionário de Dados, documentação ativa tratável por computador, cadastramento de arquivos, estruturação de dados, registro dos relacionamentos entre arquivos, rotinas para abertura de sistemas, etc.. Conta com uma série de transações "on-line" e está integrada com outras ferramentas, tais como o gerador de programas fonte "on-line" e o DB VSAM (vem a seguir).

A.4. DB VSAM

É um sistema de armazenamento de dados inter-relacionados, segundo uma estrutura e organização (arquitetura) baseados no método de acesso a arquivos VSAM (da IBM). Está integrado com o Sistema "Global" e outras ferramentas de modo a facilitar o desenvolvimento de sistemas. Tem apresentado ótimos resultados por permitir acesso fácil e rápido aos dados em atualizações "on-line" (isso é fundamental para um CPD que atende a cerca de 3 milhões de transações "on-line" por mês), por permitir atualização de arquivos de forma sequencial (forma utilizada pela grande maioria dos programas "batch") e por ser de fácil manutenção e utilização (prático).

A.5. Gerador de Programas "batch" de Relatório

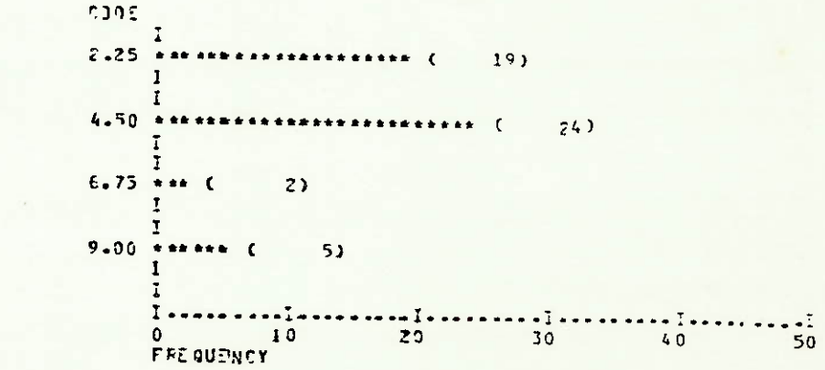
O REPORT WRITER é uma extensão da linguagem COBOL concebida para facilitar o desenvolvimento de programas "batch" destinadas à emissão de relatório. Esse sistema recebe por meio de interações "on-line" (terminais) as especificações fornecidas pelo usuário e com base nelas gera o programa fonte (às vezes sujeito a alguma complementação) em COBOL, inclusive com a codificação do REPORT SECTION, além de oferecer outras facilidades como a emulação do relatório especificado no terminal de video.

ANEXO I - DISTRIBUIÇÃO DE FREQÜÊNCIAS, HISTOGRAMA E ESTATÍSTICAS DAS AMOSTRAS DE TEMPO EFETIVO DE ATENDIMENTO DE ATIVIDADES

TEMPO DE DURACAO DAS OUTRAS ATIVIDADES

FILE NONAME (CREATION DATE = 11/08/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



MEAN	4.185	STD ERR	0.288	STD DEV	2.033
VARIANCE	4.135	KURTOSIS	0.910	SKEWNESS	1.150
VALID CASES	50	MISSING CASES	0		

TEMPO DE DURACAO DAS OUTRAS ATIVIDADES

FILE NONAME (CREATION DATE = 11/08/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

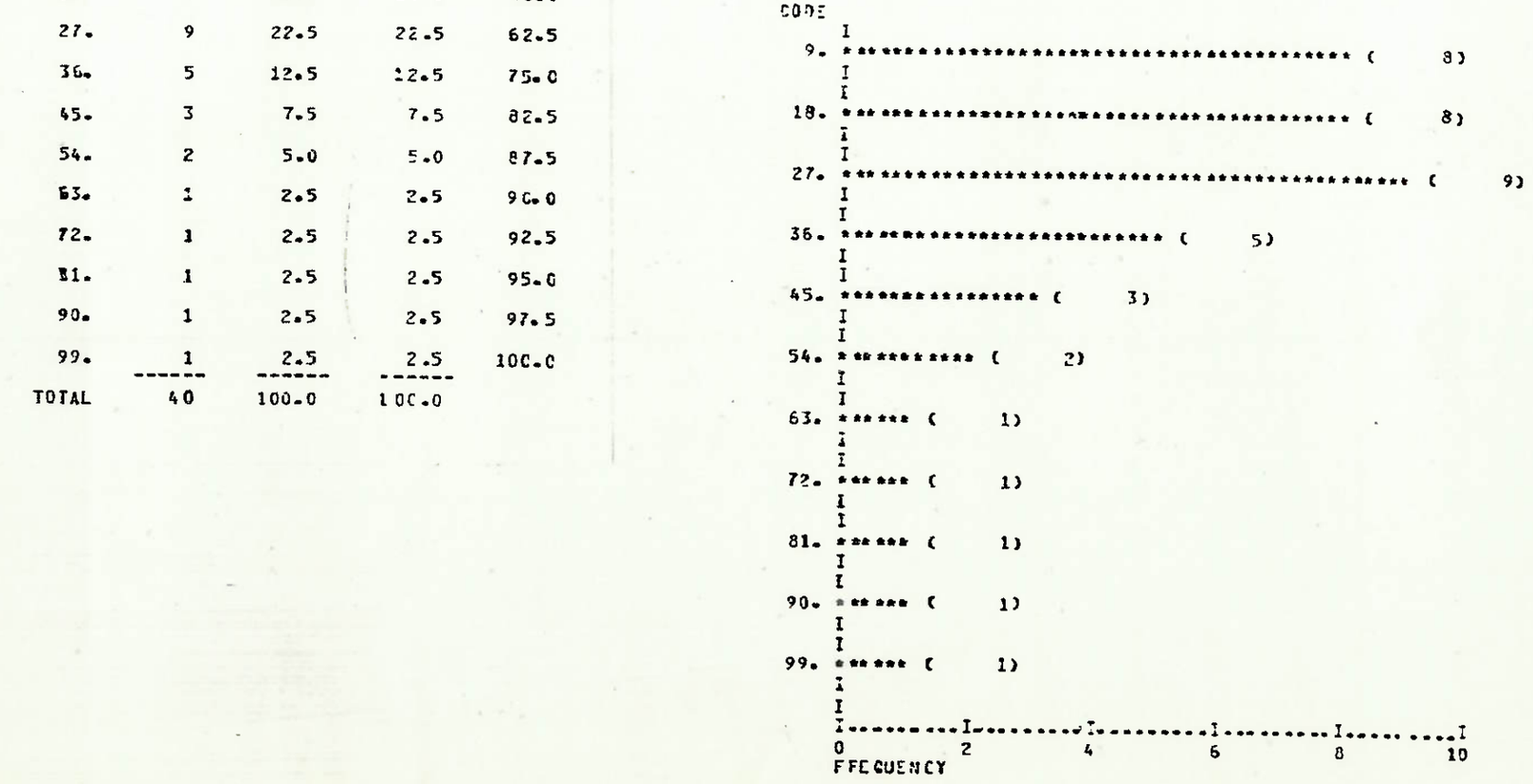
CATEGORY LABEL	CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
	2.25	19	38.0	38.0	38.0
	4.50	24	48.0	48.0	86.0
	6.75	2	4.0	4.0	90.0
	9.00	5	10.0	10.0	100.0
	TOTAL	50	100.0	100.0	

CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
9.	8	20.0	20.0	20.0
18.	8	20.0	20.0	40.0
27.	9	22.5	22.5	62.5
36.	5	12.5	12.5	75.0
45.	3	7.5	7.5	82.5
54.	2	5.0	5.0	87.5
63.	1	2.5	2.5	90.0
72.	1	2.5	2.5	92.5
81.	1	2.5	2.5	95.0
90.	1	2.5	2.5	97.5
99.	1	2.5	2.5	100.0
TOTAL	40	100.0	100.0	

TEMPO PARA DESENVOLVIMENTO DE PROGRAMAS BATCH/RELATORIO

FILE NOVNME (CREATION DATE = 11/06/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

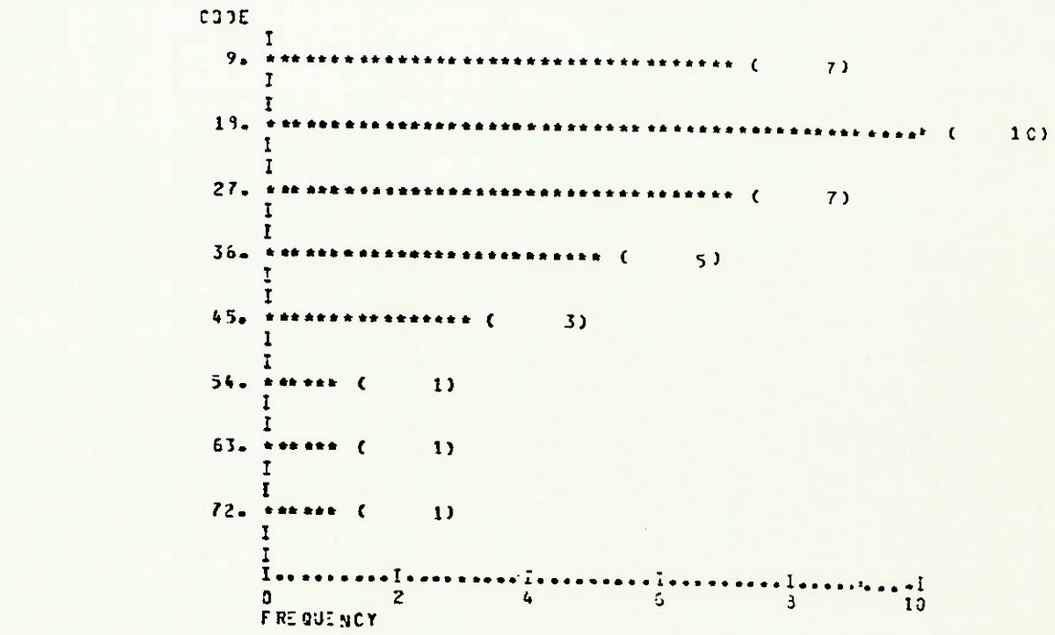


MEAN	32.175	STD ERR	3.616	STD DEV	22.867
VARIANCE	522.917	KURTOSIS	1.501	SKEWNESS	1.364
VALID CASES	40	MISSING CASES	0		

TEMPO PARA DESENVOLVIMENTO DE PROGRAMAS ON-LINE COM SCOA

FILE NONAME (CREATION DATE = 11/06/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



MEAN	26.743	STD ERR	2.673	STD DEV	15.814
VARIANCE	250.079	KURTOSIS	0.979	SKEWNESS	1.081
VALID CASES	35	MISSING CASES	0		

TEMPO PARA DESENVOLVIMENTO DE PROGRAMAS ON-LINE COM SCOA

FILE NONAME, (CREATION DATE = 11/06/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

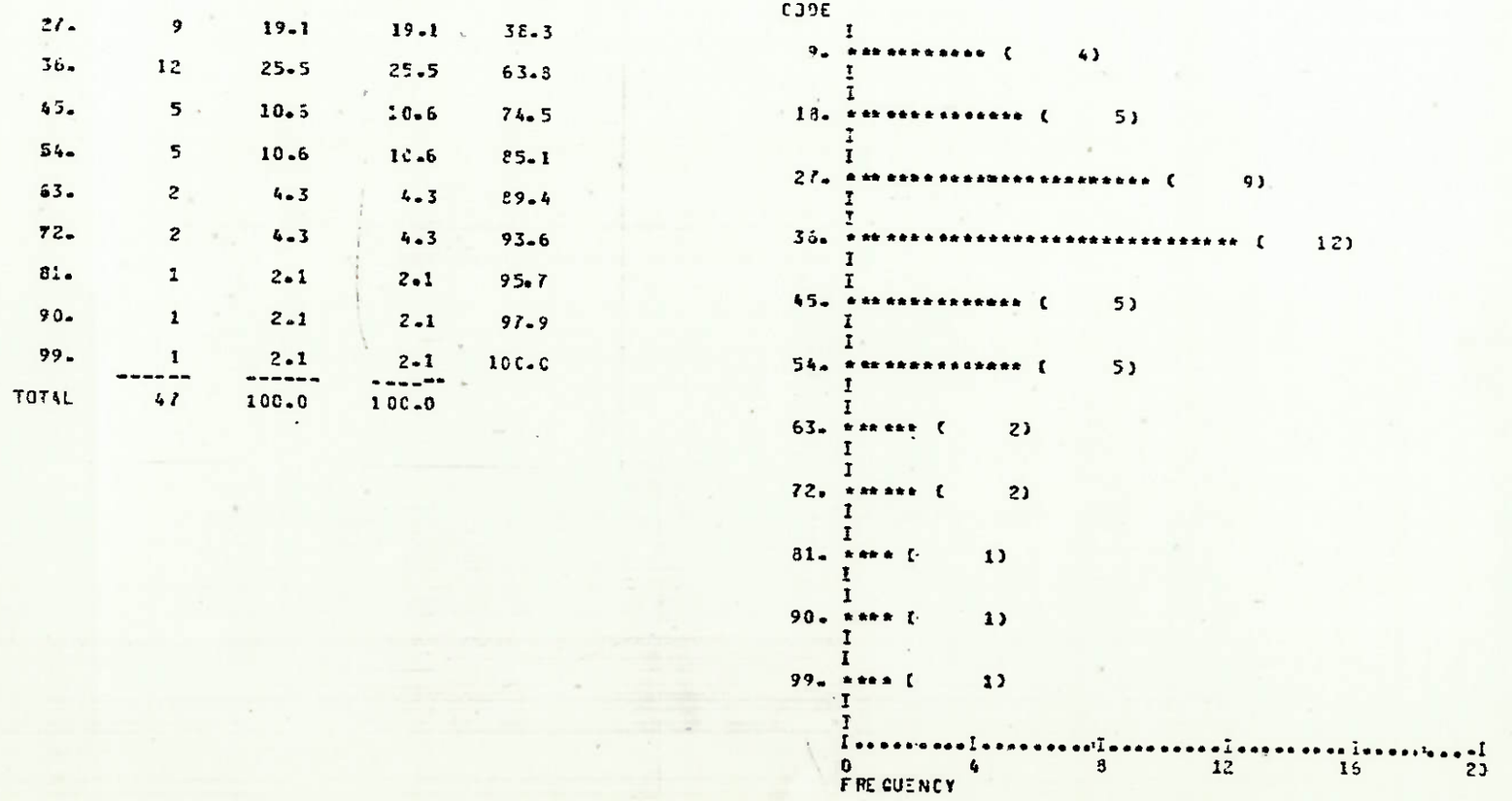
CATEGORY LABEL	CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
	9.	7	20.0	20.0	20.0
	19.	10	28.6	28.6	48.6
	27.	7	20.0	20.0	68.6
	36.	5	14.3	14.3	82.9
	45.	3	8.6	8.6	91.4
	54.	1	2.9	2.9	94.3
	63.	1	2.9	2.9	97.1
	72.	1	2.9	2.9	100.0
	TOTAL	35	100.0	100.0	

CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
9.	4	8.5	8.5	8.5
18.	5	10.6	10.6	19.1
27.	9	19.1	19.1	38.3
36.	12	25.5	25.5	63.8
45.	5	10.6	10.6	74.5
54.	5	10.6	10.6	85.1
63.	2	4.3	4.3	89.4
72.	2	4.3	4.3	93.6
81.	1	2.1	2.1	95.7
90.	1	2.1	2.1	97.9
99.	1	2.1	2.1	100.0
TOTAL	47	100.0	100.0	

TEMPO PARA DESENVOLVIMENTO DE PROGRAMAS ON-LINE SEM SCDA

FILE NONAME (CREATION DATE = 11/06/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

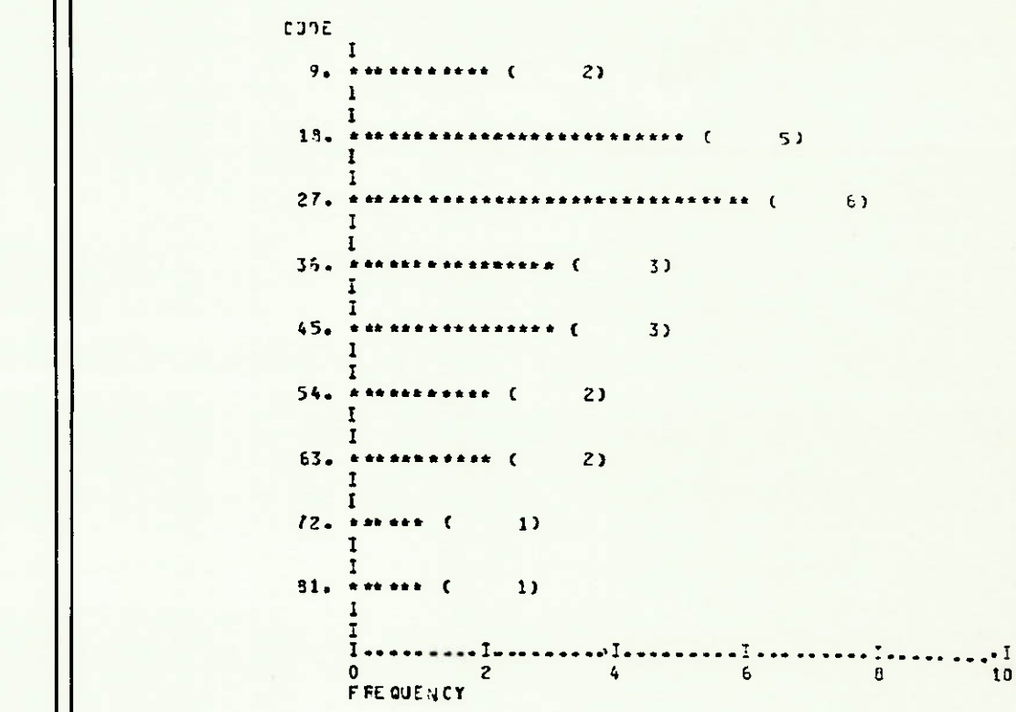


MEAN	39.064	STD ERR	3.027	STD DEV	20.752
VARIANCE	430.626	KURTOSIS	0.860	SKEWNESS	0.931
VALID CASES	47	MISSING CASES	0		

TEMPO PARA DESENV. DE PROGRAMAS EM OUTRAS LINGUAGENS

FILE NONAME (CREATION DATE = 11/29/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



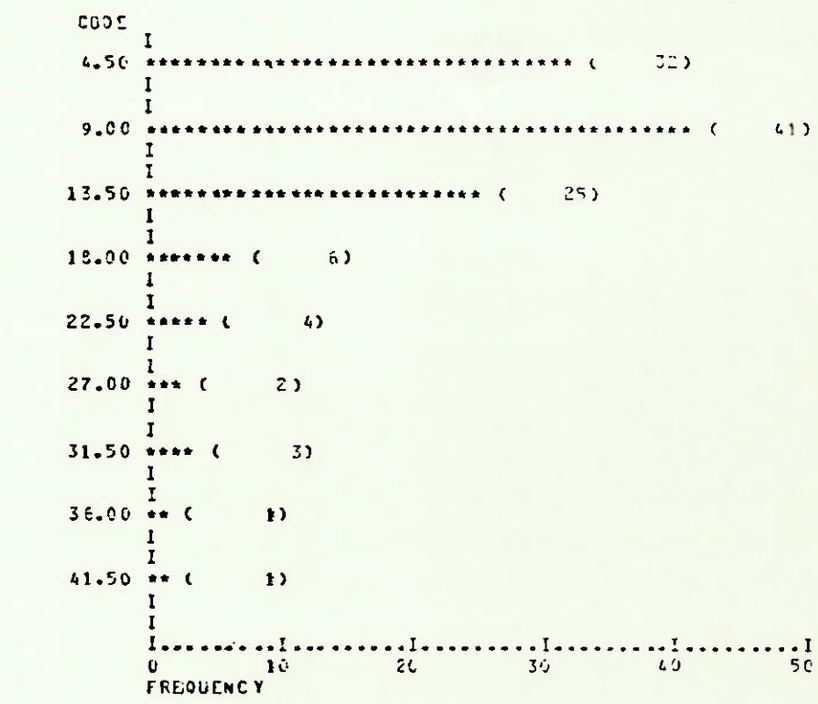
MEAN	36.000	STD ERR	3.923	STD DEV	19.615
VARIANCE	384.750	KURTOSIS	-0.269	SKEWNESS	0.709
VALID CASES	25	MISSING CASES	0		

CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
9.	2	8.0	8.0	8.0
18.	5	20.0	20.0	28.0
27.	6	24.0	24.0	52.0
36.	3	12.0	12.0	64.0
45.	3	12.0	12.0	76.0
54.	2	8.0	8.0	84.0
63.	2	8.0	8.0	92.0
72.	1	4.0	4.0	96.0
81.	1	4.0	4.0	100.0
TOTAL	25	100.0	100.0	

TEMPO PARA MANUTENCAO DE PROGRAMAS BATCH//ATUALIZACAO

FILE NONAME (CREATION DATE = 11/08/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



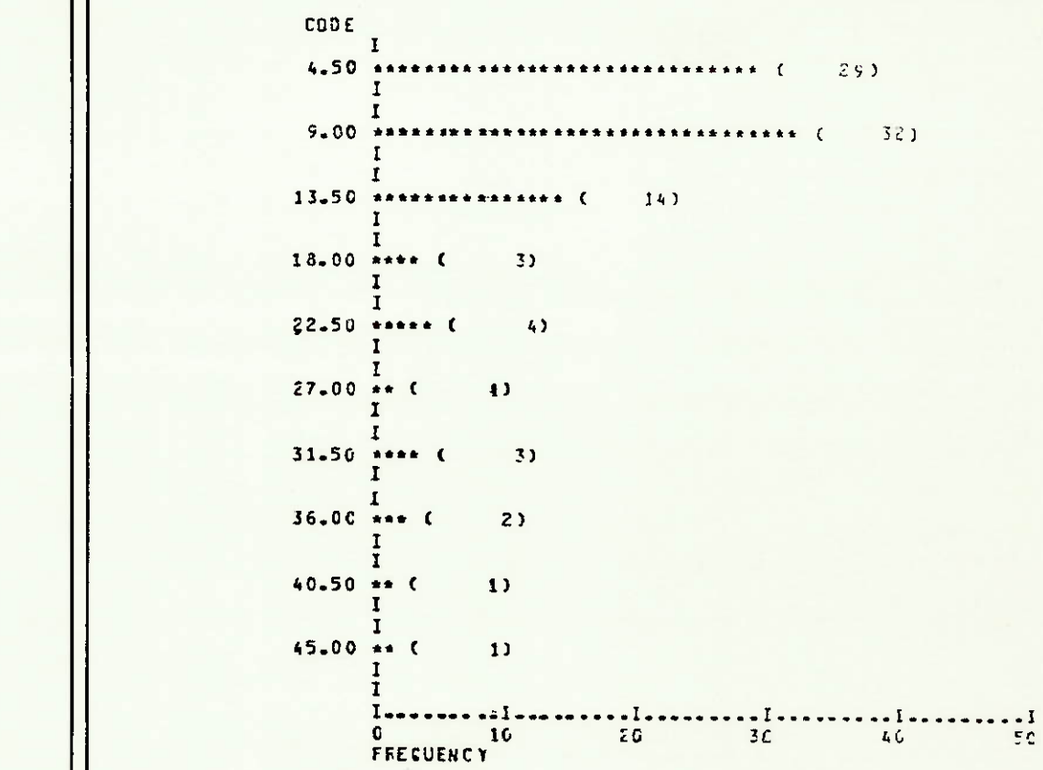
MEAN	11.083	STD ERR	0.660	STD DEV	7.147
VARIANCE	51.074	KURTOSIS	4.131	SKEWNESS	1.859
VALID CASES	115	MISSING CASES	0		

CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
4.50	32	27.8	27.8	27.8
9.00	41	35.7	35.7	63.5
13.50	25	21.7	21.7	85.2
18.00	6	5.2	5.2	90.4
22.50	4	3.5	3.5	93.9
27.00	2	1.7	1.7	95.7
31.50	3	2.6	2.6	98.3
36.00	1	0.9	0.9	99.1
41.50	1	0.9	0.9	100.0
TOTAL	115	100.0	100.0	

TEMPO PARA MANUTENCAO DE PROGRAMAS BATCH/RELATORIO

FILE NONAME (CREATION DATE = 11/09/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



MEAN	11.450	STD ERR	0.928	STD DEV	8.800
VARIANCE	77.433	KURTOSIS	3.678	SKEWNESS	1.956

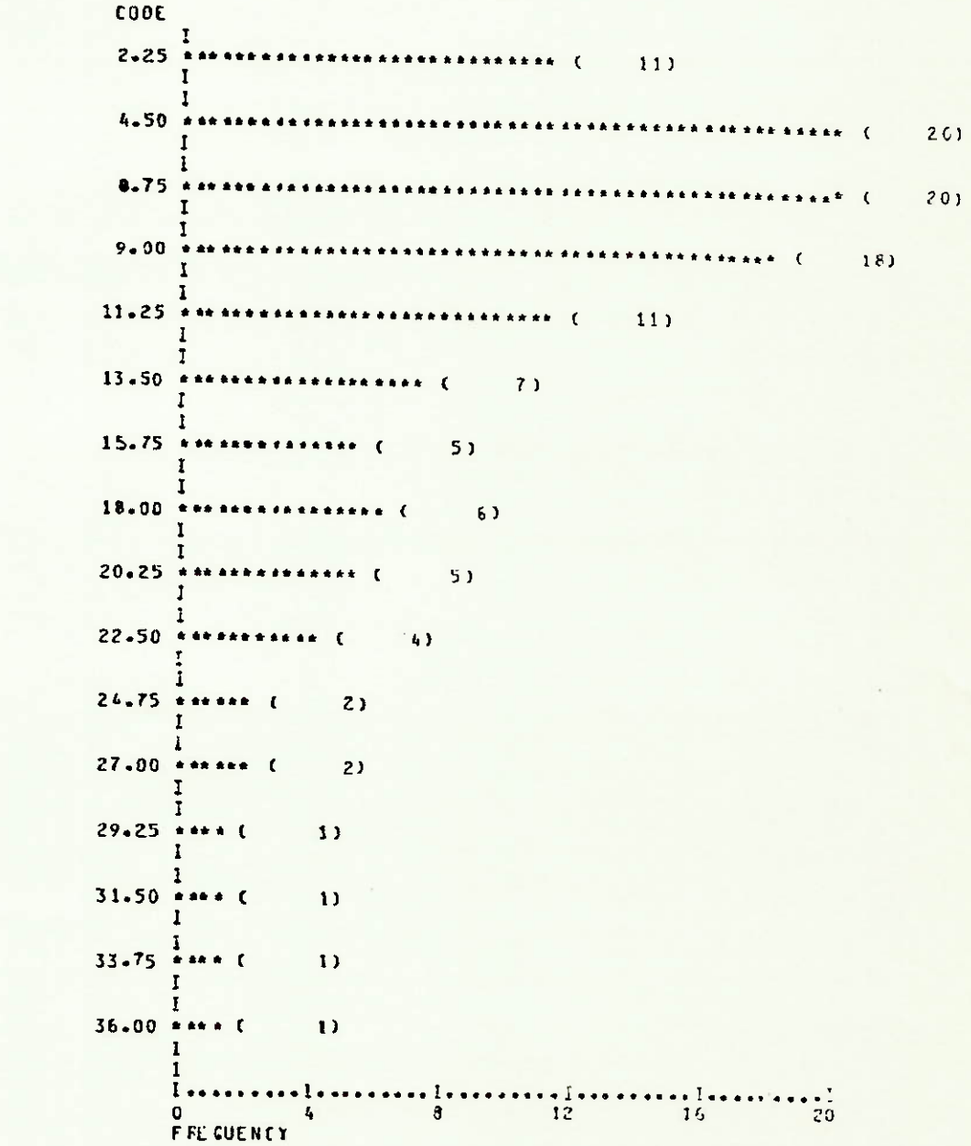
VALID CASES 90 MISSING CASES 0

CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
0.45	29	32.2	32.2	32.2
0.90	32	35.6	35.6	67.8
1.35	14	15.6	15.6	83.3
1.80	3	3.3	3.3	86.7
2.25	4	4.4	4.4	91.1
2.70	1	1.1	1.1	92.2
3.15	3	3.3	3.3	95.6
3.60	2	2.2	2.2	97.8
4.05	1	1.1	1.1	98.9
4.50	1	1.1	1.1	100.0
TOTAL	90	100.0	100.0	

TEMPO PARA MANUTENCAO DE PROGRAMAS ON-LINE COM SCOA

FILE NONAME (CREATION DATE = 11/07/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



MEAN	10.800	STD ERR	0.624	STD DEV	7.440
VARIANCE	55.350	KURTOSIS	1.217	SKEWNESS	1.246
VALID CASES	115	MISSING CASES	0		

TEMPO PARA MANUTENCAO DE PROGRAMAS ON-LINE COM SCOA

FILE NONAME (CREATION DATE = 11/07/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
2.25	11	9.6	9.6	9.6
4.50	20	17.4	17.4	27.0
6.75	20	17.4	17.4	44.3
9.00	18	15.7	15.7	60.0
11.25	11	9.6	9.6	69.6
13.50	7	6.1	6.1	75.7
15.75	5	4.3	4.3	80.0
18.00	6	5.2	5.2	85.2
20.25	5	4.3	4.3	89.6
22.50	4	3.5	3.5	93.0
24.75	2	1.7	1.7	94.8
27.00	2	1.7	1.7	96.5
29.25	1	0.9	0.9	97.4
31.50	1	0.9	0.9	98.3
33.75	1	0.9	0.9	99.1
36.00	1	0.9	0.9	100.0
TOTAL	115	100.0	100.0	

TEMPO PARA MANUTENCAO DE PROGRAMAS ON-LINE SUM SOCA

FILE NONAME (CREATION DATE = 11/07/84)

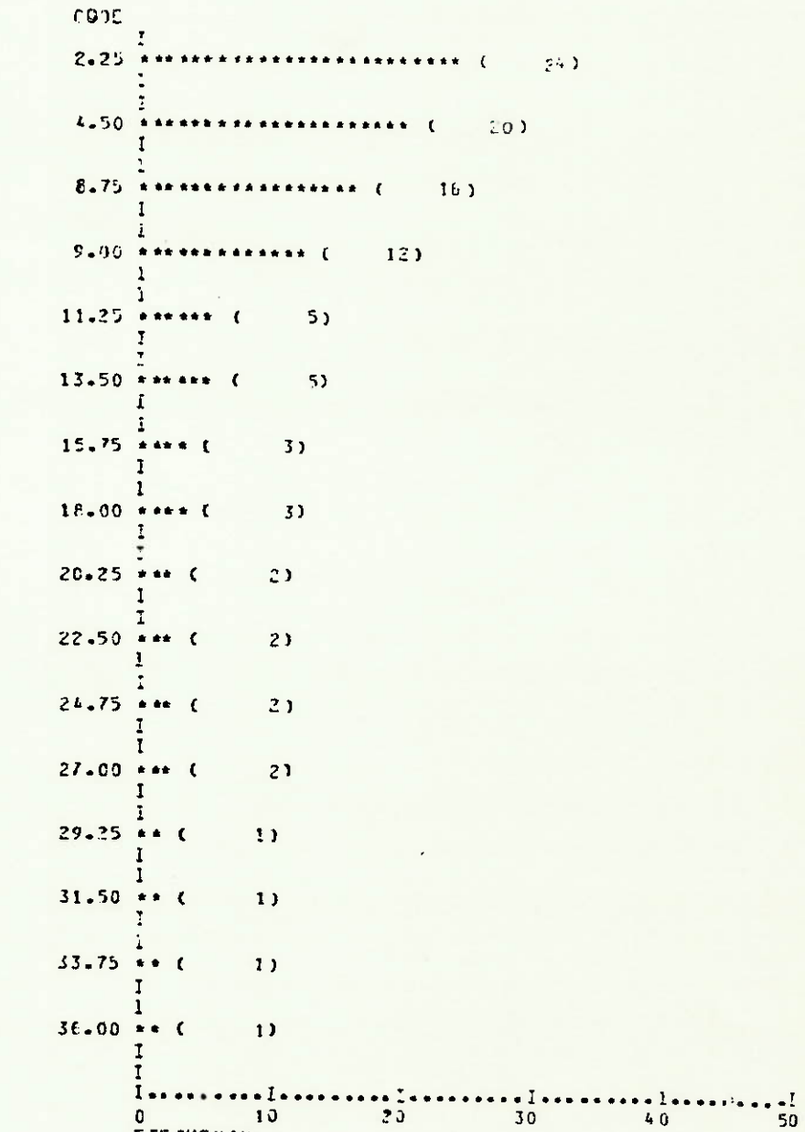
TEMPO TEMPO EFETIVO EM HORAS UTEIS

CATEGORY LABEL	CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
	2.25	24	24.0	24.0	24.0
	4.50	20	20.0	20.0	44.0
	6.75	16	16.0	16.0	60.0
	9.00	12	12.0	12.0	72.0
	11.25	5	5.0	5.0	77.0
	13.50	5	5.0	5.0	82.0
	15.75	3	3.0	3.0	85.0
	18.00	3	3.0	3.0	88.0
	20.25	2	2.0	2.0	90.0
	22.50	2	2.0	2.0	92.0
	24.75	2	2.0	2.0	94.0
	27.00	2	2.0	2.0	96.0
	29.25	1	1.0	1.0	97.0
	31.50	1	1.0	1.0	98.0
	33.75	1	1.0	1.0	99.0
	36.00	1	1.0	1.0	100.0
	TOTAL	100	100.0	100.0	

TEMPO PARA MANUTENCAO DE PROGRAMAS ONLINE (CPM 70)

FILE NONAME (CREATION DATE = 11/07/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

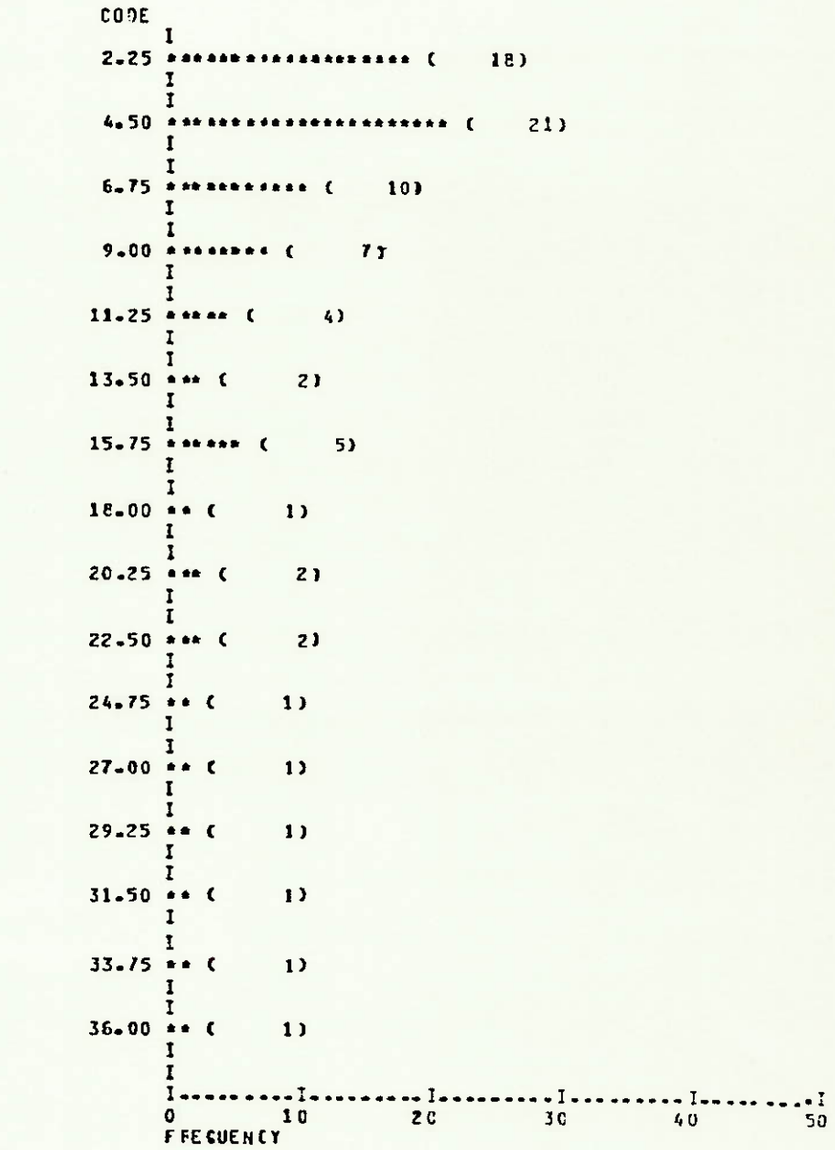


MEAN	9.045	STD ERR	0.785	STD DEV	7.853
VARIANCE	61.668	KURTOSIS	2.087	SKEWNESS	1.600
VALID CASES	100	MISSING CASES	0		

TEMPO PARA MANUTENCAO DE PROGRAMAS EM OUTRAS LINGUAGENS

FILE NONAME (CREATION DATE = 11/07/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS



MEAN	9.000	STD ERR	0.923	STD DEV	8.149
VARIANCE	66.404	KURTOSIS	2.165	SKEWNESS	1.650
VALID CASES	78	MISSING CASES	0		

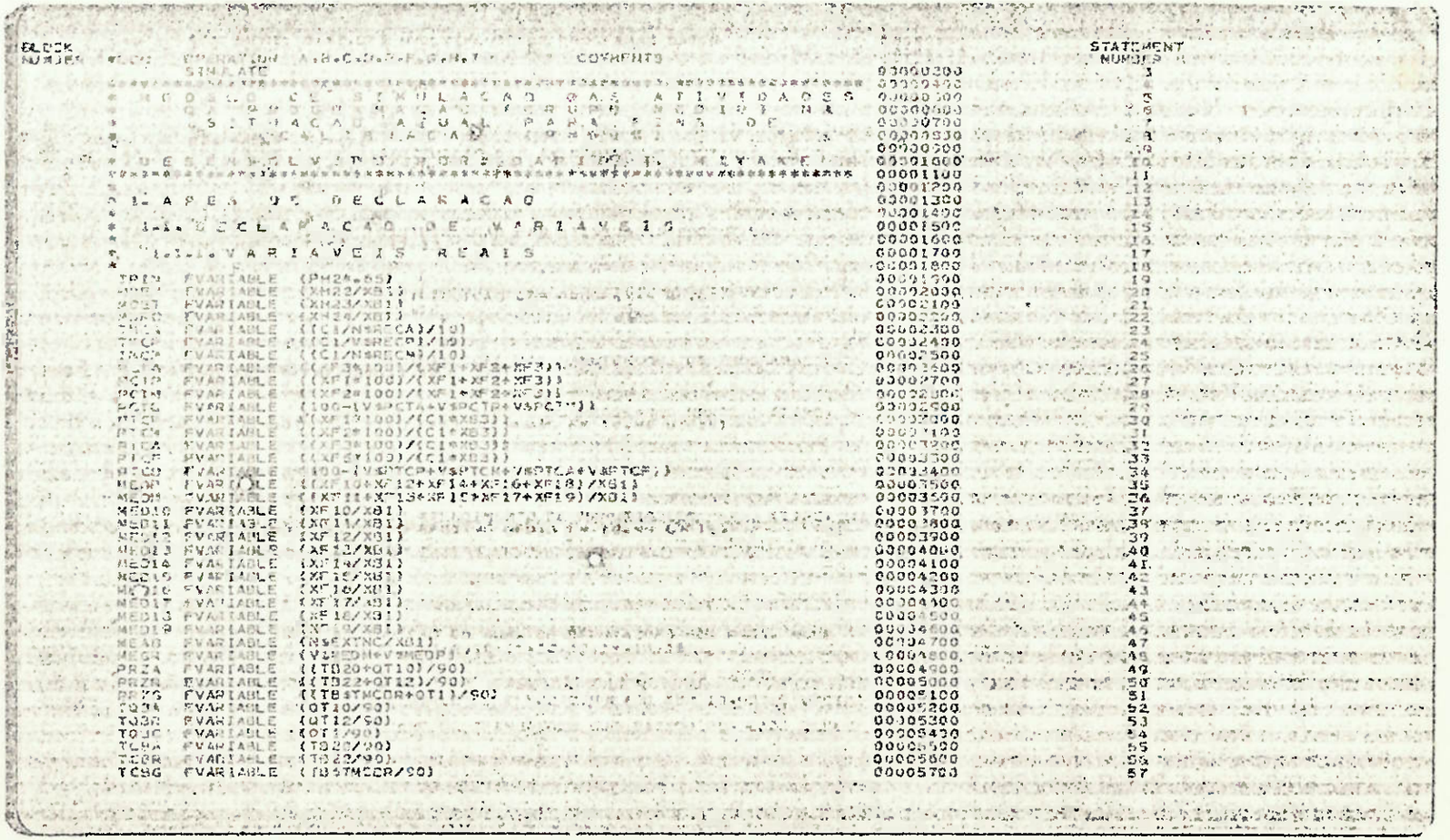
TEMPO PARA MANUTENCAO DE PROGRAMAS EM OUTRAS LINGUAGENS

FILE NONAME (CREATION DATE = 11/07/84)

TEMPO TEMPO EFETIVO EM HORAS UTEIS

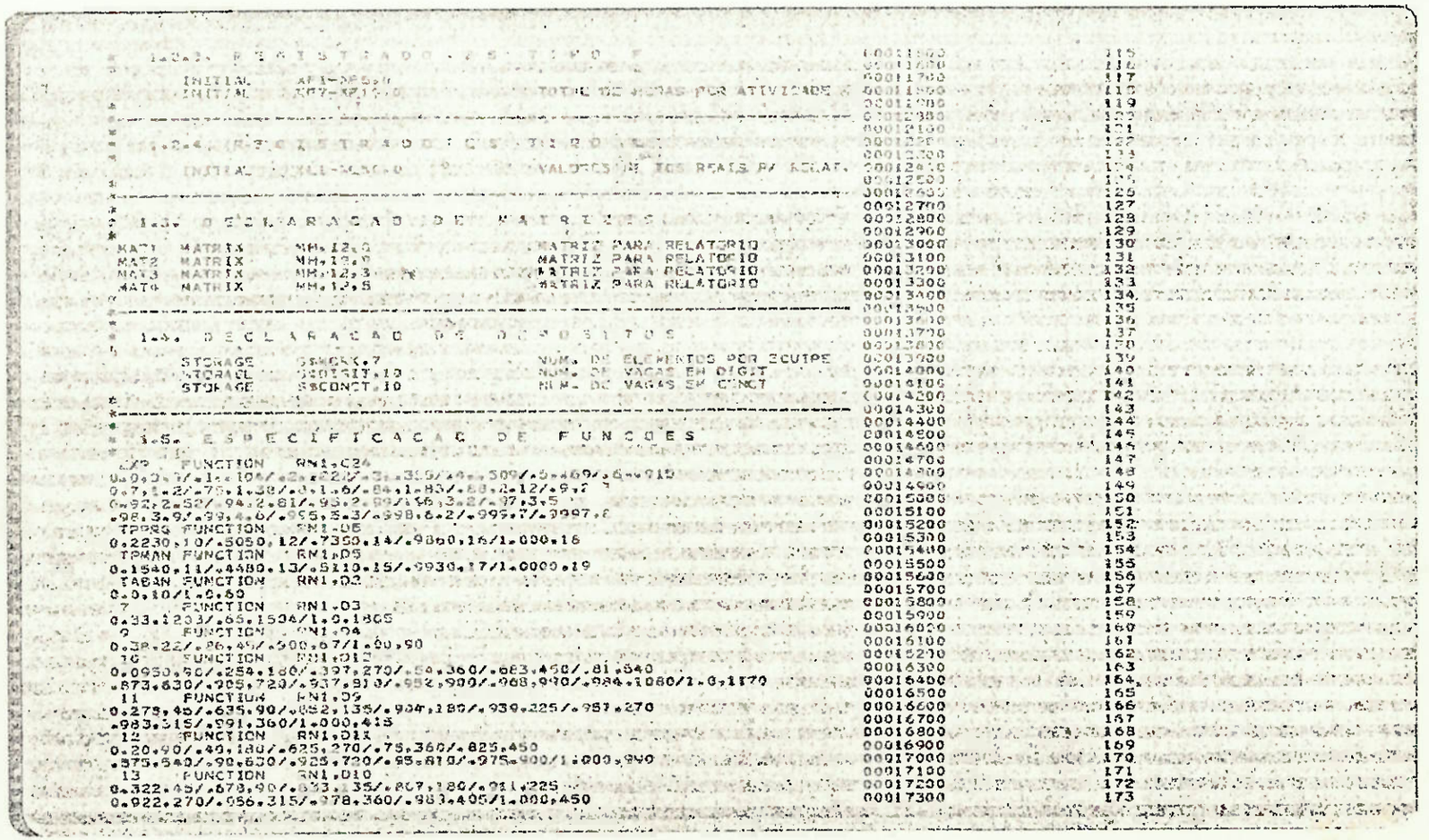
CATEGORY LABEL	CODE	ABSOLUTE FREQ	RELATIVE FREQ (PCT)	ADJUSTED FREQ (PCT)	CUM FREQ (PCT)
	2.25	18	23.1	23.1	23.1
	4.50	21	26.9	26.9	50.0
	6.75	10	12.8	12.8	62.8
	9.00	7	9.0	9.0	71.8
	11.25	4	5.1	5.1	76.9
	13.50	2	2.6	2.6	79.5
	15.75	5	6.4	6.4	85.9
	18.00	1	1.3	1.3	87.2
	20.25	2	2.6	2.6	89.7
	22.50	2	2.6	2.6	92.3
	24.75	1	1.3	1.3	93.6
	27.00	1	1.3	1.3	94.9
	29.25	1	1.3	1.3	96.2
	31.50	1	1.3	1.3	97.4
	33.75	1	1.3	1.3	98.7
	36.00	1	1.3	1.3	100.0
	TOTAL	78	100.0	100.0	

ANEXO II - PROGRAMA FONTE PARA A SIMULAÇÃO DA SITUAÇÃO ATUAL

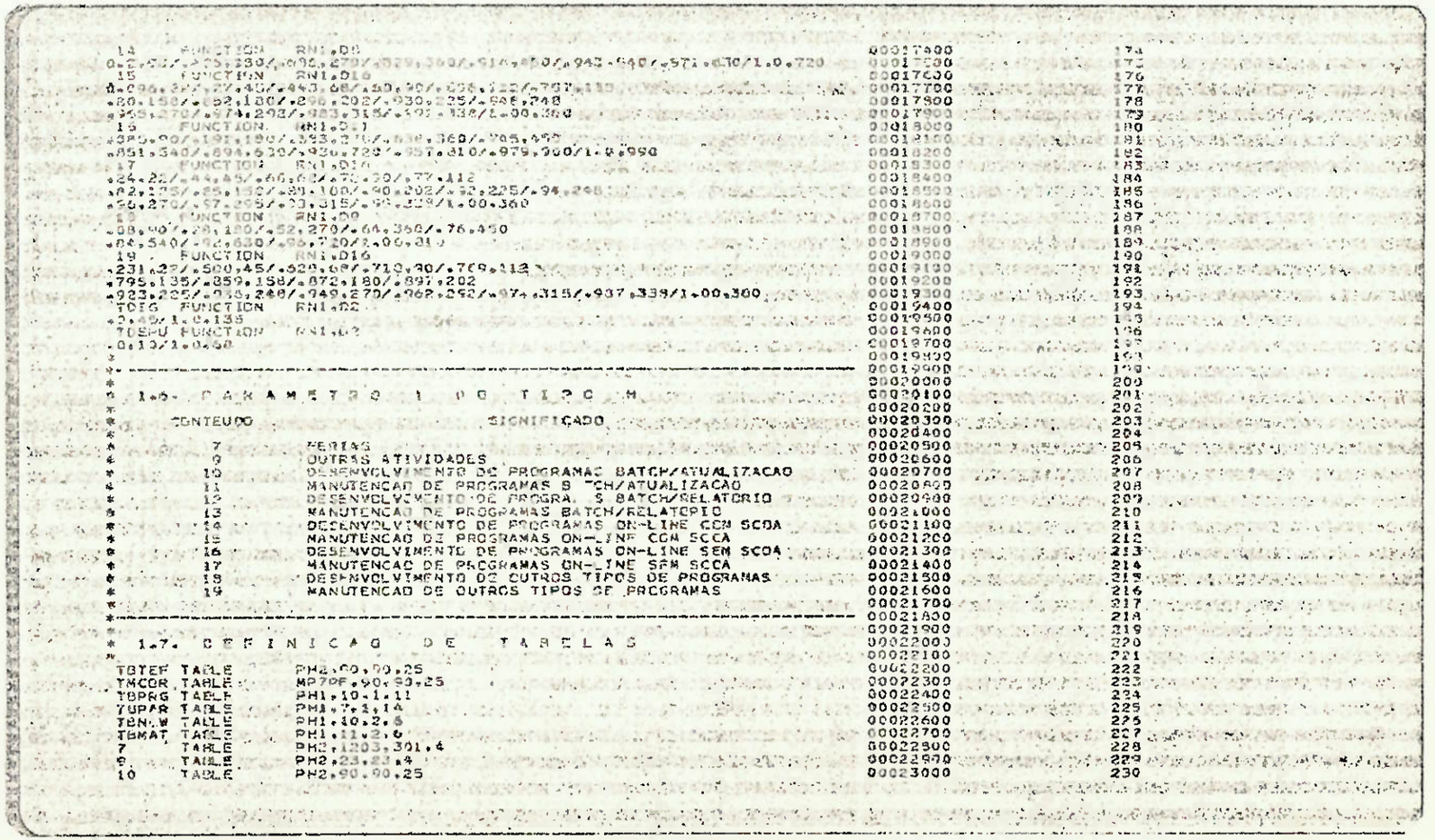


```
BLOCK
NUMBER  NEST   OPERATION  A.B.C.D.E.F.G.H.I        COMMENTS                                  STATEMENT
                SINTATE                                                                      NUMBER
                                                                                          00000200      3
*****************************************************************************************00000300      4
*  M O D E L O   D E   S I M U L A C A O   D A S   A T I V I D A D E S                     00000400      5
*        P R O G R A M A   P A R A   G R U P O   M O D E L O   N A                         00000500      6
*     S I T U A C A O   A T U A L   P A R A   E N S   D E                                  00000600      7
*           F I N A L   D A   A C A O   C O M E R C I A L                                  00000700      8
*                                                                                          00000800      9
*  U E S   -   C L I E N T E   C O R P O R A T I V O   -   M Y A N T E                     00000900     10
*****************************************************************************************00001000     11
                                                                                          00001100     12
* 1. A P E A   D E   D E C L A R A C A O                                                   00001200     13
*                                                                                          00001300     14
*   1.1. D E C L A R A C A O   D E   V A R I A V E I S                                     00001400     15
*                                                                                          00001500     16
*     1.1.1. V A R I A V E I S   R E A I S                                                 00001600     17
*                                                                                          00001700     18
 TREA   FVARIABLE  (PM2A.65)                                                              00001800     19
 XM22   FVARIABLE  (XM22/XB1)                                                             00001900     20
 XM23   FVARIABLE  (XM23/XB1)                                                             00002000     21
 XM24   FVARIABLE  (XM24/XB1)                                                             00002100     22
 TRCA   FVARIABLE  ((C1/NSRCCA)/10)                                                       00002200     23
 TRCO   FVARIABLE  ((C1/VSRECO)/10)                                                       00002300     24
 TRCN   FVARIABLE  ((C1/NSRECN)/10)                                                       00002400     25
 PCTA   FVARIABLE  ((XF3*100)/(XF1+XF2+XF3))                                              00002500     26
 PCTO   FVARIABLE  ((XF1*100)/(XF1+XF2+XF3))                                              00002600     27
 PCTN   FVARIABLE  ((XF2*100)/(XF1+XF2+XF3))                                              00002700     28
 PCTC   FVARIABLE  ((100-(VSPCTA+VSPCTP+VSPCT**))                                         00002800     29
 MTCO   FVARIABLE  ((XF1*100)/(C1*XB3))                                                   00002900     30
 PTCN   FVARIABLE  ((XF2*100)/(C1*XB3))                                                   00003000     31
 PTCA   FVARIABLE  ((XF3*100)/(C1*XB3))                                                   00003100     32
 PTCP   FVARIABLE  ((XF5*100)/(C1*XB3))                                                   00003200     33
 PTCO   FVARIABLE  (100-(VSPTCP+VSPTCN+VSPTCA+VSPTCP))                                    00003300     34
 MEDIA  FVARIABLE  ((XF10+XF12+XF14+XF16+XF18)/XB1)                                       00003400     35
 MEDIA  FVARIABLE  ((XF11+XF13+XF15+XF17+XF19)/XB1)                                       00003500     36
 MED10  FVARIABLE  (XF10/XB1)                                                             00003600     37
 MED11  FVARIABLE  (XF11/XB1)                                                             00003700     38
 MED12  FVARIABLE  (XF12/XB1)                                                             00003800     39
 MED13  FVARIABLE  (XF13/XB1)                                                             00003900     40
 MED14  FVARIABLE  (XF14/XB1)                                                             00004000     41
 MED15  FVARIABLE  (XF15/XB1)                                                             00004100     42
 MED16  FVARIABLE  (XF16/XB1)                                                             00004200     43
 MED17  FVARIABLE  (XF17/XB1)                                                             00004300     44
 MED18  FVARIABLE  (XF18/XB1)                                                             00004400     45
 MED19  FVARIABLE  (XF19/XB1)                                                             00004500     46
 MEA0   FVARIABLE  (NSEXTNC/XB1)                                                          00004600     47
 MEDR   FVARIABLE  (VSMEDP/XB1)                                                           00004700     48
 PRZA   FVARIABLE  ((TD20+QT10)/90)                                                       00004800     49
 PRZN   FVARIABLE  ((TD22+QT12)/90)                                                       00004900     50
 PRZO   FVARIABLE  ((TB4TMCOR+QT1)/90)                                                    00005000     51
 TQ1A   FVARIABLE  (QT10/90)                                                              00005100     52
 TQ2R   FVARIABLE  (QT12/90)                                                              00005200     53
 TQ1C   FVARIABLE  (QT1/90)                                                               00005300     54
 TC9A   FVARIABLE  (TD20/90)                                                              00005400     55
 TC9R   FVARIABLE  (TD22/90)                                                              00005500     56
 TC9C   FVARIABLE  (TB4TMCOR/90)                                                          00005600     57
```

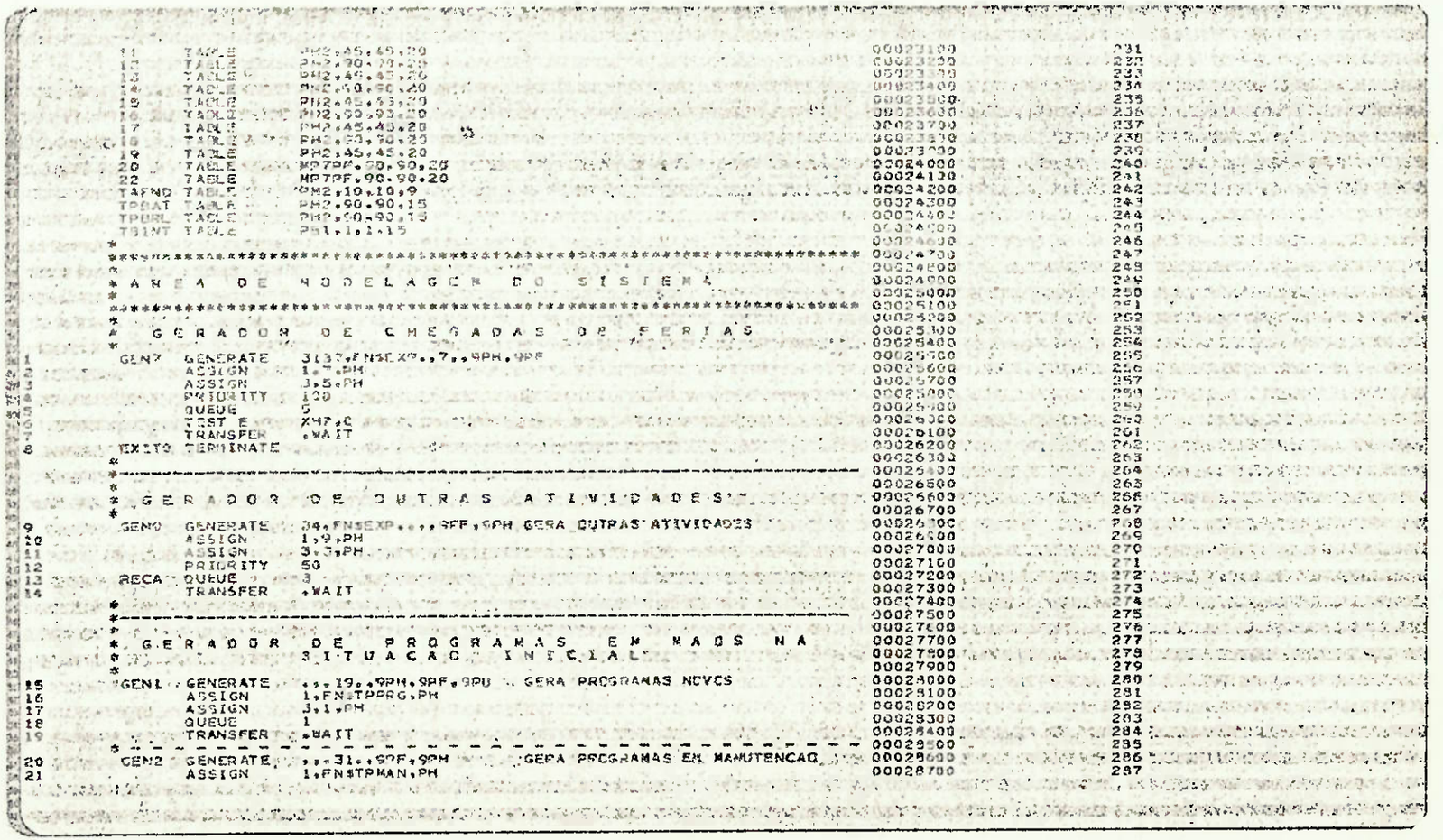
```
TEMA   FVARIABLE  (TS(10/90)                                                    00005000   59
TEMV   FVARIABLE  (TS(12/90)                                                    00005100   59
TEMG   FVARIABLE  (TS$TBTEP/90)                                                 00005200   60
*                                                                               00005300   61
*--------------------------------------------------------------------------     00005400   62
*                                                                               00005500   63
*    1.1.2. V A R I A V E I S   I N T E I R A S                                 00005600   64
*                                                                               00005700   65
MES    VARIABLE   (P$182)                                                       00005800   66
PLAST  VARIABLE   (PH2-PH5+PH4)                                                 00005900   67
DIAMF  VARIABLE   (((C1+90)/90)*90-C1)                                          00006000   68
TUTU   VARIABLE   (01+02+03)                                                    00006100   69
PNEA   VARIABLE   (XH10+XH12+XH14+XH16+XH18)                                    00006200   70
PNEM   VARIABLE   (01+XH10+XH12+XH14+XH16+XH18+06+07+CENCT+SSDIG(T)             00006300   71
MPEA   VARIABLE   (XH11+XH13+XH15+XH17+XH19)                                    00006400   72
MPEM   VARIABLE   (02)V$MPEA)                                                   00006500   73
TATA   VARIABLE   (XH7+XH9+V$PNEA+V$MPEA)                                       00006600   74
TLEA   VARIABLE   (V$PNEA+V$MPEA)                                               00006700   75
TLEM   VARIABLE   (V$PNEM+V$MPEM)                                               00006800   76
TPEN   VARIABLE   (N$EXIT1+N$EXIT5)                                             00006900   77
TMEN   VARIABLE   (N$EXIT2+N$EXIT4)                                             00007000   78
NPEM   VARIABLE   (XH8+01+02)                                                   00007100   79
PENT   VARIABLE   (N$EXIT1+N$EXIT5-XH5)                                         00007200   80
MENT   VARIABLE   (N$EXIT2+N$EXIT4-XH6)                                         00007300   81
TENT   VARIABLE   (V$PENT+V$MENT)                                               00007400   82
DEMP   VARIABLE   (V$RECP-XH20)                                                 00007500   83
DEMA   VARIABLE   (N$RECM-XH21)                                                 00007600   84
DENT   VARIABLE   (V$DEMP+V$DEMA)                                               00007700   85
RECP   VARIABLE   (N$RCP10+N$RCP12+N$RCP14+N$RCP16+N$RCP18)                     00007800   86
TEFC   VARIABLE   (PH1-10)                                                      00007900   87
*                                                                               00008000   88
*                                                                               00008100   89
*--------------------------------------------------------------------------     00008200   90
*                                                                               00008300   91
*  1.2. D E C L A R A C A O   D E   R E G I S T R A D O R E S                    00008400   92
*                                                                               00008500   93
*    1.2.1. R E G I S T R A D O R E S   T I P O   B                             00008600   94
*                                                                               00008700   95
      INITIAL    X81.12          NUMERO DE MESES SIMULADOS                    00008800   96
      INITIAL    X82.0           CONTADOR DE MESES                            00008900   97
      INITIAL    X83.7           NUM. DE PROGRAMADORES                        00009000   98
*                                                                               00009100   99
*--------------------------------------------------------------------------     00009200  100
*                                                                               00009300  101
*    1.2.2. R E G I S T R A D O R E S   T I P O   H                             00009400  102
*                                                                               00009500  103
      INITIAL    XH5.0           GUARDA V$TPEN DO MES ANTERIOR                00009600  104
      INITIAL    XH6.0           "    V$TMEN  "  "    "                       00009700  105
      INITIAL    XH7-XH19.0      NUM. DE ATIVIDADES EM ATENDI/                00009800  106
      INITIAL    XH20.0          GUARDA V$RECP DO MES ANTERIOR                00009900  107
      INITIAL    XH21.0          "    V$RCOM  "  "    "                       00010000  108
      INITIAL    XH22.0          ACUMULA V$TLEM TODO MES                      00010100  109
      INITIAL    XH23.0          "    V$PENT  "  "                           00011000  110
      INITIAL    XH24.0          "    V$DENT  "  "                           00011100  111
*                                                                               00011200  112
*--------------------------------------------------------------------------     00011300  113
*                                                                               00011400  114
```



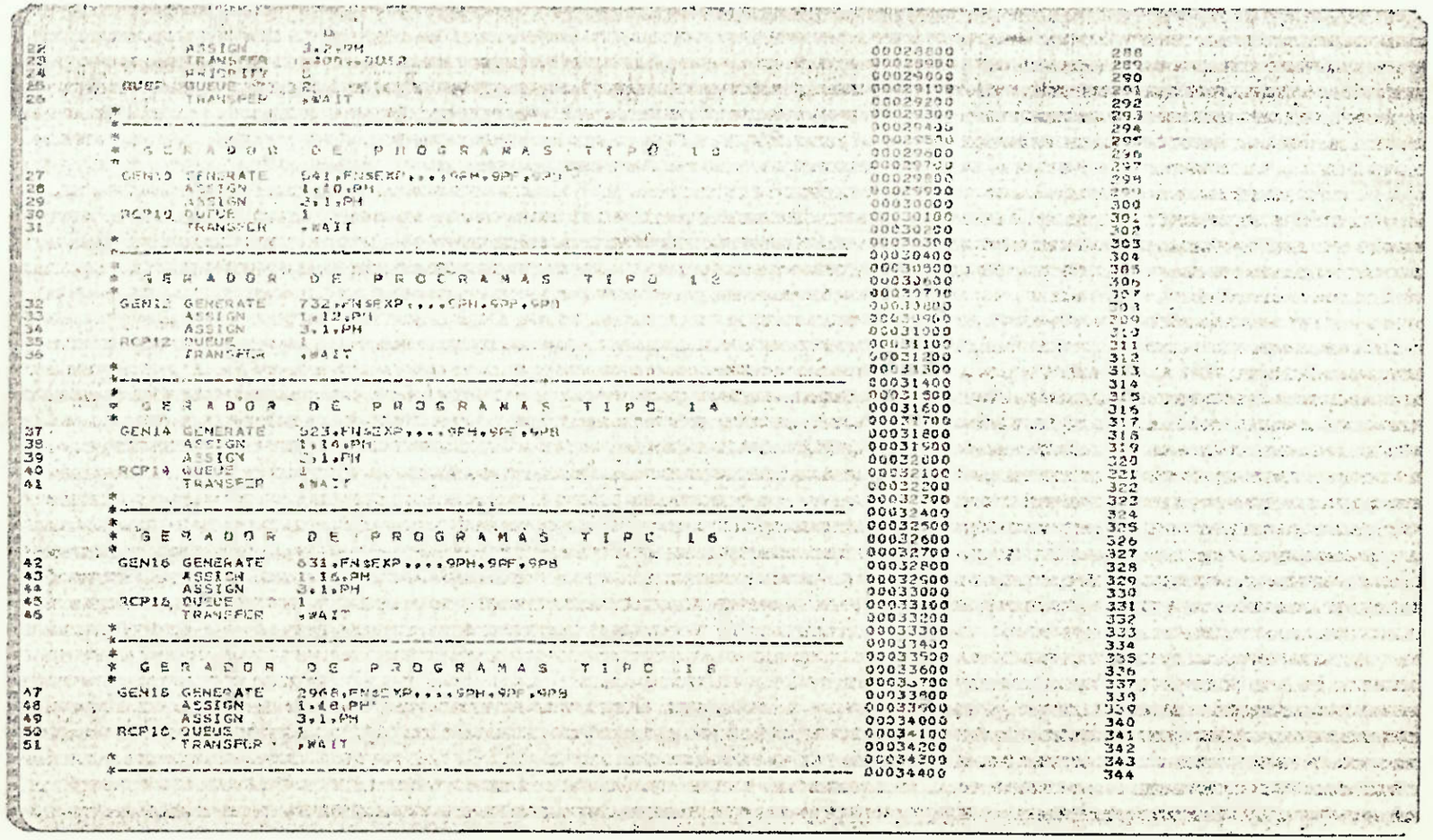
```
*  1.2.3. REGISTRADO AS TIPO F                                          00011500
*                                                                       00011600
         INITIAL    AF1=2F5.0                                          00011700
         INITIAL    AF7=XF1.0          TOTAL DE HORAS POR ATIVIDADE    00011800
*                                                                       00011900
*-------------------------------------------------------------------    00012000
*                                                                       00012100
*  1.2.4. REGISTRADO AS TIPO R                                          00012200
*                                                                       00012300
         INITIAL    AR1=2F5.0          VALORES EM PONTOS REAIS P/ RELAT. 00012400
*                                                                       00012500
*-------------------------------------------------------------------    00012600
*                                                                       00012700
*  1.3. DECLARACAO DE MATRIZES                                          00012800
*                                                                       00012900
MAT1   MATRIX     MM=12,0            MATRIZ PARA RELATORIO              00013000
MAT2   MATRIX     MM=12,0            MATRIZ PARA RELATORIO              00013100
MAT3   MATRIX     MM=12,3            MATRIZ PARA RELATORIO              00013200
MAT4   MATRIX     MM=12,5            MATRIZ PARA RELATORIO              00013300
*                                                                       00013400
*-------------------------------------------------------------------    00013500
*                                                                       00013600
*  1.4. DECLARACAO DE DEPOSITOS                                         00013700
*                                                                       00013800
         STORAGE   SMACRX,7          NUM. DE ELEMENTOS POR EQUIPE       00013900
         STORAGE   SDEPSIT,10        NUM. DE VAGAS EM DEPOT             00014000
         STORAGE   SSCONCT,10        NUM. DE VAGAS EM CONCT             00014100
*                                                                       00014200
*-------------------------------------------------------------------    00014300
*                                                                       00014400
*  1.5. ESPECIFICACAO DE FUNCOES                                        00014500
*                                                                       00014600
*2    FUNCTION   RN1,C24                                                00014700
0.0,0.37/.1,.104/.2,.224/.3,.349/.4,.509/.5,.669/.6,.815               00014800
0.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12/.9,2                         00014900
0.92,2.52/.94,2.81/.96,3.2/.98,3.8/.97,3.5                             00015000
.98,3.9/.99,4.6/.995,5.3/.998,6.2/.999,7/.9997,8                       00015100
TPPRO FUNCTION   RN1,D6                                                00015200
0.2230,10/.5050,12/.7350,14/.980,16/1.000,16                           00015300
TPMAN FUNCTION   RN1,D5                                                00015400
0.1540,11/.4480,13/.5110,15/.9930,17/1.0000,19                         00015500
TABAN FUNCTION   RN1,D2                                                00015600
0.0,10/1.0,60                                                          00015700
7     FUNCTION   RN1,D3                                                00015800
0.33,1203/.66,1504/1.0,1805                                            00015900
9     FUNCTION   RN1,D4                                                00016000
0.38,22/.86,40/.900,67/1.00,90                                         00016100
10    FUNCTION   RN1,D12                                               00016200
0.0950,56/.254,180/.397,270/.54,360/.683,450/.81,540                   00016300
.873,630/.905,720/.937,810/.952,900/.968,990/.984,1080/1.0,1170        00016400
11    FUNCTION   RN1,D9                                                00016500
0.278,45/.635,90/.852,135/.934,180/.939,225/.957,270                   00016600
.983,315/.991,360/1.000,415                                            00016700
12    FUNCTION   RN1,D11                                               00016800
0.20,90/.40,180/.625,270/.75,360/.825,450                              00016900
.875,540/.90,630/.925,720/.95,810/.975,900/1.000,940                   00017000
13    FUNCTION   RN1,D10                                               00017100
0.322,45/.678,90/.833,135/.907,180/.911,225                            00017200
0.922,270/.956,315/.978,360/.983,405/1.000,450                         00017300
```



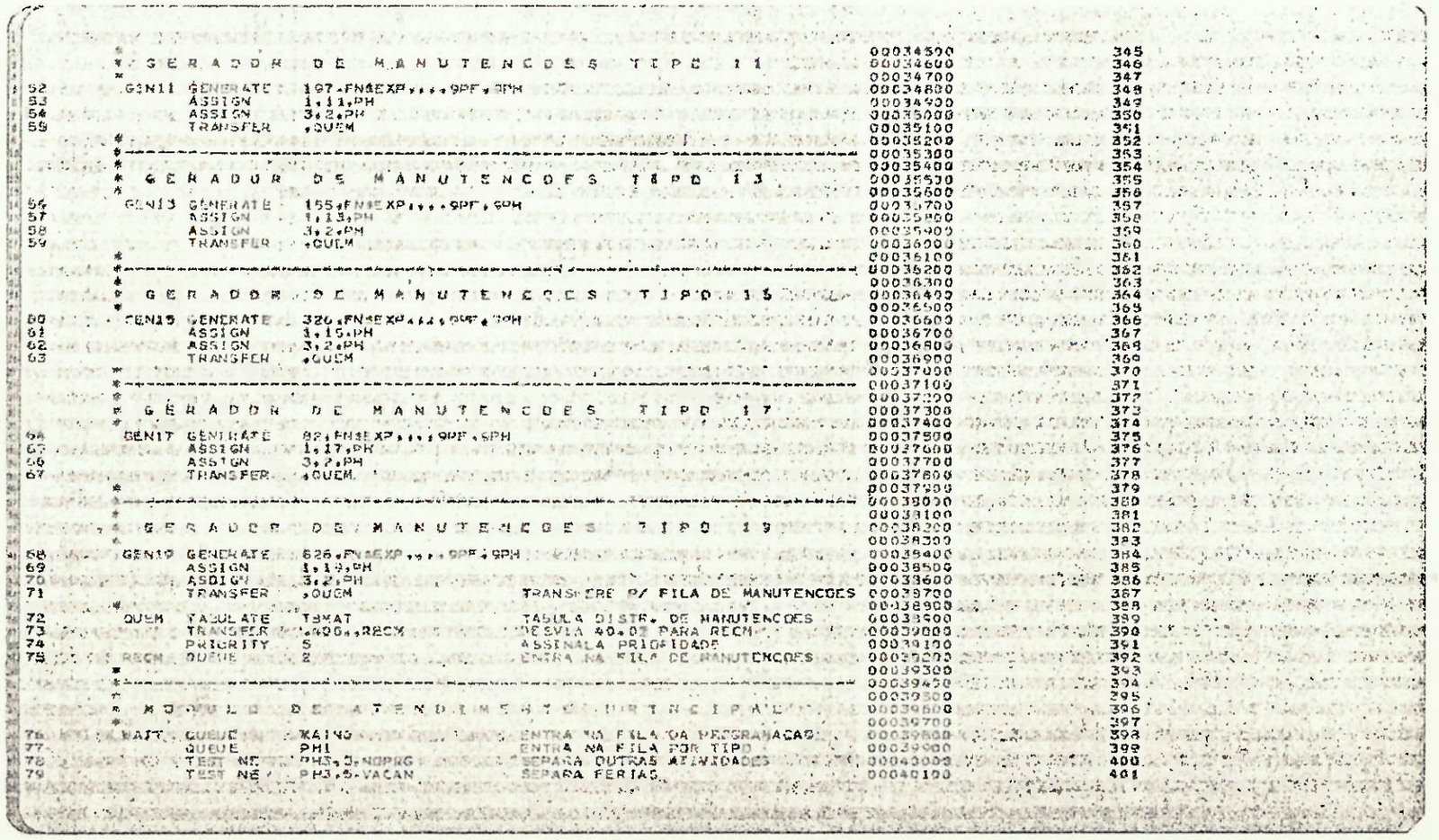
```
 14      FUNCTION    RN1,D8                                                  00017800                  174
0,2,.33/,.25,.330/,.096,.270/,.029,.360/,.914,.450/,.943,.540/,.971,.630/1.0,.720  00017900          175
 15      FUNCTION    RN1,D16                                                 00018000                  176
0,.096,.27/,.27,.405/,.443,.68/,.50,.90/,.606,1.22/,.737,1.35               00018100                  177
,.80,1.58/,.82,1.80/,.896,2.02/,.930,2.25/,.986,2.48                        00018200                  178
,.995,2.70/,.97,4.20/,.983,3.15/,.991,4.38/1.00,4.50                        00018300                  179
 16      FUNCTION    RN1,D.1                                                 00018400                  180
,.895,.90/,.191,.180/,.393,.27/,.638,.360/,.785,.450                        00018500                  181
,.851,.540/,.894,.630/,.930,.720/,.957,.810/,.979,.900/1.0,.990             00018600                  182
 17      FUNCTION    RN1,D16                                                 00018700                  183
,.24,.22/,.44,.45/,.66,.68/,.76,.90/,.77,.112                               00018800                  184
,.82,.135/,.85,.158/,.80,.180/,.90,.202/,.92,.225/,.94,.248                 00018900                  185
,.96,.270/,.97,.293/,.98,.315/,.99,.339/1.00,.360                           00019000                  186
 18      FUNCTION    RN1,D8                                                  00019100                  187
,.08,.90/,.28,.180/,.52,.270/,.64,.360/,.76,.450                            00019200                  188
,.84,.540/,.92,.630/,.96,.720/1.00,.810                                     00019300                  189
 19      FUNCTION    RN1,D16                                                 00019400                  190
,.231,.22/,.500,.45/,.529,.68/,.710,.90/,.769,.112                          00019500                  191
,.795,.135/,.859,.158/,.872,.180/,.897,.202                                 00019600                  192
,.923,.225/,.938,.248/,.949,.270/,.962,.292/,.97,.315/,.987,.339/1.00,.300  00019700                  193
 TOFG    FUNCTION    RN1,D2                                                  00019800                  194
,.5,.49/1.0,.135                                                            00019900                  195
 TOSPU   FUNCTION    RN1,D2                                                  00020000                  196
,.0,.13/1.0,.50                                                             00020100                  197
*                                                                           00020200                  198
*-------------------------------------------------------------------------  00020300                  199
*                                                                           00020400                  200
*  1.6.  P A R A M E T R O  1  D O  T I P O  M                              00020500                  201
*                                                                           00020600                  202
*     CONTEUDO                    SIGNIFICADO                               00020700                  203
*                                                                           00020800                  204
*          7      FERIAS                                                    00020900                  205
*          9      OUTRAS ATIVIDADES                                         00021000                  206
*         10      DESENVOLVIMENTO DE PROGRAMAS BATCH/ATUALIZACAO            00021100                  207
*         11      MANUTENCAO DE PROGRAMAS B TCH/ATUALIZACAO                 00021200                  208
*         12      DESENVOLVIMENTO DE PROGRA. S BATCH/RELATORIO              00021300                  209
*         13      MANUTENCAO DE PROGRAMAS BATCH/RELATORIO                   00021400                  210
*         14      DESENVOLVIMENTO DE PROGRAMAS ON-LINE COM SCDA             00021500                  211
*         15      MANUTENCAO DE PROGRAMAS ON-LINE COM SCCA                  00021600                  212
*         16      DESENVOLVIMENTO DE PROGRAMAS ON-LINE SEM SCOA             00021700                  213
*         17      MANUTENCAO DE PROGRAMAS ON-LINE SEM SCCA                  00021800                  214
*         18      DESENVOLVIMENTO DE OUTROS TIPOS DE PROGRAMAS              00021900                  215
*         19      MANUTENCAO DE OUTROS TIPOS DE PROGRAMAS                   00022000                  216
*                                                                           00022100                  217
*-------------------------------------------------------------------------  00022200                  218
*                                                                           00022300                  219
*  1.7.  D E F I N I C A O  D E  T A B E L A S                              00022400                  220
*                                                                           00022500                  221
 TBTEF   TABLE       PH2,90,90,25                                            00022600                  222
 TMCOR   TABLE       MP70F,90,90,25                                          00022700                  223
 TBPRG   TABLE       PH1,10,1,11                                             00022800                  224
 TUPAR   TABLE       PH1,7,1,14                                              00022900                  225
 TBNOW   TABLE       PH1,10,2,6                                              00022600                  226
 TBMAT   TABLE       PH1,11,2,6                                              00022700                  227
 7       TABLE       PH2,1203,301,4                                          00022800                  228
 9       TABLE       PH2,23,23,4                                             00022900                  229
 10      TABLE       PH2,90,90,25                                            00023000                  230
```



```
                11      TABLE     PH2,45,45,20                                  00023100                  231
                12      TABLE     PH2,70,70,20                                  00023200                  232
                13      TABLE     PH2,45,45,20                                  00023300                  233
                14      TABLE     PH2,50,50,20                                  00023400                  234
                15      TABLE     PH2,45,55,20                                  00023500                  235
                16      TABLE     PH2,90,90,20                                  00023600                  236
                17      TABLE     PH2,45,45,20                                  00023700                  237
                18      TABLE     PH2,50,50,20                                  00023800                  238
                19      TABLE     PH2,45,45,20                                  00023900                  239
                20      TABLE     MR7PF,50,90,20                                00024000                  240
                22      TABLE     MR7PF,90,90,20                                00024100                  241
                TAFND   TABLE     PH2,10,10,9                                   00024200                  242
                TPBAT   TABLE     PH2,90,90,15                                  00024300                  243
                TPBRL   TABLE     PH2,60,90,15                                  00024400                  244
                TBINT   TABLE     P$1,1,1,15                                    00024500                  245
                *                                                               00024600                  246
                *****************************************************************00024700                  247
                *                                                               00024800                  248
                * A R E A   D E   M O D E L A G E M   D O   S I S T E M A       00024900                  249
                *                                                               00025000                  250
                *****************************************************************00025100                  251
                *                                                               00025200                  252
                *  G E R A D O R   D E   C H E G A D A S   D E   F E R I A S    00025300                  253
                *                                                               00025400                  254
1               GEN?    GENERATE  3137,FNSEXP,,7,,9PH,9PF                       00025500                  255
2                       ASSIGN    1,7,PH                                        00025600                  256
3                       ASSIGN    3,5,PH                                        00025700                  257
4                       PRIORITY  100                                           00025800                  258
5                       QUEUE     5                                             00025900                  259
6                       TEST E    X$7,0                                         00026000                  260
7                       TRANSFER  ,WAIT                                         00026100                  261
8               EXITO   TERMINATE                                               00026200                  262
                *                                                               00026300                  263
                *---------------------------------------------------------------00026400                  264
                *                                                               00026500                  265
                *  G E R A D O R   D E   O U T R A S   A T I V I D A D E S      00026600                  266
                *                                                               00026700                  267
9               GEN0    GENERATE  34,FNSEXP,,,,9PF,9PH GERA OUTRAS ATIVIDADES 00026800                  268
10                      ASSIGN    1,9,PH                                        00026900                  269
11                      ASSIGN    3,3,PH                                        00027000                  270
12                      PRIORITY  50                                            00027100                  271
13              RECA    QUEUE     3                                             00027200                  272
14                      TRANSFER  ,WAIT                                         00027300                  273
                *                                                               00027400                  274
                *---------------------------------------------------------------00027500                  275
                *                                                               00027600                  276
                *  G E R A D O R   D E   P R O G R A M A S   E M   M A O S   N A00027700                  277
                *                 S I T U A C A O   I N I C I A L               00027800                  278
                *                                                               00027900                  279
15              *GEN1   GENERATE  ,,,19,,9PH,9PF,9P0 .. GERA PROGRAMAS NOVOS   00028000                  280
16                      ASSIGN    1,FN$TPPRG,PH                                 00028100                  281
17                      ASSIGN    3,1,PH                                        00028200                  282
18                      QUEUE     1                                             00028300                  283
19                      TRANSFER  ,WAIT                                         00028400                  284
                *- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - 00028500                  285
20              GEN2    GENERATE  ,,,31,,9PF,9PH       GERA PROGRAMAS EM MANUTENCAO 00028600                  286
21                      ASSIGN    1,FN$TPMAN,PH                                 00028700                  287
```

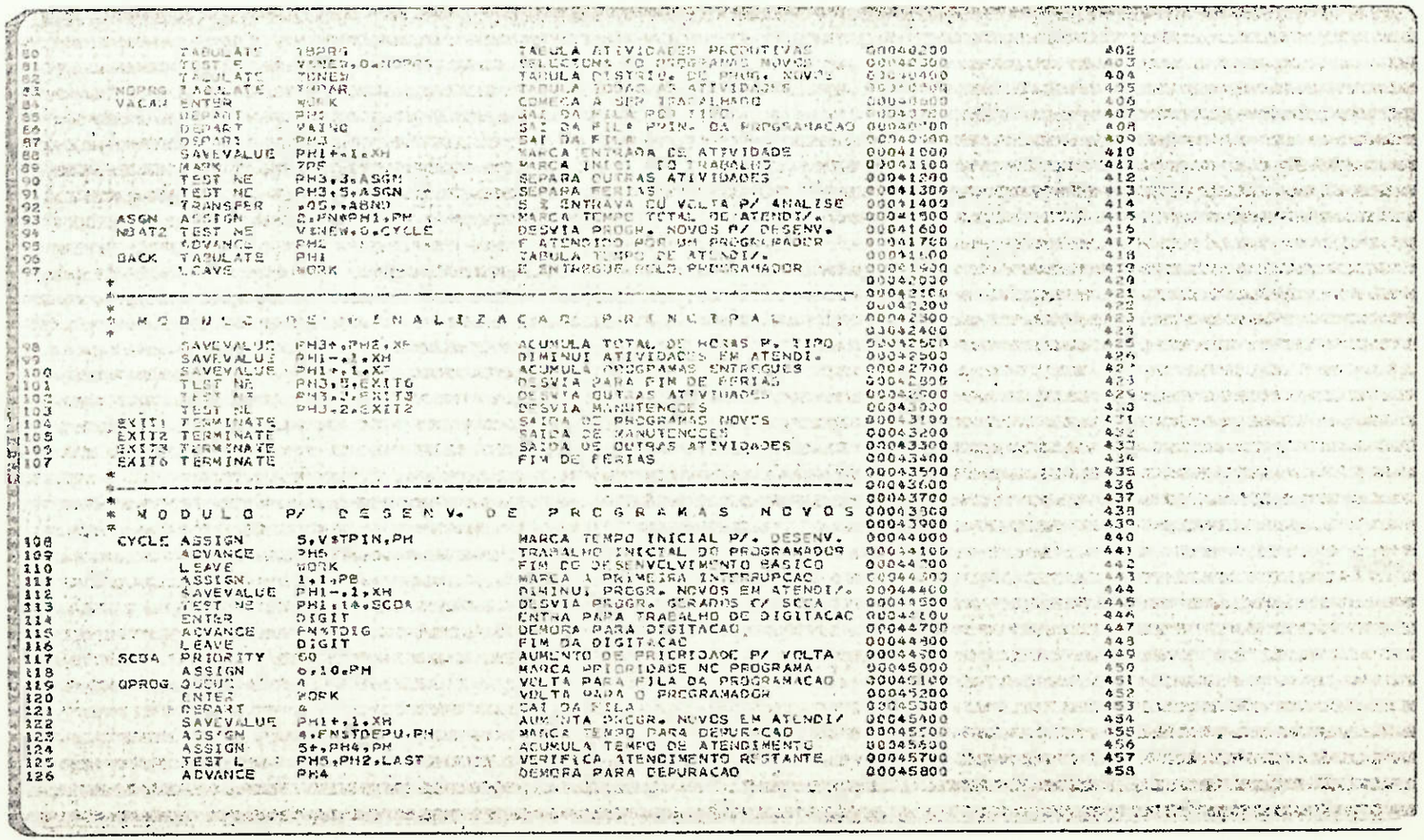


```
                         ASSIGN    3,2,PH
                         TRANSFER  ,SBR,,QUE2
                         PRIORITY  0
                QUE2     QUEUE     2
                         TRANSFER  ,WAIT
                *
                *-------------------------------------------------------
                *
                *  G E R A D O R   D E   P R O G R A M A S   T I P O  1 0
                *
                GEN10    GENERATE  641,FNSEXP,,,,9PH,9PF,9P0
                         ASSIGN    1,10,PH
                         ASSIGN    3,1,PH
                RCP10    QUEUE     1
                         TRANSFER  ,WAIT
                *
                *-------------------------------------------------------
                *
                *  G E R A D O R   D E   P R O G R A M A S   T I P O  1 2
                *
                GEN12    GENERATE  732,FNSEXP,,,,9PH,9PF,9P0
                         ASSIGN    1,12,PH
                         ASSIGN    3,1,PH
                RCP12    QUEUE     1
                         TRANSFER  ,WAIT
                *
                *-------------------------------------------------------
                *
                *  G E R A D O R   D E   P R O G R A M A S   T I P O  1 4
                *
                GEN14    GENERATE  823,FNSEXP,,,,9PH,9PF,9P0
                         ASSIGN    1,14,PH
                         ASSIGN    3,1,PH
                RCP14    QUEUE     1
                         TRANSFER  ,WAIT
                *
                *-------------------------------------------------------
                *
                *  G E R A D O R   D E   P R O G R A M A S   T I P O  1 6
                *
                GEN16    GENERATE  631,FNSEXP,,,,9PH,9PF,9P0
                         ASSIGN    1,16,PH
                         ASSIGN    3,1,PH
                RCP16    QUEUE     1
                         TRANSFER  ,WAIT
                *
                *-------------------------------------------------------
                *
                *  G E R A D O R   D E   P R O G R A M A S   T I P O  1 8
                *
                GEN18    GENERATE  2968,FNSEXP,,,,9PH,9PF,9P0
                         ASSIGN    1,18,PH
                         ASSIGN    3,1,PH
                RCP18    QUEUE     1
                         TRANSFER  ,WAIT
                *
                *-------------------------------------------------------
```

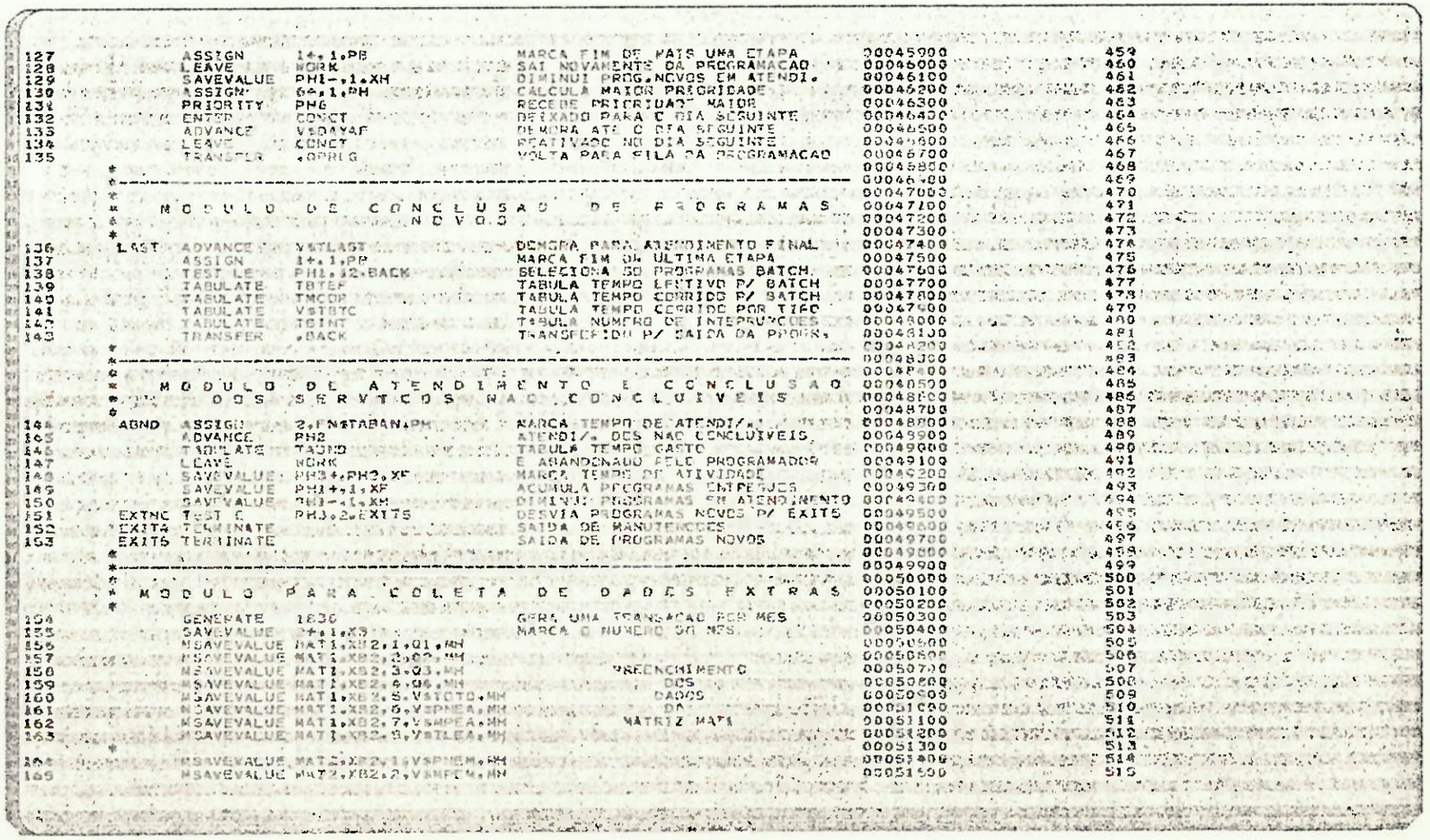


```
                *
                * G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 1
                *
 52     GEN11   GENERATE    197,FN$EXP,,,,9PF,9PH
 53             ASSIGN      1,11,PH
 54             ASSIGN      3,2,PH
 55             TRANSFER    ,QUEM
                *
                *-------------------------------------------------------------
                *
                *
                * G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 3
                *
 56     GEN13   GENERATE    155,FN$EXP,,,,9PF,9PH
 57             ASSIGN      1,13,PH
 58             ASSIGN      3,2,PH
 59             TRANSFER    ,QUEM
                *
                *-------------------------------------------------------------
                *
                * G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 5
                *
 60     GEN15   GENERATE    320,FN$EXP,,,,9PF,9PH
 61             ASSIGN      1,15,PH
 62             ASSIGN      3,2,PH
 63             TRANSFER    ,QUEM
                *
                *-------------------------------------------------------------
                *
                * G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 7
                *
 64     GEN17   GENERATE    82,FN$EXP,,,,9PF,9PH
 65             ASSIGN      1,17,PH
 66             ASSIGN      3,2,PH
 67             TRANSFER    ,QUEM
                *
                *-------------------------------------------------------------
                *
                * G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 9
                *
 68     GEN19   GENERATE    626,FN$EXP,,,,9PF,9PH
 69             ASSIGN      1,19,PH
 70             ASSIGN      3,2,PH
 71             TRANSFER    ,QUEM                   TRANSFERE P/ FILA DE MANUTENCOES
                *
 72     QUEM    TABULATE    TBMAT                   TABULA DISTR. DE MANUTENCOES
 73             TRANSFER    ,400,,RECM              DESVIA 40,0% PARA RECM
 74             PRIORITY    5                       ASSINALA PRIORIDADE
 75     RECM    QUEUE       2                       ENTRA NA FILA DE MANUTENCOES
                *
                *-------------------------------------------------------------
                *
                * M O D U L O   D E   A T E N D I M E N T O   P R I N C I P A L
                *
 76     WAIT    QUEUE       MAINT                   ENTRA NA FILA DA PROGRAMACAO
 77             QUEUE       PH1                     ENTRA NA FILA POR TIPO
 78             TEST NE     PH3,0,NOPRG             SEPARA OUTRAS ATIVIDADES
 79             TEST NE     PH3,0,VACAN             SEPARA FERIAS

                00034500                                    345
                00034600                                    346
                00034700                                    347
                00034800                                    348
                00034900                                    349
                00035000                                    350
                00035100                                    351
                00035200                                    352
                00035300                                    353
                00035400                                    354
                00035500                                    355
                00035600                                    356
                00035700                                    357
                00035800                                    358
                00035900                                    359
                00036000                                    360
                00036100                                    361
                00036200                                    362
                00036300                                    363
                00036400                                    364
                00036500                                    365
                00036600                                    366
                00036700                                    367
                00036800                                    368
                00036900                                    369
                00037000                                    370
                00037100                                    371
                00037200                                    372
                00037300                                    373
                00037400                                    374
                00037500                                    375
                00037600                                    376
                00037700                                    377
                00037800                                    378
                00037900                                    379
                00038000                                    380
                00038100                                    381
                00038200                                    382
                00038300                                    383
                00038400                                    384
                00038500                                    385
                00038600                                    386
                00038700                                    387
                00038800                                    388
                00038900                                    389
                00039000                                    390
                00039100                                    391
                00039200                                    392
                00039300                                    393
                00039400                                    394
                00039500                                    395
                00039600                                    396
                00039700                                    397
                00039800                                    398
                00039900                                    399
                00040000                                    400
                00040100                                    401
```

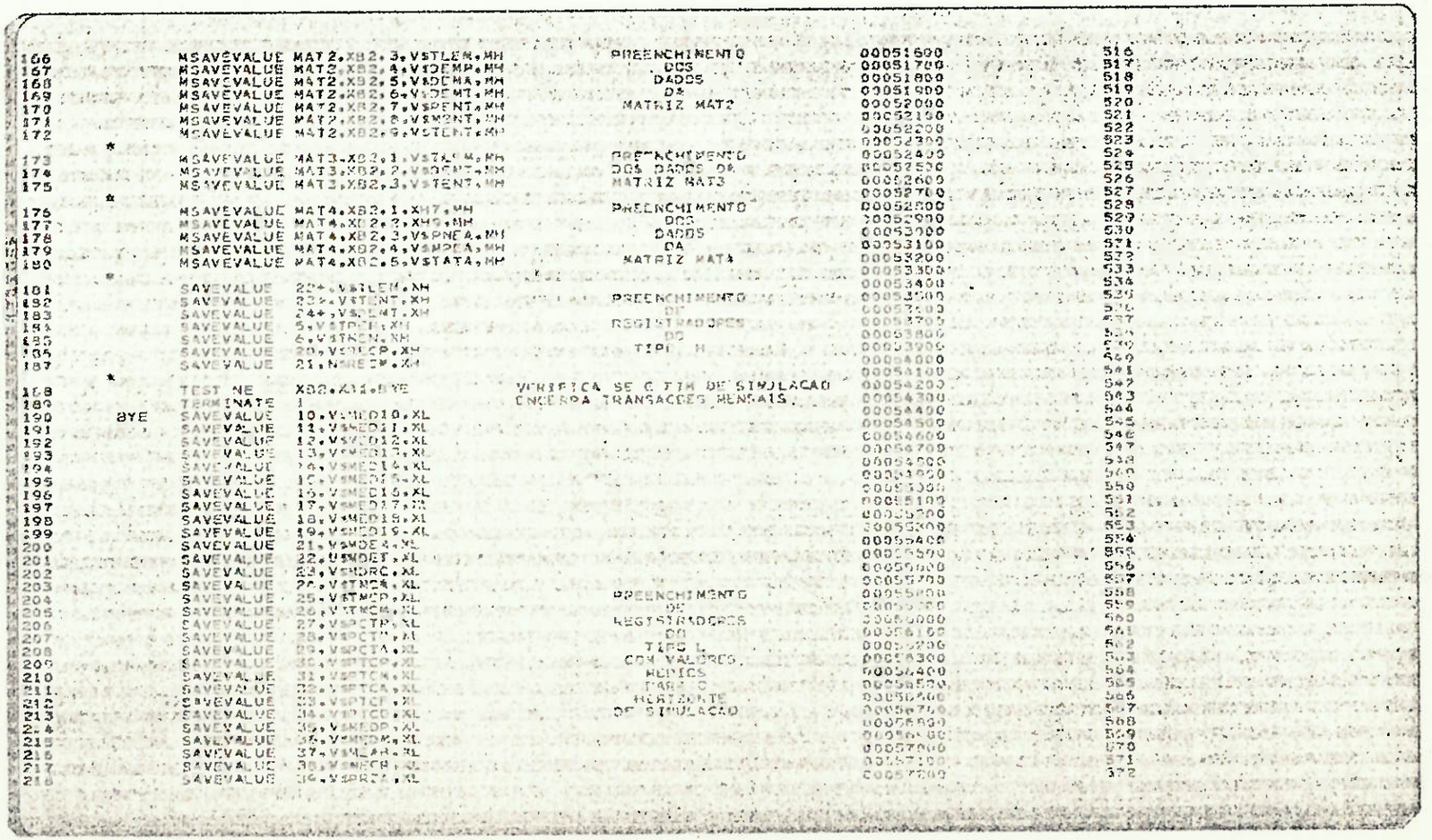


```
                TABULATE    TBPRG                TABELA ATIVIDADES PRODUTIVAS
                TEST E      VENEW,0,NOPRG        SELECIONA P/ PROGRAMAS NOVOS
                TABULATE    TDNEW                TABULA DISTRIB. DE PROG. NOVOS
                TABULATE    TWORK                TABULA TEMPO DE ATIVIDADES
NOPRG           ENTER       WORK                 COMECA A SER TRABALHADO
                DEPART      PHS                  SAI DA FILA P/ FIM
                DEPART      WAITQ                SAI DA FILA P/IN. DA PROGRAMACAO
                DEPART      PH3                  SAI DA FILA
                SAVEVALUE   PH1+,1,XH            MARCA ENTRADA DE ATIVIDADE
                MARK        TPS                  MARCA INICIO DO TRABALHO
                TEST NE     PH3,3,ASGN           SEPARA OUTRAS ATIVIDADES
                TEST NE     PH3,5,ASGN           SEPARA FERIAS
                TRANSFER    ,05,,ABND            S E ENTRAVA OU VOLTA P/ ANALISE
ASGN            ASSIGN      5,FN$PH1,PH          MARCA TEMPO TOTAL DE ATENDIM.
N3AT2           TEST NE     V$NEW,0,CYCLE        DESVIA PROGR. NOVOS P/ DESENV.
                ADVANCE     PH1                  E ATENDIDO POR UM PROGRAMADOR
BACK            TABULATE    PH1                  TABULA TEMPO DE ATENDIM.
                LEAVE       WORK                 E ENTREGUE PELO PROGRAMADOR
*
*-------------------------------------------------------------------------
*
* M O D U L O   D E   F I N A L I Z A C A O   P R I N C I P A L
*
                SAVEVALUE   PH3+,PH2,XF          ACUMULA TOTAL DE HORAS P. TIPO
                SAVEVALUE   PH1-,1,XH            DIMINUI ATIVIDADES EM ATENDI.
                SAVEVALUE   PH1+,1,XT            ACUMULA PROGRAMAS ENTREGUES
                TEST NE     PH3,5,EXIT0          DESVIA PARA FIM DE FERIAS
                TEST NE     PH3,3,EXIT3          DESVIA OUTRAS ATIVIDADES
                TEST NE     PH3,2,EXIT2          DESVIA MANUTENCOES
EXIT1           TERMINATE                        SAIDA DE PROGRAMAS NOVOS
EXIT2           TERMINATE                        SAIDA DE MANUTENCOES
EXIT3           TERMINATE                        SAIDA DE OUTRAS ATIVIDADES
EXIT0           TERMINATE                        FIM DE FERIAS
*
*-------------------------------------------------------------------------
*
* M O D U L O   P/   D E S E N V.   D E   P R O G R A M A S   N O V O S
*
CYCLE           ASSIGN      5,V$TPIN,PH          MARCA TEMPO INICIAL P/. DESENV.
                ADVANCE     PH5                  TRABALHO INICIAL DO PROGRAMADOR
                LEAVE       WORK                 FIM DO DESENVOLVIMENTO BASICO
                ASSIGN      1,1,PB               MARCA A PRIMEIRA INTERRUPCAO
                SAVEVALUE   PH1-,1,XH            DIMINUI PROGR. NOVOS EM ATENDIM.
                TEST NE     PH1,14,SCOA          DESVIA PROGR. GERADOS P/ SCOA
                ENTER       DIGIT                ENTRA PARA TRABALHO DE DIGITACAO
                ADVANCE     PN%TDIG              DEMORA PARA DIGITACAO
                LEAVE       DIGIT                FIM DA DIGITACAO
SCOA            PRIORITY    60                   AUMENTO DE PRIORIDADE P/ VOLTA
                ASSIGN      6,10,PH              MARCA PRIORIDADE NO PROGRAMA
QPROG           QUEUE       4                    VOLTA PARA FILA DA PROGRAMACAO
                ENTER       WORK                 VOLTA PARA O PROGRAMADOR
                DEPART      4                    SAI DA FILA
                SAVEVALUE   PH1+,1,XH            AUMENTA PROGR. NOVOS EM ATENDIM
                ASSIGN      4,FN$TDEPU,PH        MARCA TEMPO PARA DEPURACAO
                ASSIGN      5+,PH4,PH            ACUMULA TEMPO DE ATENDIMENTO
                TEST L      PH5,PH2,LAST         VERIFICA ATENDIMENTO RESTANTE
                ADVANCE     PH4                  DEMORA PARA DEPURACAO
```

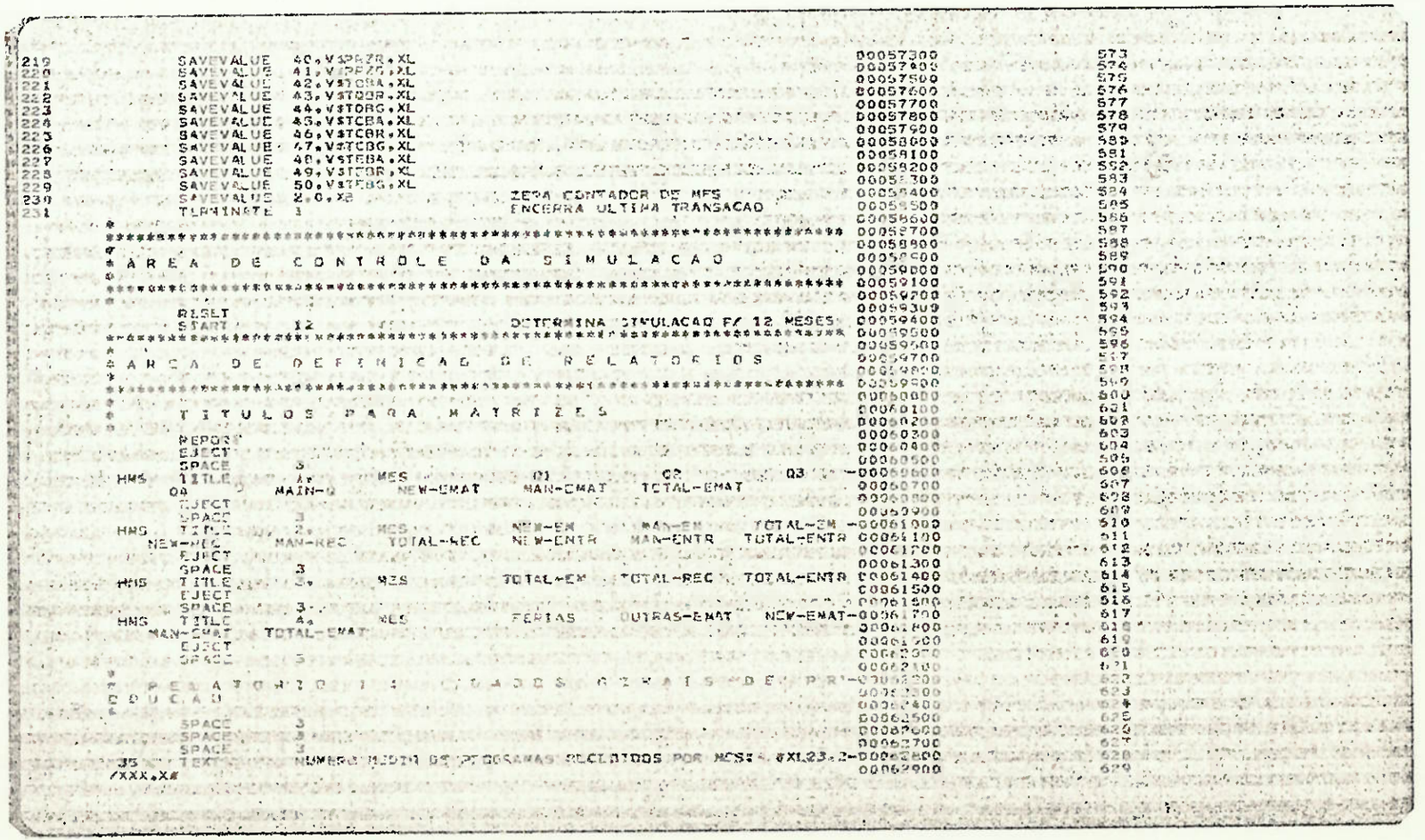


```
127                ASSIGN    1+,1,PR          MARCA FIM DE MAIS UMA ETAPA
128                LEAVE     WORK             SAI MOVIMENTO DA PROGRAMACAO
129                SAVEVALUE PH1-,1,XH        DIMINUI PROG.NOVOS EM ATENDI.
130                ASSIGN    6+,1,PH          CALCULA MAIOR PRIORIDADE
131                PRIORITY  PH6              RECEBE PRIORIDADE MAIOR
132                ENTER     CONCT            DEIXADO PARA O DIA SEGUINTE
133                ADVANCE   V$DAYAF          DEMORA ATE O DIA SEGUINTE
134                LEAVE     CONCT            REATIVADO NO DIA SEGUINTE
135                TRANSFER  ,OPRIG           VOLTA PARA FILA DA PROGRAMACAO
                   *
                   *-------------------------------------------------------
                   *
                   *
                   *    M O D U L O   D E   C O N C L U S A O   D E   P R O G R A M A S
                   *                    N O V O S
                   *
136                LAST      ADVANCE  V$TLAST          DEMORA PARA ATENDIMENTO FINAL
137                          ASSIGN   1+,1,PR          MARCA FIM DA ULTIMA ETAPA
138                          TEST LE  PH1,32,BACK      SELECIONA SO PROGRAMAS BATCH.
139                          TABULATE TBTEF            TABULA TEMPO LENTIVO P/ BATCH
140                          TABULATE TMCOP            TABULA TEMPO CORRIDO P/ BATCH
141                          TABULATE V$TBTC           TABULA TEMPO CORRIDO POR TIPO
142                          TABULATE TBINT            TABULA NUMERO DE INTERRUPCOES
143                          TRANSFER ,BACK            TRANSFERIDO P/ SAIDA DA PROGRA.
                   *
                   *-------------------------------------------------------
                   *
                   *
                   *    M O D U L O   D E   A T E N D I M E N T O   E   C O N C L U S A O
                   *       D O S   S E R V I C O S   N A O   C O N C L U I V E I S
                   *
144                ABND      ASSIGN   2,FN$TABAN,PH  MARCA TEMPO DE ATENDI/.
145                          ADVANCE  PH2              ATENDI/. DOS NAO CONCLUIVEIS
146                          TABULATE TABND            TABULA TEMPO GASTO
147                          LEAVE    WORK             E ABANDONADO PELO PROGRAMADOR
148                          SAVEVALUE PH3+,PH2,XF     MARCA TEMPO DE ATIVIDADE
149                          SAVEVALUE PH1+,1,XF       ACUMULA PROGRAMAS EMPREGUES
150                          SAVEVALUE PH1-,1,XH       DIMINUI PROGRAMAS EM ATENDIMENTO
151                EXTNC     TEST G   PH3,2,EXITS     DESVIA PROGRAMAS NOVOS P/ EXITS
152                EXITA     TERMINATE                 SAIDA DE MANUTENCOES
153                EXITS     TERMINATE                 SAIDA DE PROGRAMAS NOVOS
                   *
                   *-------------------------------------------------------
                   *
                   *
                   *    M O D U L O   P A R A   C O L E T A   D E   D A D O S   E X T R A S
                   *
154                          GENERATE 1830             GERA UMA TRANSACAO POR MES
155                          SAVEVALUE 3+,1,X3         MARCA O NUMERO DO MES
156                          MSAVEVALUE MAT1,XB2,1,Q1,MM
157                          MSAVEVALUE MAT1,XB2,2,Q2,MM
158                          MSAVEVALUE MAT1,XB2,3,Q3,MM
159                          MSAVEVALUE MAT1,XB2,4,Q4,MM
160                          MSAVEVALUE MAT1,XB2,5,V$TCTO,MM
161                          MSAVEVALUE MAT1,XB2,6,V$PMEA,MM
162                          MSAVEVALUE MAT1,XB2,7,V$MPEA,MM
163                          MSAVEVALUE MAT1,XB2,8,V$TLEA,MM
                   *
164                          MSAVEVALUE MAT2,XB2,1,V$PMEM,MM
165                          MSAVEVALUE MAT2,XB2,2,V$MPEM,MM
```

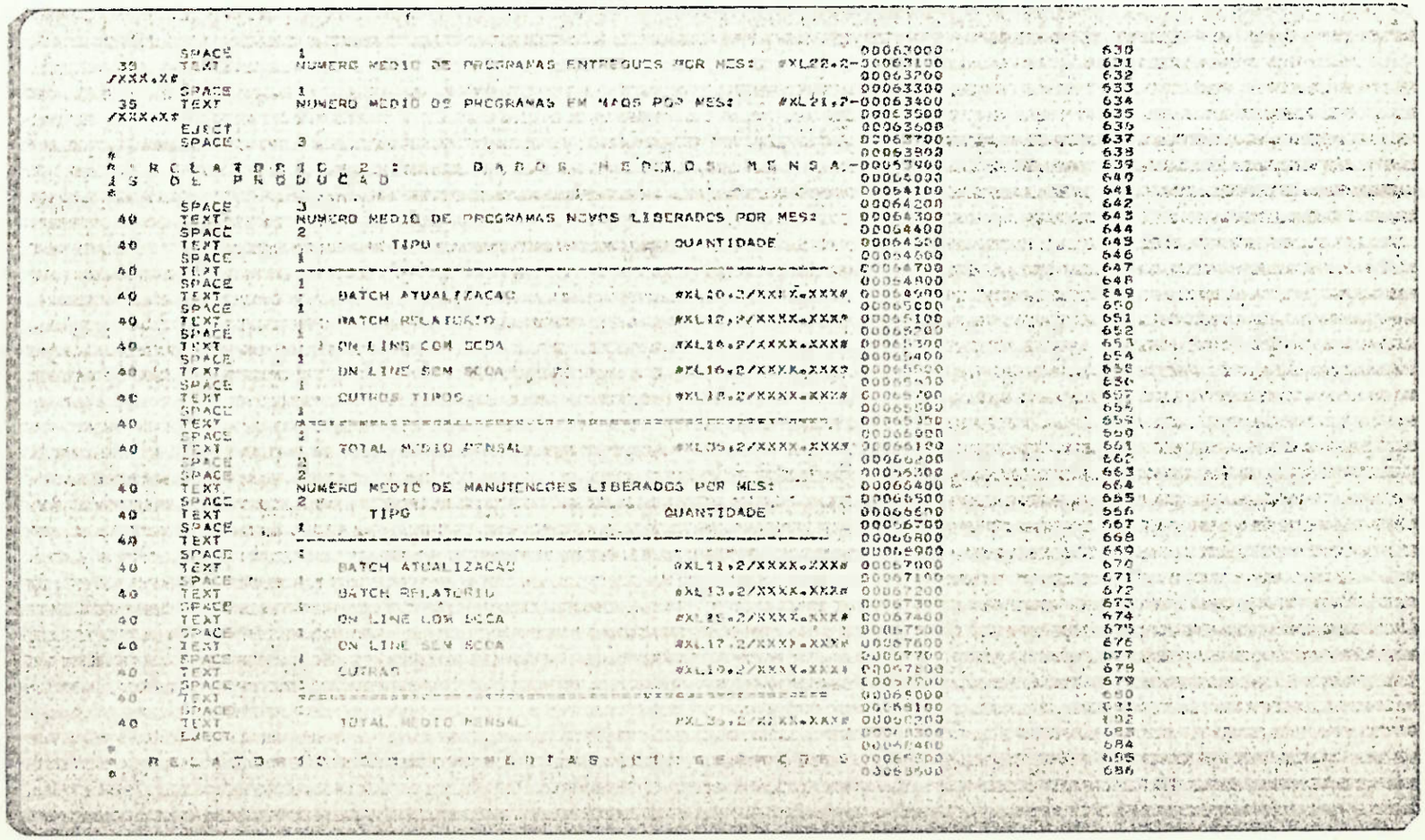
PREENCHIMENTO DOS DADOS DA MATRIZ MAT1



```
166          MSAVEVALUE MAT2,XB2,3,VSTLEM,MH      PREENCHIMENTO          00051600          516
167          MSAVEVALUE MAT2,XB2,4,VIDEMP,MH        DOS                  00051700          517
168          MSAVEVALUE MAT2,XB2,5,VADCMA,MH       DADOS                 00051800          518
169          MSAVEVALUE MAT2,XB2,6,VSDEMT,MH         DA                  00051900          519
170          MSAVEVALUE MAT2,XB2,7,VSPENT,MH    MATRIZ MAT2              00052000          520
171          MSAVEVALUE MAT2,XB2,8,VSM2NT,MH                             00052100          521
172          MSAVEVALUE MAT2,XB2,9,VSTENT,MH                             00052200          522
               *                                                         00052300          523
173          MSAVEVALUE MAT3,XB2,1,VSTLEM,MH      PREENCHIMENTO          00052400          524
174          MSAVEVALUE MAT3,XB2,2,VSDENT,MH        DOS DADOS DA         00052500          525
175          MSAVEVALUE MAT3,XB2,3,VSTENT,MH    MATRIZ MAT3              00052600          526
               *                                                         00052700          527
175          MSAVEVALUE MAT4,XB2,1,XH7,MH         PREENCHIMENTO          00052800          528
177          MSAVEVALUE MAT4,XB2,2,XH9,MH           DOS                  00052900          529
178          MSAVEVALUE MAT4,XB2,3,VSPNEA,MH       DADOS                 00053000          530
179          MSAVEVALUE MAT4,XB2,4,VSMDEA,MH         DA                  00053100          531
180          MSAVEVALUE MAT4,XB2,5,VSTATA,MH    MATRIZ MAT4              00053200          532
               *                                                         00053300          533
181          SAVEVALUE  22+,VSTLEM,XH                                    00053400          534
182          SAVEVALUE  23+,VSTENT,XH             PREENCHIMENTO          00053500          535
183          SAVEVALUE  24+,VSTEMT,XH               DE                   00053600          536
184          SAVEVALUE  5,VSTPCE,XH               REGISTRADORES          00053700          537
185          SAVEVALUE  6,VSTRCN,XH                 DO                   00053800          538
186          SAVEVALUE  20,VSDECP,XH              TIPO H                 00053900          539
187          SAVEVALUE  21,NRE2M,XH                                      00054000          540
               *                                                         00054100          541
188          TEST NE    X$$,X11,BYE                                      00054200          542
189          TERMINATE  1                    VERIFICA SE O TIM DE SIMULACAO 00054300       543
190     BYE  SAVEVALUE  10,VSMED10,XL        ENCERRA TRANSACOES MENSAIS.  00054400          544
191          SAVEVALUE  11,VSMED11,XL                                    00054500          545
192          SAVEVALUE  12,VSMED12,XL                                    00054600          546
193          SAVEVALUE  13,VSMED13,XL                                    00054700          547
194          SAVEVALUE  14,VSMED14,XL                                    00054800          548
195          SAVEVALUE  1C,VSMED15,XL                                    00054900          549
196          SAVEVALUE  16,VSMED16,XL                                    00055000          550
197          SAVEVALUE  17,VSMED17,XL                                    00055100          551
198          SAVEVALUE  18,VSMED18,XL                                    00055200          552
199          SAVEVALUE  19,VSMED19,XL                                    00055300          553
200          SAVEVALUE  21,VSMDE4,XL                                     00055400          554
201          SAVEVALUE  22,VSMDET,XL                                     00055500          555
202          SAVEVALUE  23,VSMDRC,XL                                     00055600          556
203          SAVEVALUE  24,VSTMC4,XL                                     00055700          557
204          SAVEVALUE  25,VSTMCD,XL              PREENCHIMENTO          00055800          558
205          SAVEVALUE  26,VSTMCS,XL                DE                   00055900          559
206          SAVEVALUE  27,VSPCTP,XL              REGISTRADORES          00056000          560
207          SAVEVALUE  28,VSPCTT,XL                DO                   00056100          561
208          SAVEVALUE  29,VSPCTA,XL              TIPO L                 00056200          562
209          SAVEVALUE  30,VSPTCD,XL            COM VALORES              00056300          563
210          SAVEVALUE  31,VSPTCM,XL              MEDIOS                 00056400          564
211          SAVEVALUE  32,VSPTCA,XL              PARA O                 00056500          565
212          SAVEVALUE  33,VSPTCT,XL             RELATORIO               00056600          566
213          SAVEVALUE  34,VSPTCC,XL           DE SIMULACAO              00056700          567
214          SAVEVALUE  35,VSMEDM,XL                                     00056800          568
215          SAVEVALUE  36,VSMEDM,XL                                     00056900          569
216          SAVEVALUE  37,VSMLAR,XL                                     00057000          570
217          SAVEVALUE  38,VSMECR,XL                                     00057100          571
218          SAVEVALUE  39,VSPRIA,XL                                     00057200          572
```

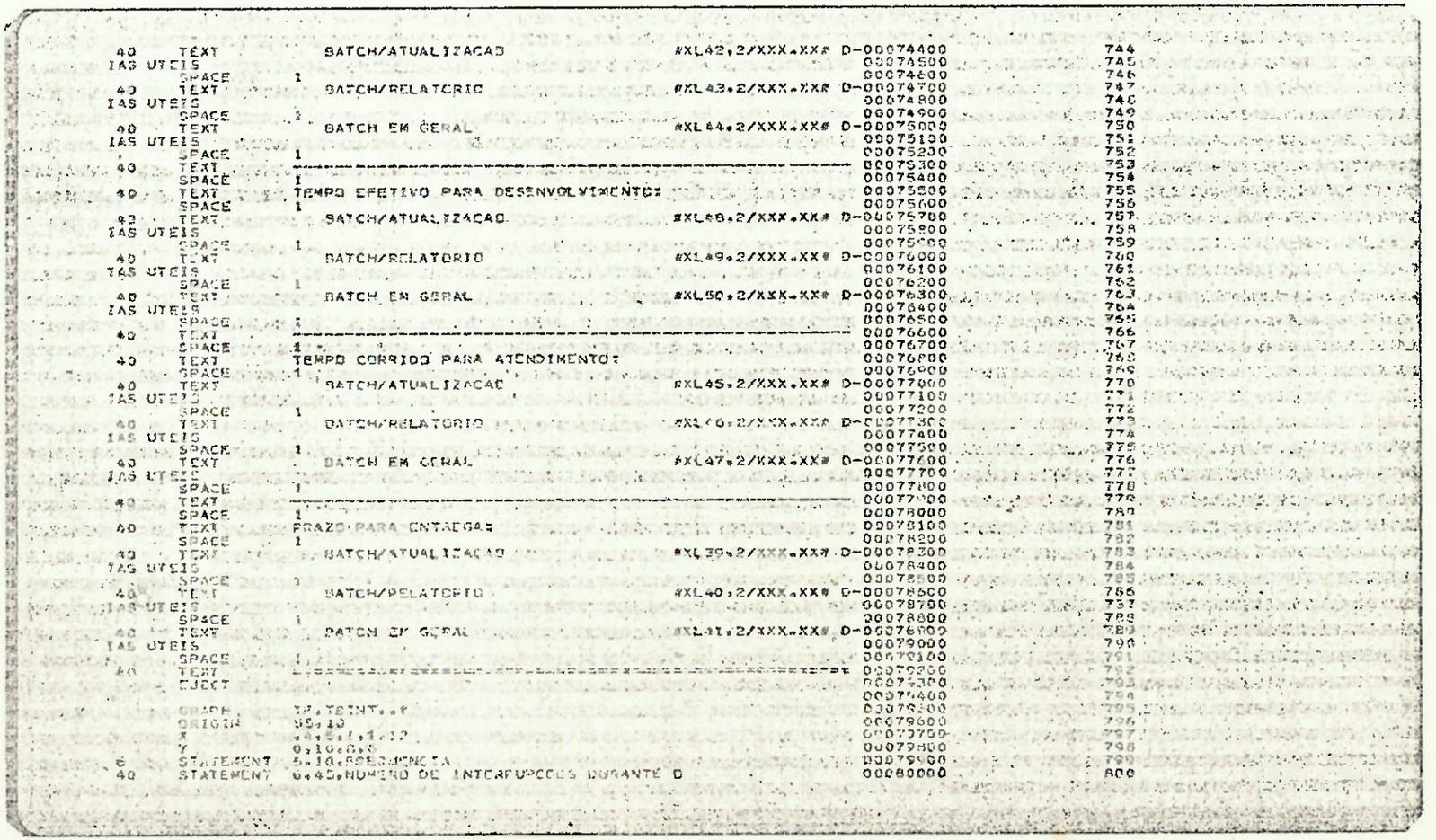


```
219          SAVEVALUE  40,V$PRZR,XL                                                00057300                573
220          SAVEVALUE  41,V$PRZG,XL                                                00057400                574
221          SAVEVALUE  42,V$TCBA,XL                                                00057500                575
222          SAVEVALUE  43,V$TCBR,XL                                                00057600                576
223          SAVEVALUE  44,V$TCBG,XL                                                00057700                577
224          SAVEVALUE  45,V$TCBA,XL                                                00057800                578
225          SAVEVALUE  46,V$TCBR,XL                                                00057900                579
226          SAVEVALUE  47,V$TCBG,XL                                                00058000                580
227          SAVEVALUE  48,V$TEBA,XL                                                00058100                581
228          SAVEVALUE  49,V$TEBR,XL                                                00058200                582
229          SAVEVALUE  50,V$TEBG,XL                                                00058300                583
230          SAVEVALUE  2,0,X2          ZERA CONTADOR DE MES                        00058400                584
231          TERMINATE  1               ENCERRA ULTIMA TRANSACAO                    00058500                585
*                                                                                   00058600                586
*********************************************************************************  00058700                587
*                                                                                   00058800                588
* A R E A   D E   C O N T R O L E   D A   S I M U L A C A O                         00058900                589
*                                                                                   00059000                590
*********************************************************************************  00059100                591
*                                                                                   00059200                592
          RESET                                                                    00059300                593
          START      12              DETERMINA SIMULACAO P/ 12 MESES                00059400                594
*********************************************************************************  00059500                595
*                                                                                   00059600                596
* A R E A   D E   D E F I N I C A O   D E   R E L A T O R I O S                     00059700                597
*                                                                                   00059800                598
*********************************************************************************  00059900                599
*                                                                                   00060000                600
*     T I T U L O S   P A R A   M A T R I Z E S                                     00060100                601
*                                                                                   00060200                602
          REPORT                                                                   00060300                603
          EJECT                                                                    00060400                604
          SPACE      3                                                             00060500                605
HMS       TITLE      4,*     MES          Q1          Q2          Q3               00060600                606
     Q4             MAIN-Q        NEW-EMAT    MAN-EMAT    TOTAL-EMAT              00060700                607
          EJECT                                                                    00060800                608
          SPACE      3                                                             00060900                609
HMS       TITLE      3,*     MES          NEW-EM      MAN-EM      TOTAL-EM         00061000                610
     NEW-REC        MAN-REC     TOTAL-REC   NEW-ENTR    MAN-ENTR    TOTAL-ENTR   00061100                611
          EJECT                                                                    00061200                612
          SPACE      3                                                             00061300                613
HMS       TITLE      3,*     MES          TOTAL-EM    TOTAL-REC   TOTAL-ENTR       00061400                614
          EJECT                                                                    00061500                615
          SPACE      3                                                             00061600                616
HMS       TITLE      4,*     MES          FERIAS      OUTRAS-EMAT NEW-EMAT         00061700                617
     MAN-EMAT       TOTAL-EMAT                                                    00061800                618
          EJECT                                                                    00061900                619
          SPACE      3                                                             00062000                620
*                                                                                   00062100                621
*     R E L A T O R I O   D E   C U S T O S   A N U A I S   D E   P R O D U C A O   00062200                622
                                                                                   00062300                623
                                                                                   00062400                624
          SPACE      3                                                             00062500                625
          SPACE      3                                                             00062600                626
          SPACE      3                                                             00062700                627
 35       TEXT       NUMERO MEDIO DE PROGRAMAS SELECIONADOS POR MES= ,FXL23.2-    00062800                628
/XXX.X#                                                                            00062900                629
```

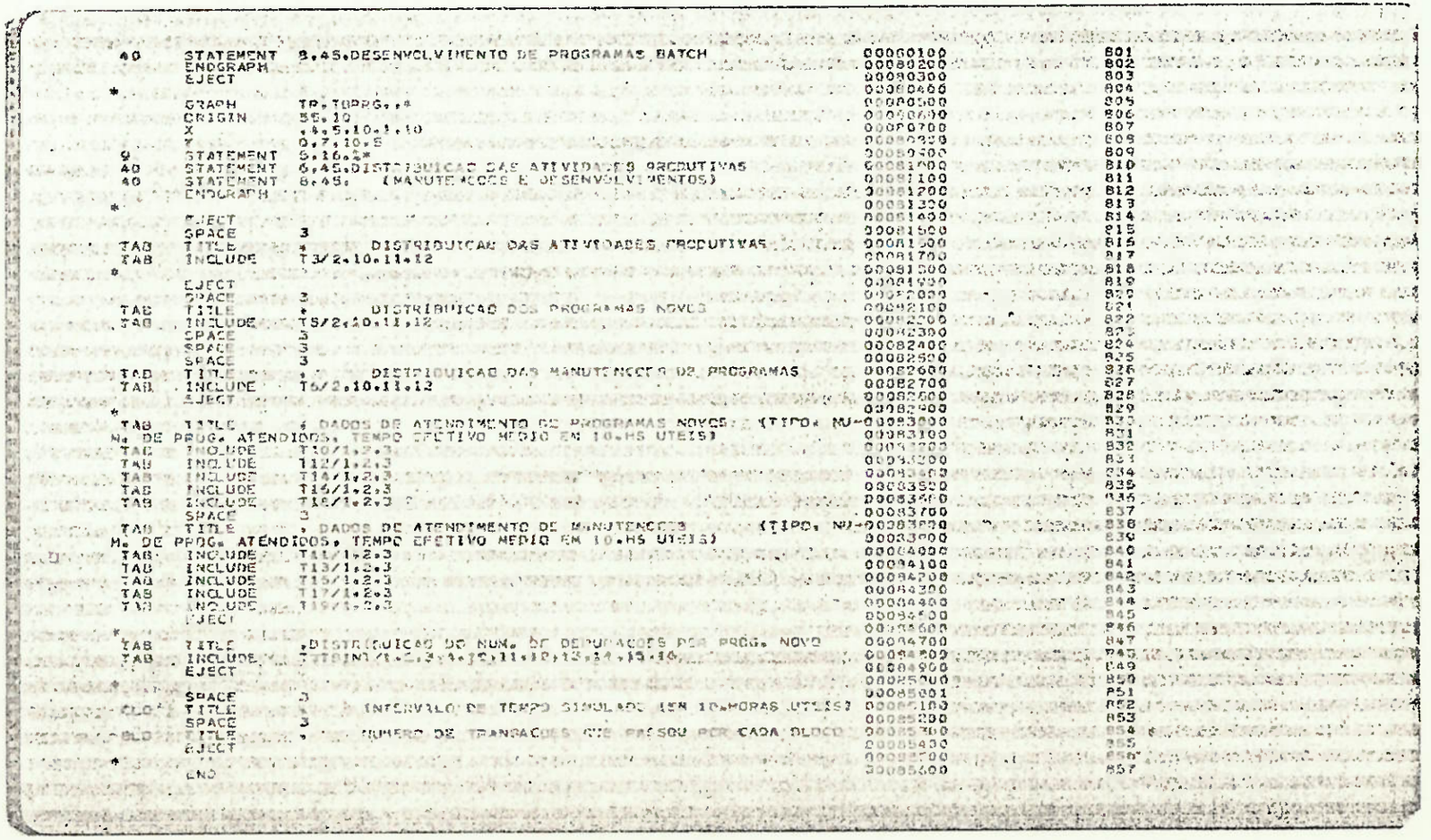


```
      SPACE   1                                                          00062000                630
30    TEXT    NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES:  #XL22,2-     00063100                631
/XXX,X#                                                                  00063200                632
      SPACE   1                                                          00063300                633
35    TEXT    NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES:    #XL21,2-     00063400                634
/XXX,X#                                                                  00063500                635
      EJECT                                                              00063600                636
      SPACE   3                                                          00063700                637
*                                                                        00063800                638
*  RELATORIO 2 :      DADOS MEDIDOS MENSAL-                            00063900                639
*  IS DE PRODUCAO                                                       00064000                640
*                                                                       00064100                641
      SPACE   3                                                          00064200                642
40    TEXT    NUMERO MEDIO DE PROGRAMAS NOVOS LIBERADOS POR MES:         00064300                643
      SPACE   2                                                          00064400                644
40    TEXT      TIPO                        QUANTIDADE                   00064500                645
      SPACE   1                                                          00064600                646
40    TEXT    ---------------------------------------------------        00064700                647
      SPACE   1                                                          00064800                648
40    TEXT      BATCH ATUALIZACAO           #XL10,2/XXXX,XXX#            00064900                649
      SPACE   1                                                          00065000                650
40    TEXT      BATCH RELATORIO             #XL12,2/XXXX,XXX#            00065100                651
      SPACE   1                                                          00065200                652
40    TEXT      ON LINE COM BCDA            #XL14,2/XXXX,XXX#            00065300                653
      SPACE   1                                                          00065400                654
40    TEXT      ON LINE SEM BCDA            #XL16,2/XXXX,XXX#            00065500                655
      SPACE   1                                                          00065600                656
40    TEXT      OUTROS TIPOS                #XL18,2/XXXX,XXX#            00065700                657
      SPACE   1                                                          00065800                658
40    TEXT    ---------------------------------------------------        00065900                659
      SPACE   1                                                          00066000                660
40    TEXT      TOTAL MEDIO MENSAL          #XL35,2/XXXX,XXX#            00066100                661
      SPACE   1                                                          00066200                662
      SPACE   2                                                          00066300                663
40    TEXT    NUMERO MEDIO DE MANUTENCOES LIBERADOS POR MES:             00066400                664
      SPACE   2                                                          00066500                665
40    TEXT      TIPO                        QUANTIDADE                   00066600                666
      SPACE   1                                                          00066700                667
40    TEXT    ---------------------------------------------------        00066800                668
      SPACE   1                                                          00066900                669
40    TEXT      BATCH ATUALIZACAO           #XL11,2/XXXX,XXX#            00067000                670
      SPACE   1                                                          00067100                671
40    TEXT      BATCH RELATORIO             #XL13,2/XXXX,XXX#            00067200                672
      SPACE   1                                                          00067300                673
40    TEXT      ON LINE COM BCDA            #XL15,2/XXXX,XXX#            00067400                674
      SPACE   1                                                          00067500                675
40    TEXT      ON LINE SEM BCDA            #XL17,2/XXXX,XXX#            00067600                676
      SPACE   1                                                          00067700                677
40    TEXT      OUTROS                      #XL19,2/XXXX,XXX#            00067800                678
      SPACE   1                                                          00067900                679
40    TEXT    ---------------------------------------------------        00068000                680
      SPACE   1                                                          00068100                681
40    TEXT      TOTAL MEDIO MENSAL          #XL35,2/XXXX,XXX#            00068200                682
      EJECT                                                              00068300                683
*                                                                        00068400                684
*  RELATORIO 3 : ESTATISTICAS DIARIAS DE PRODUCAO                        00068500                685
*                                                                        00068600                686
```

```
                SPACE   3                                                       00068700                687
 25             TEXT    TEMPO MEDIO ENTRE CHEGADAS DE OUTRAS ATIVIDADES:      -00068800               688
                #XL24.2/XX.X# HORAS                                             00068900                689
                SPACE   1                                                       00069000                690
 25             TEXT    TEMPO MEDIO ENTRE CHEGADAS DE PROGRAMAS NOVOS:        -00069100               691
                #XL25.2/XX.X# HORAS                                             00069200                692
                SPACE   1                                                       00069300                693
 25             TEXT    TEMPO MEDIO ENTRE CHEGADAS DE PROGRAMAS PARA MANUTENC-00069400               694
AO:             #XL26.2/XX.X# HORAS                                             00069500                695
                EJECT                                                           00069600                696
*                                                                               00069700                697
*  RELATORIO 4 : APROVEITAMENTO DO TEM-00069800                                 00069900                698
 PO EFETIVAMENTE TRABALHADO                                                     00070000                699
*                                                                               00070100                700
                SPACE   3                                                       00070200                701
 25             TEXT    PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE-00070300               702
 PROGRAMAS NOVOS:  #XL27.2/XX.X# %                                              00070400                703
                SPACE   1                                                       00070500                704
 25             TEXT    PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROG-00070600               705
RAMAS:          #XL28.2/XX.X# %                                                 00070700                706
                SPACE   1                                                       00070800                707
 25             TEXT    PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:-00070900               708
                #XL29.2/XX.X# %                                                 00071000                709
                SPACE   3                                                       00071100                710
                SPACE   3                                                       00071200                711
*                                                                               00071300                712
*  RELATORIO 5:  APROVEITAMENTO DO TE-00071400                                  00071500                713
*PO CORRIDO                                                                     00071600                714
*                                                                               00071700                715
                SPACE   3                                                       00071800                716
 25             TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO-00071900               717
                #XL30.2/XX.X# %                                                 00072000                718
                SPACE   1                                                       00072100                719
 25             TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE P-00072200               720
ROGRAMAS:       #XL31.2/XX.X# %                                                 00072300                721
                SPACE   1                                                       00072400                722
 25             TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDAD-00072500               723
ES:             #XL32.2/XX.X# %                                                 00072600                724
                SPACE   1                                                       00072700                725
 25             TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:        -00072800               726
                #XL33.2/XX.X# %                                                 00072900                727
                SPACE   1                                                       00073000                728
 25             TEXT    PORCENTAGEM DO TEMPO CORRIDO OCIOSO:                  -00073100               729
                #XL34.2/XX.X# %                                                 00073200                730
                EJECT                                                           00073300                731
                SPACE   3                                                       00073400                732
*                                                                               00073500                733
*  RELATORIO 6 :  TEMPOS MEDIOS PARA -00073600                                  00073700                734
 DESENVOLVIMENTO                                                                00073800                735
*                                                                               00073900                736
                SPACE   3                                                       00074000                737
 40             TEXT    TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH  00074100                738
                SPACE   2                                                       00074200                739
 40             TEXT    ========================================               00074300                740
                SPACE   1                                                       00074400                741
 40             TEXT    TEMPO EM FILA:                                         00074500                742
                SPACE   1                                                       00074600                743
```

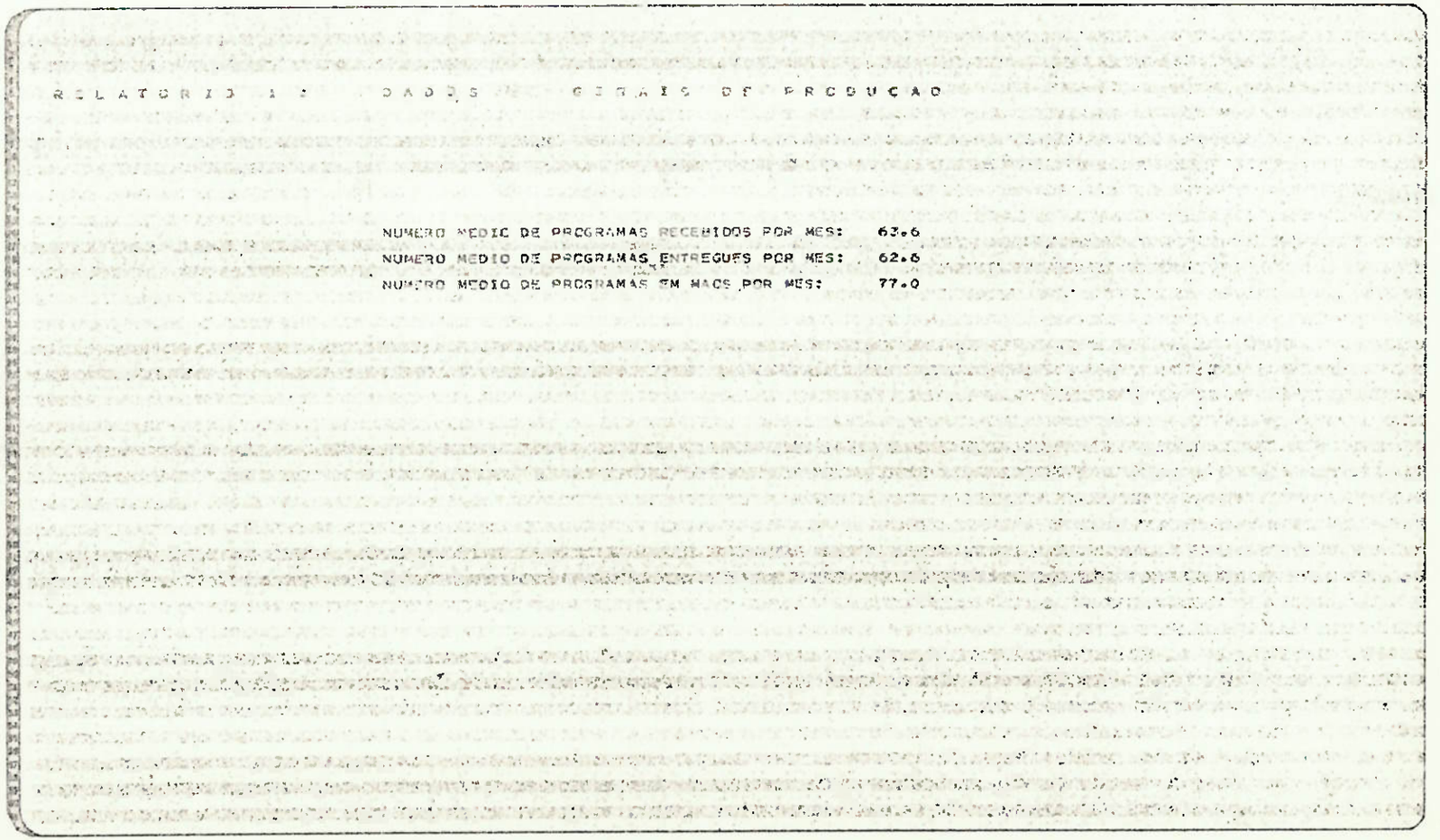


```
 40    TEXT      BATCH/ATUALIZACAO                #XL42,2/XXX,XX# D-00074400                744
IAS UTEIS                                                 00074500                745
       SPACE  1                                           00074600                746
 40    TEXT      BATCH/RELATORIO                  #XL43,2/XXX,XX# D-00074700                747
IAS UTEIS                                                 00074800                748
       SPACE  1                                           00074900                749
 40    TEXT      BATCH EM GERAL                   #XL44,2/XXX,XX# D-00075000                750
IAS UTEIS                                                 00075100                751
       SPACE  1                                           00075200                752
 40    TEXT      ------------------------------------------------- 00075300                753
       SPACE  1                                           00075400                754
 40    TEXT      TEMPO EFETIVO PARA DESENVOLVIMENTO:                00075500                755
       SPACE  1                                           00075600                756
 40    TEXT      BATCH/ATUALIZACAO                #XL48,2/XXX,XX# D-00075700                757
IAS UTEIS                                                 00075800                758
       SPACE  1                                           00075900                759
 40    TEXT      BATCH/RELATORIO                  #XL49,2/XXX,XX# D-00076000                760
IAS UTEIS                                                 00076100                761
       SPACE  1                                           00076200                762
 40    TEXT      BATCH EM GERAL                   #XL50,2/XXX,XX# D-00076300                763
IAS UTEIS                                                 00076400                764
       SPACE  1                                           00076500                765
 40    TEXT      ------------------------------------------------- 00076600                766
       SPACE  1                                           00076700                767
 40    TEXT      TEMPO CORRIDO PARA ATENDIMENTO:                    00076800                768
       SPACE  1                                           00076900                769
 40    TEXT      BATCH/ATUALIZACAO                #XL45,2/XXX,XX# D-00077000                770
IAS UTEIS                                                 00077100                771
       SPACE  1                                           00077200                772
 40    TEXT      BATCH/RELATORIO                  #XL46,2/XXX,XX# D-00077300                773
IAS UTEIS                                                 00077400                774
       SPACE  1                                           00077500                775
 40    TEXT      BATCH EM GERAL                   #XL47,2/XXX,XX# D-00077600                776
IAS UTEIS                                                 00077700                777
       SPACE  1                                           00077800                778
 40    TEXT      ------------------------------------------------- 00077900                779
       SPACE  1                                           00078000                780
 40    TEXT      PRAZO PARA ENTREGA:                                00078100                781
       SPACE  1                                           00078200                782
 40    TEXT      BATCH/ATUALIZACAO                #XL39,2/XXX,XX# D-00078300                783
IAS UTEIS                                                 00078400                784
       SPACE  1                                           00078500                785
 40    TEXT      BATCH/RELATORIO                  #XL40,2/XXX,XX# D-00078600                786
IAS UTEIS                                                 00078700                787
       SPACE  1                                           00078800                788
 40    TEXT      BATCH EM GERAL                   #XL41,2/XXX,XX# D-00078900                789
IAS UTEIS                                                 00079000                790
       SPACE  1                                           00079100                791
 40    TEXT      ------------------------------------------------- 00079200                792
       EJECT                                              00079300                793
                                                          00079400                794
       GBLA   &F,&INT,&I                                  00079500                795
       ORIGIN 95,10                                       00079600                796
       X      &I,&I,1,12                                  00079700                797
       Y      0,10,0,0                                    00079800                798
 6     STATEMENT  5,10,&FREQUENCIA                        00079900                799
 40    STATEMENT  6,40,NUMERO DE INTERRUPCOES DURANTE O   00080000                800
```



```
 40     STATEMENT  8,45,DESENVOLVIMENTO DE PROGRAMAS BATCH                  00080100
        ENDGRAPH                                                           00080200
        EJECT                                                              00080300
*                                                                          00080400
        GRAPH      TP,TBPRG,,*                                             00080500
        ORIGIN     55,10                                                   00080600
        X          ,4,5,10,1,10                                            00080700
        Y          0,7,10,5                                                00080800
 9      STATEMENT  5,16,5*                                                 00080900
 40     STATEMENT  6,45,DISTRIBUICAO DAS ATIVIDADES PRODUTIVAS             00081000
 40     STATEMENT  8,45,   (MANUTENCOES E DESENVOLVIMENTOS)                00081100
        ENDGRAPH                                                           00081200
*                                                                          00081300
        EJECT                                                              00081400
        SPACE      3                                                       00081500
 TAB    TITLE     *          DISTRIBUICAO DAS ATIVIDADES PRODUTIVAS       00081600
 TAB    INCLUDE   T3/2,10,11,12                                            00081700
*                                                                          00081800
        EJECT                                                              00081900
        SPACE      3                                                       00082000
 TAB    TITLE     *          DISTRIBUICAO DOS PROGRAMAS NOVOS             00082100
 TAB    INCLUDE   T5/2,10,11,12                                            00082200
        SPACE      3                                                       00082300
        SPACE      3                                                       00082400
        SPACE      3                                                       00082500
 TAB    TITLE     *          DISTRIBUICAO DAS MANUTENCOES DE PROGRAMAS    00082600
 TAB    INCLUDE   T6/2,10,11,12                                            00082700
        EJECT                                                              00082800
*                                                                          00082900
 TAB    TITLE     ,DADOS DE ATENDIMENTO DE PROGRAMAS NOVOS: (TIPO, NU-     00083000
Nº DE PROG. ATENDIDOS, TEMPO EFETIVO MEDIO EM 10.HS UTEIS)                 00083100
 TAB    INCLUDE   T10/1,2,3                                                00083200
 TAB    INCLUDE   T12/1,2,3                                                00083300
 TAB    INCLUDE   T14/1,2,3                                                00083400
 TAB    INCLUDE   T15/1,2,3                                                00083500
 TAB    INCLUDE   T16/1,2,3                                                00083600
        SPACE      3                                                       00083700
 TAB    TITLE     ,DADOS DE ATENDIMENTO DE MANUTENCOES          (TIPO, NU-00083800
Nº DE PROG. ATENDIDOS, TEMPO EFETIVO MEDIO EM 10.HS UTEIS)                 00083900
 TAB    INCLUDE   T11/1,2,3                                                00084000
 TAB    INCLUDE   T13/1,2,3                                                00084100
 TAB    INCLUDE   T15/1,2,3                                                00084200
 TAB    INCLUDE   T17/1,2,3                                                00084300
 TAB    INCLUDE   T19/1,2,3                                                00084400
        EJECT                                                              00084500
*                                                                          00084600
 TAB    TITLE     ,DISTRIBUICAO DO NUM. DE DEPURACOES POR PROG. NOVO       00084700
 TAB    INCLUDE   T7(IN)/1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16           00084800
        EJECT                                                              00084900
*                                                                          00085000
        SPACE      3                                                       00085001
 CLO    TITLE     *          INTERVALO DE TEMPO SIMULADO (EM 10.HORAS UTEIS)00085100
        SPACE      3                                                       00085200
 CLO    TITLE     *          NUMERO DE TRANSACOES QUE PASSOU POR CADA BLOCO00085300
        EJECT                                                              00085400
*                                                                          00085500
        END                                                                00085600
```

ANEXO III - PRINCIPAIS RELATÓRIOS DAS SIMULAÇÕES PROCESSADAS PARA CALIBRAÇÃO DO MODELO



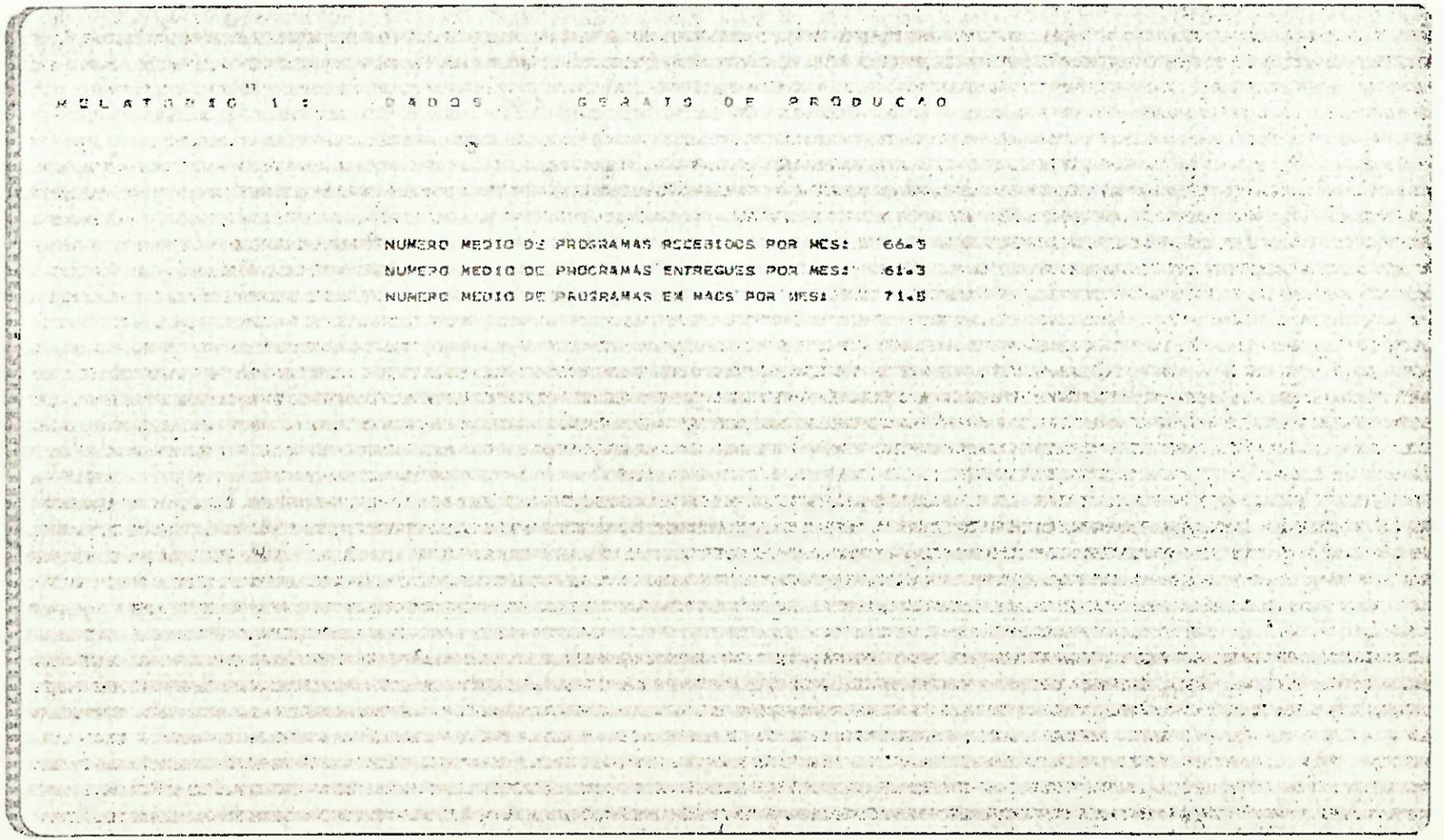
RELATORIO 1 : DADOS GERAIS DE PRODUÇÃO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 63.6
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 62.6
NUMERO MEDIO DE PROGRAMAS EM MACS POR MES: 77.0

ANEXO III.A - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DA SITUAÇÃO ATUAL PARA P = 33,3%

MOS HALFWORD MATRIX MATZ	NEW-EM	MAN-EM	TOTAL-EM	NEW-REC	MAN-REC	TOTAL-REC	NEW-ENTR	MAN-ENTR	TOTAL-ENTR
ROW/COLUMN	1	2	3	4	5	6	7	8	9
1	22	51	73	15	57	72	12	37	49
2	23	50	73	11	51	62	10	43	53
3	19	51	70	17	48	65	13	53	66
4	21	65	86	13	55	67	10	41	51
5	25	52	77	15	47	62	12	60	72
6	31	46	77	18	56	74	12	62	74
7	32	44	76	18	43	61	17	45	62
8	26	64	90	8	58	66	12	38	50
9	31	51	82	9	39	48	16	42	58
10	20	55	75	11	54	65	12	60	72
11	19	54	73	16	58	74	15	60	76
12	18	44	62	12	46	58	13	56	69

CONTINUAÇÃO DO ANEXO III.A

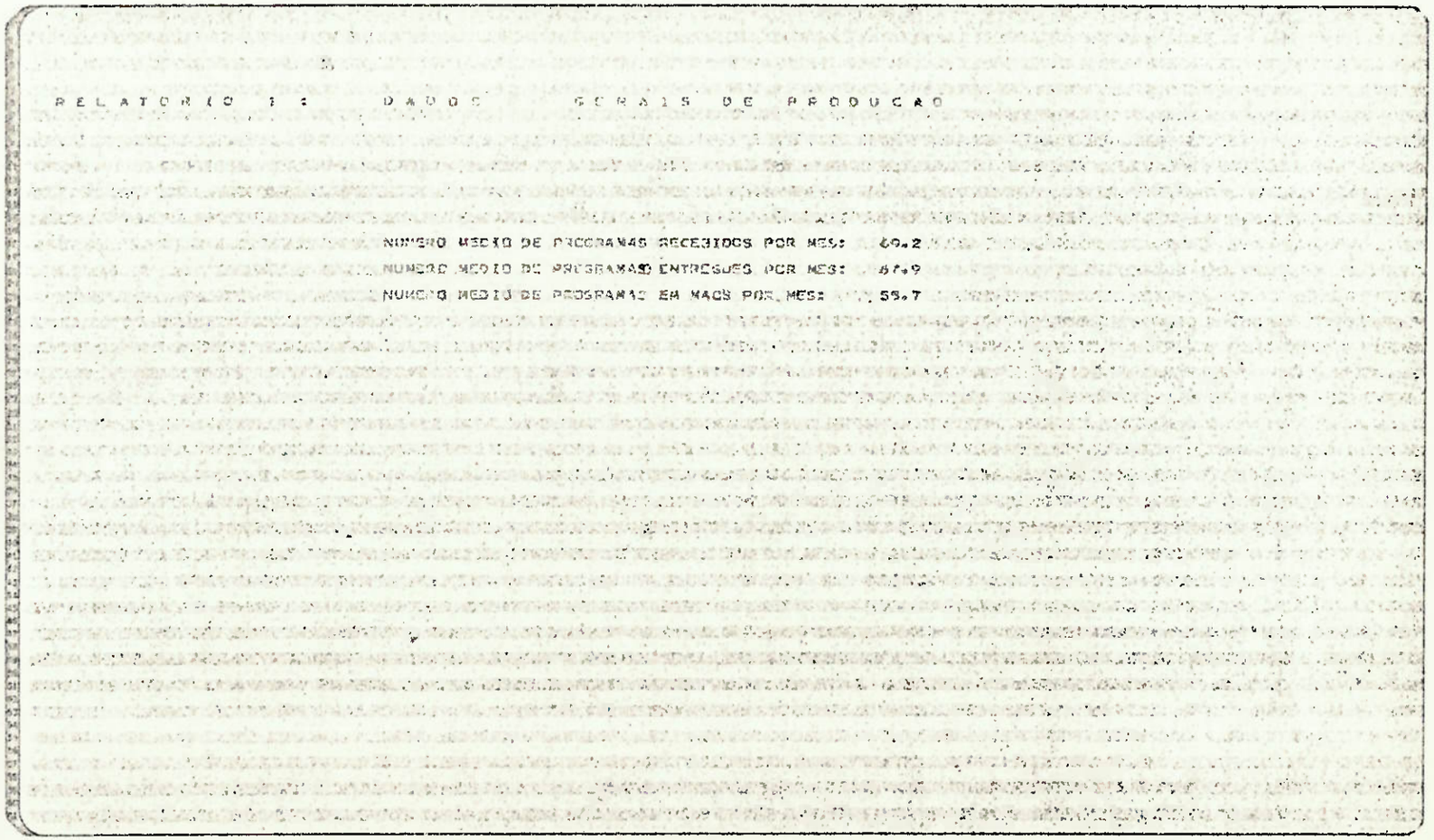


ANEXO III.B - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DA SITUAÇÃO ATUAL PARA P = 44,4%

MES

HALFWORD MATRIX	NEW-EM MAT2	MAN-EM	TOTAL-EM	NEW-REC	MAN-REC	TOTAL-REC	NEW-ENTR	MAN-ENTR	TOTAL-ENTR
ROW/COLUMN	1	2	3	4	5	6	7	8	9
1	26	35	61	18	41	59	11	37	48
2	28	49	77	13	52	65	11	38	49
3	22	49	71	17	53	70	19	53	72
4	29	42	71	18	48	66	15	55	70
5	26	38	64	15	46	61	18	50	68
6	20	28	48	10	53	63	16	63	79
7	19	39	58	16	55	71	17	44	61
8	19	41	60	10	53	63	10	51	61
9	15	47	62	9	59	68	13	53	66
10	31	47	78	24	50	74	8	50	59
11	34	58	92	13	53	66	10	42	52
12	38	75	113	18	55	73	14	38	52

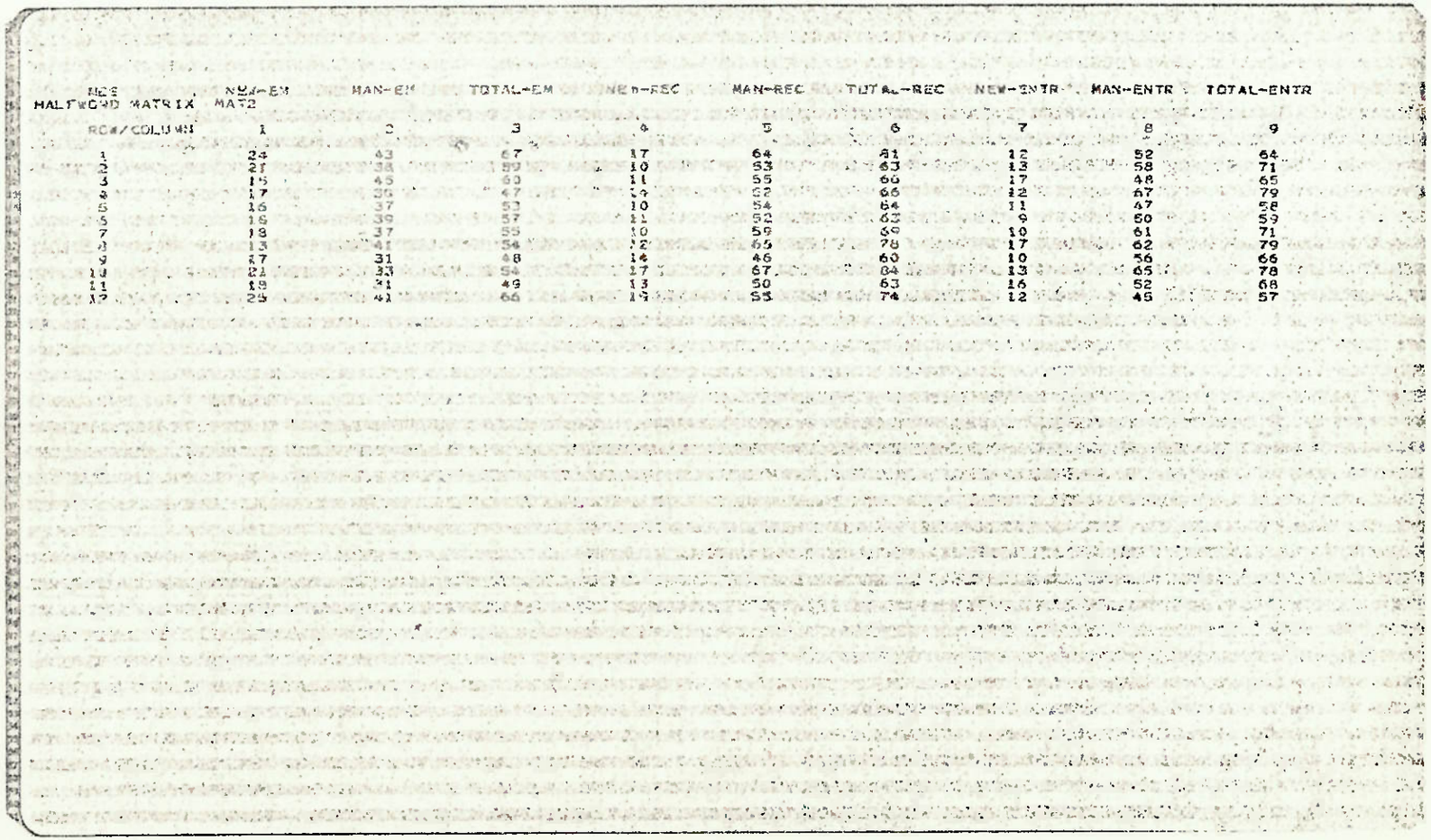
CONTINUAÇÃO DO ANEXO III.B



RELATORIO 1 : DADOS GERAIS DE PRODUCAO

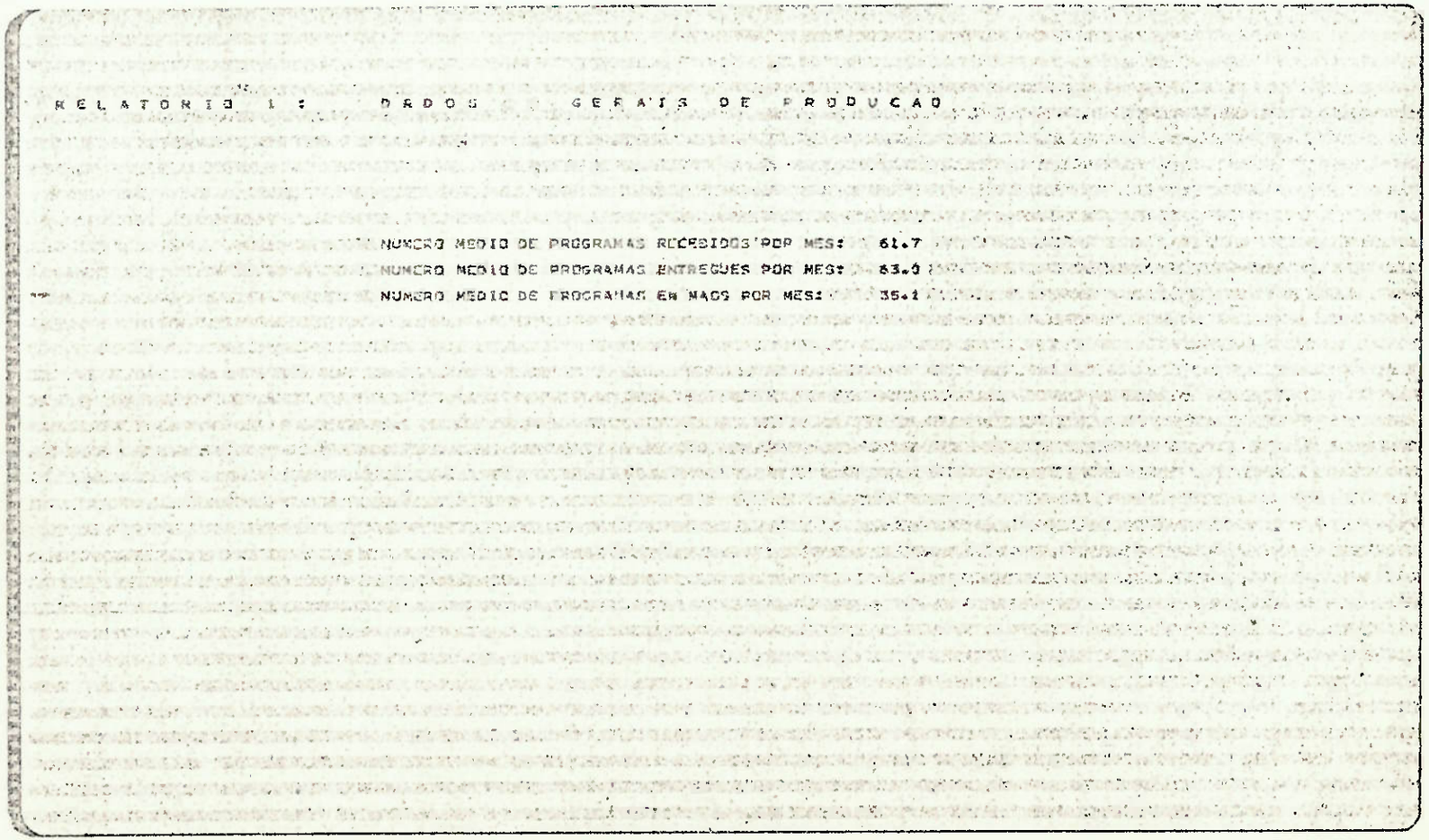
NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 60.2
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 61.9
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 55.7

ANEXO III.C - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DA SITUAÇÃO ATUAL PARA P = 55,6%



NOS HALFWORD MATRIX	NEW-EM MAT2	MAN-EM	TOTAL-EM	NEW-REC	MAN-REC	TOTAL-REC	NEW-ENTR	MAN-ENTR	TOTAL-ENTR
ROW/COLUMN	1	2	3	4	5	6	7	8	9
1	24	43	67	17	64	81	12	52	64
2	21	38	59	10	53	63	13	58	71
3	15	45	60	11	55	66	17	48	65
4	17	30	47	14	52	66	12	67	79
5	16	37	53	10	54	64	11	47	58
6	18	39	57	11	52	63	9	50	59
7	18	37	55	10	58	68	10	61	71
8	13	41	54	12	66	78	17	62	79
9	17	31	48	14	46	60	10	56	66
10	22	32	54	17	67	84	13	65	78
11	19	31	49	13	50	63	16	52	68
12	25	41	66	19	53	74	12	45	57

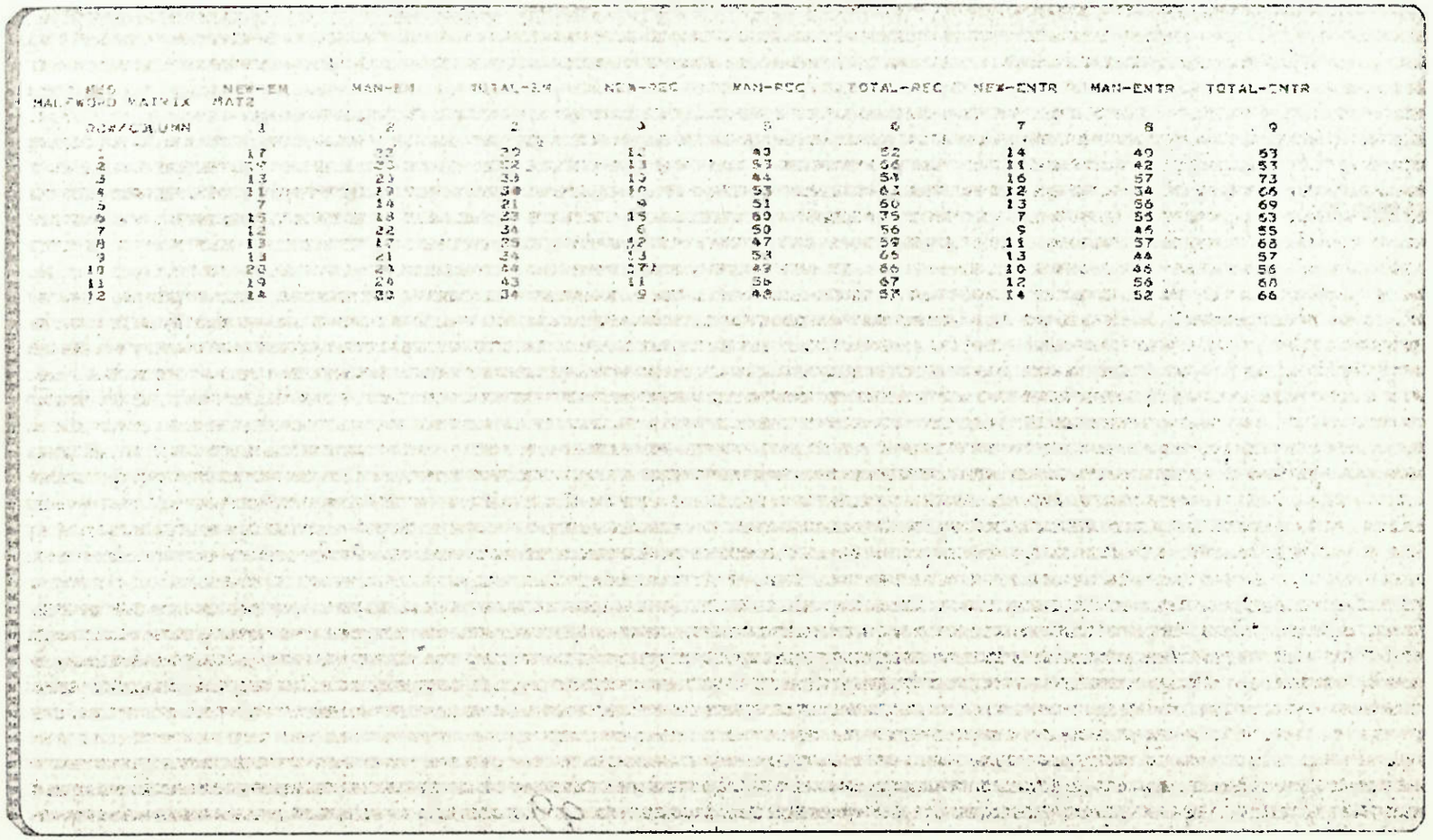
CONTINUAÇÃO DO ANEXO III.C



RELATORIO 1 : DADOS GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 61.7
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 63.0
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 35.1

ANEXO III.D - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DA SITUAÇÃO ATUAL PARA P = 66,6%



MEO MALFWORD MATRIX ROW/COLUMN	NEW-EM MATZ 1	MAN-EM 2	TOTAL-EM 3	NEW-REC 4	MAN-REC 5	TOTAL-REC 6	NEW-ENTR 7	MAN-ENTR 8	TOTAL-ENTR 9
1	17	22	39	12	40	52	14	40	53
2	19	23	42	11	51	62	11	42	51
3	13	25	38	10	44	54	16	57	73
4	11	27	38	10	55	65	12	54	66
5	7	14	21	9	51	60	13	56	69
6	15	18	33	15	60	75	17	53	63
7	12	22	34	6	50	56	9	46	55
8	13	17	30	12	47	59	11	57	68
9	11	21	32	11	53	64	13	44	57
10	20	24	44	11	49	60	10	46	56
11	19	20	43	11	56	67	12	56	68
12	14	25	39	9	48	57	14	52	66

CONTINUAÇÃO DO ANEXO III.D

ANEXO IV - RESULTADOS DA SIMULAÇÃO DA ATUAL SITUAÇÃO COM O MODELO CALIBRADO

(p = 60,0%)

RELATORIO 1 : DADOS GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 65.0
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 64.4
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 45.5

MOS HALFWORD MATRIX

ROW/COLUMN	NEW-EM RATE	MAN-EM	TOTAL-EM	NEW-REC	MAN-REC	TOTAL-REC	NEW-ENTR	MAN-ENTR	TOTAL-ENTR
	1	2	3	4	5	6	7	8	9
1	14	22	36	12	40	52	17	49	66
2	12	31	43	12	53	65	14	44	58
3	19	26	45	7	50	57	10	45	55
4	17	33	50	12	53	65	14	46	60
5	15	37	52	11	61	72	12	57	69
6	27	23	50	10	43	53	7	59	66
7	17	36	53	9	58	50	19	37	56
8	17	32	49	12	47	59	12	61	73
9	17	25	42	16	54	70	16	61	77
10	12	31	43	12	55	67	17	59	76
11	24	19	36	22	40	62	10	49	59
12	25	31	57	14	55	69	12	36	48

RELATORIO 2 : DADOS MEDIOS MENSAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS NOVOS LIBERADOS POR MES:

TIPO	QUANTIDADE
BATCH ATUALIZACAO	3.833
BATCH RELATORIO	1.833
ON LINE COM SCOA	4.083
ON LINE SEM SCOA	3.333
OUTROS TIPOS	0.250
TOTAL MEDIO MENSAL	13.333

NUMERO MEDIO DE MANUTENCOES LIBERADOS POR MES:

TIPO	QUANTIDADE
BATCH ATUALIZACAO	9.166
BATCH RELATORIO	12.250
ON LINE COM SCOA	5.750
ON LINE SEM SCOA	21.583
OUTRAS	2.333
TOTAL MEDIO MENSAL	51.083

RELATORIO 4: APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	38.0 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	43.4 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	18.5 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	34.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	39.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	17.0 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.0 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	2.0 %

RELATORIO 0 1 TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	18.87 DIAS UTEIS
BATCH/RELATORIO	16.36 DIAS UTEIS
BATCH EM GERAL	17.98 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	5.39 DIAS UTEIS
BATCH/RELATORIO	3.31 DIAS UTEIS
BATCH EM GERAL	4.63 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	9.19 DIAS UTEIS
BATCH/RELATORIO	6.22 DIAS UTEIS
BATCH EM GERAL	8.10 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	28.06 DIAS UTEIS
BATCH/RELATORIO	22.58 DIAS UTEIS
BATCH EM GERAL	26.08 DIAS UTEIS

DISTRIBUICAO DOS PROGRAMAS NOVOS

ENTRIES IN TABLE
136

NON-WEIGHTED

UPPER LIMIT	OBSERVED FREQUENCY	PER CENT OF TOTAL
10	51	37.41
12	22	16.12
14	34	29.03
16	25	24.19
18	4	3.22

REMAINING FREQUENCIES ARE ALL ZERO

DISTRIBUICAO DAS MANUTENCOES DE PROGRAMAS

ENTRIES IN TABLE
913

NON-WEIGHTED

UPPER LIMIT	OBSERVED FREQUENCY	PER CENT OF TOTAL
11	114	18.59
13	140	23.81
15	74	12.07
17	251	40.74
19	24	4.56

REMAINING FREQUENCIES ARE ALL ZERO

DADOS DE ATENDIMENTO DE PROGRAMAS NOVOS (TIPO, NUM. DE PROG. ATENDIDOS, TEMPO EFETIVO MEDIO EM 10.HS UTEIS)

TABLE 10
ENTRIES IN TABLE 38 MEAN ARGUMENT 486.526 NON-WEIGHTED

TABLE 12
ENTRIES IN TABLE 22 MEAN ARGUMENT 290.636 NON-WEIGHTED

TABLE 14
ENTRIES IN TABLE 45 MEAN ARGUMENT 260.000 NON-WEIGHTED

TABLE 16
ENTRIES IN TABLE 39 MEAN ARGUMENT 306.922 NON-WEIGHTED

TABLE 18
ENTRIES IN TABLE 3 MEAN ARGUMENT 240.000 NON-WEIGHTED

DADOS DE ATENDIMENTO DE MANUTENCOES (TIPO, NUM. DE PROG. ATENDIDOS, TEMPO EFETIVO MEDIO EM 10.HS UTEIS)

TABLE 11
ENTRIES IN TABLE 106 MEAN ARGUMENT 128.632 NON-WEIGHTED

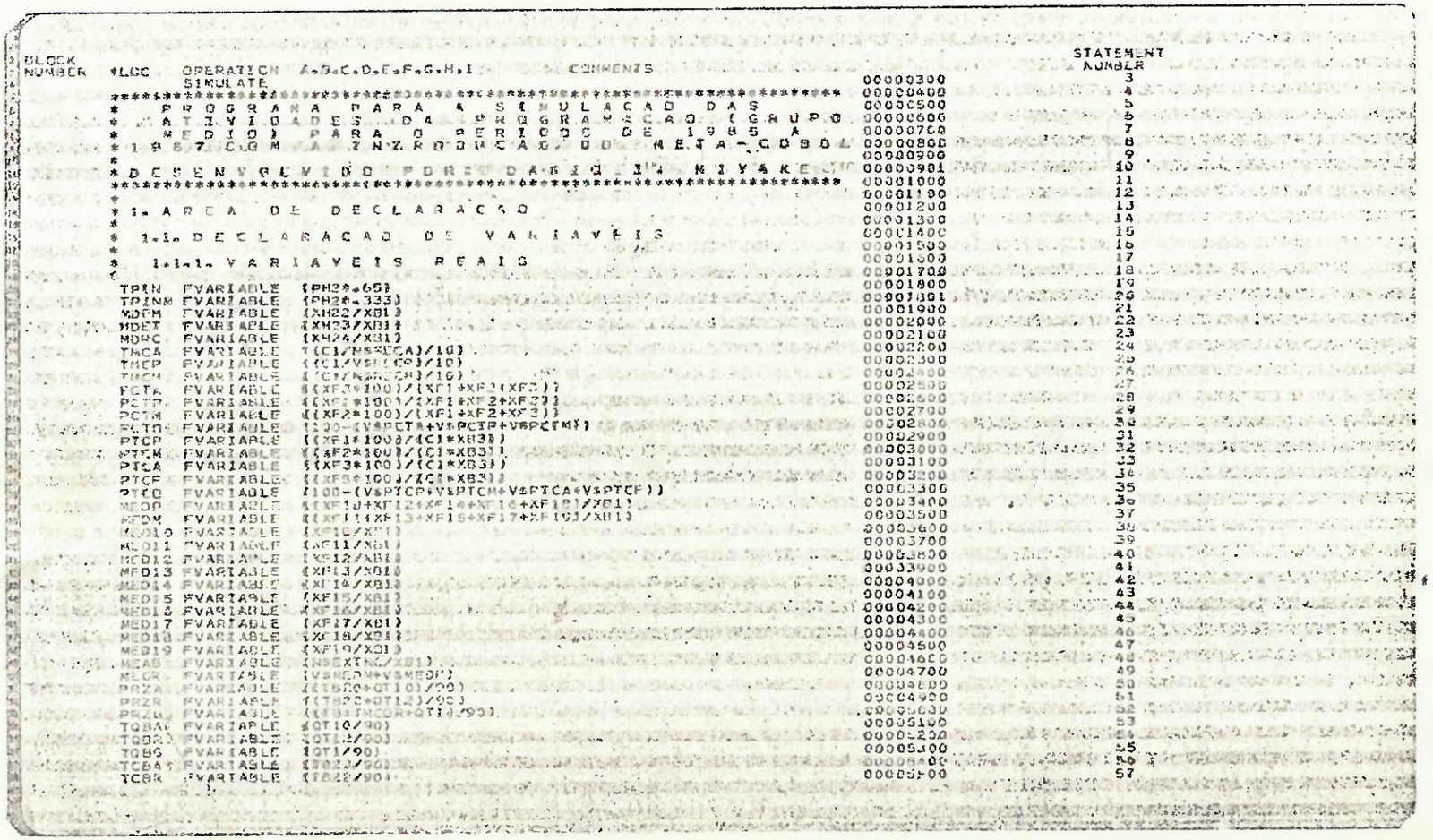
TABLE 13
ENTRIES IN TABLE 142 MEAN ARGUMENT 119.788 NON-WEIGHTED

TABLE 15
ENTRIES IN TABLE 64 MEAN ARGUMENT 174.219 NON-WEIGHTED

TABLE 17
ENTRIES IN TABLE 245 MEAN ARGUMENT 79.224 NON-WEIGHTED

TABLE 19
ENTRIES IN TABLE 26 MEAN ARGUMENT 30.423 NON-WEIGHTED

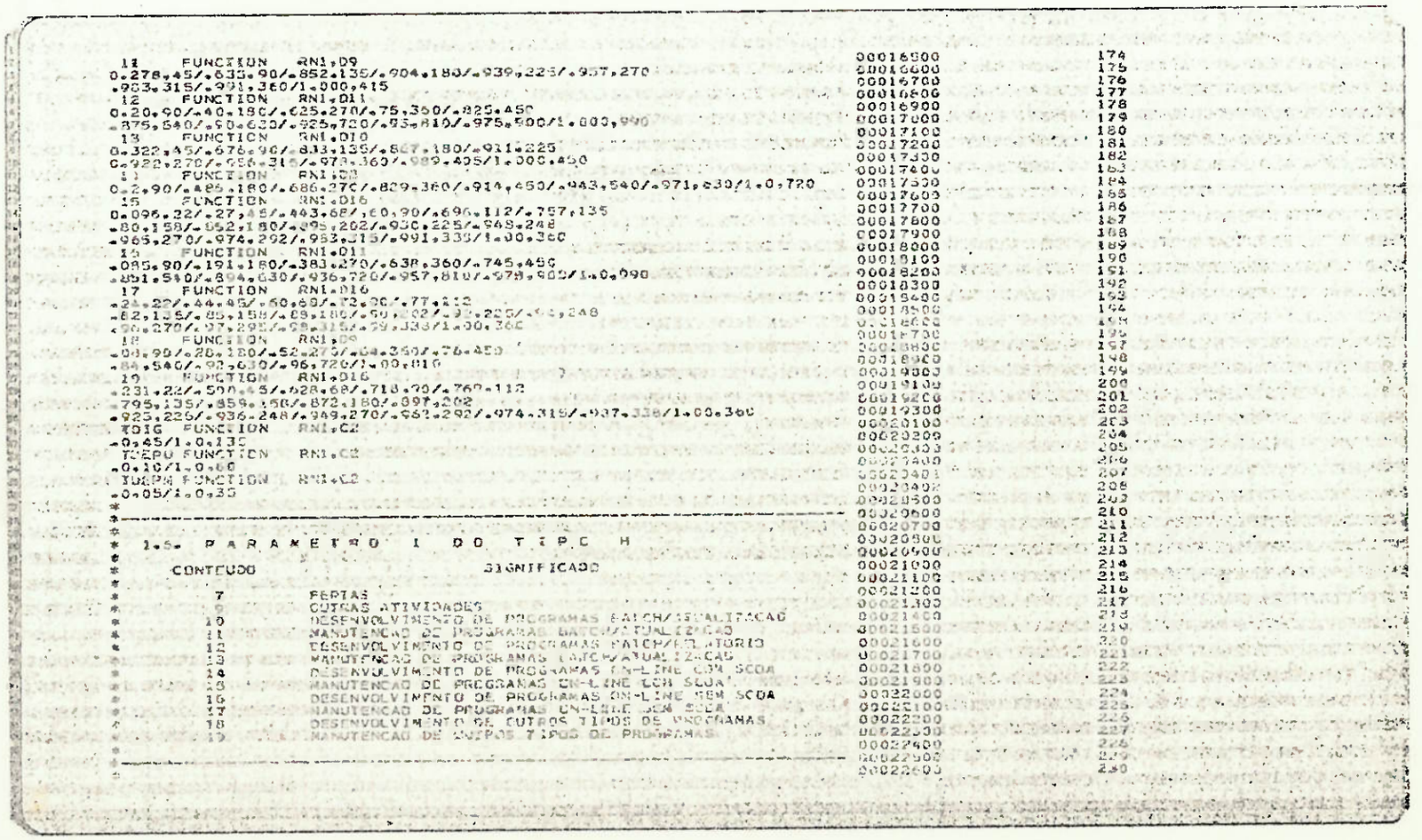
ANEXO V - PROGRAMA FONTE PARA A SIMULAÇÃO DO PERÍODO DE 1985 a 1987 CONTANDO COM A INTRODUÇÃO DO META COBOL



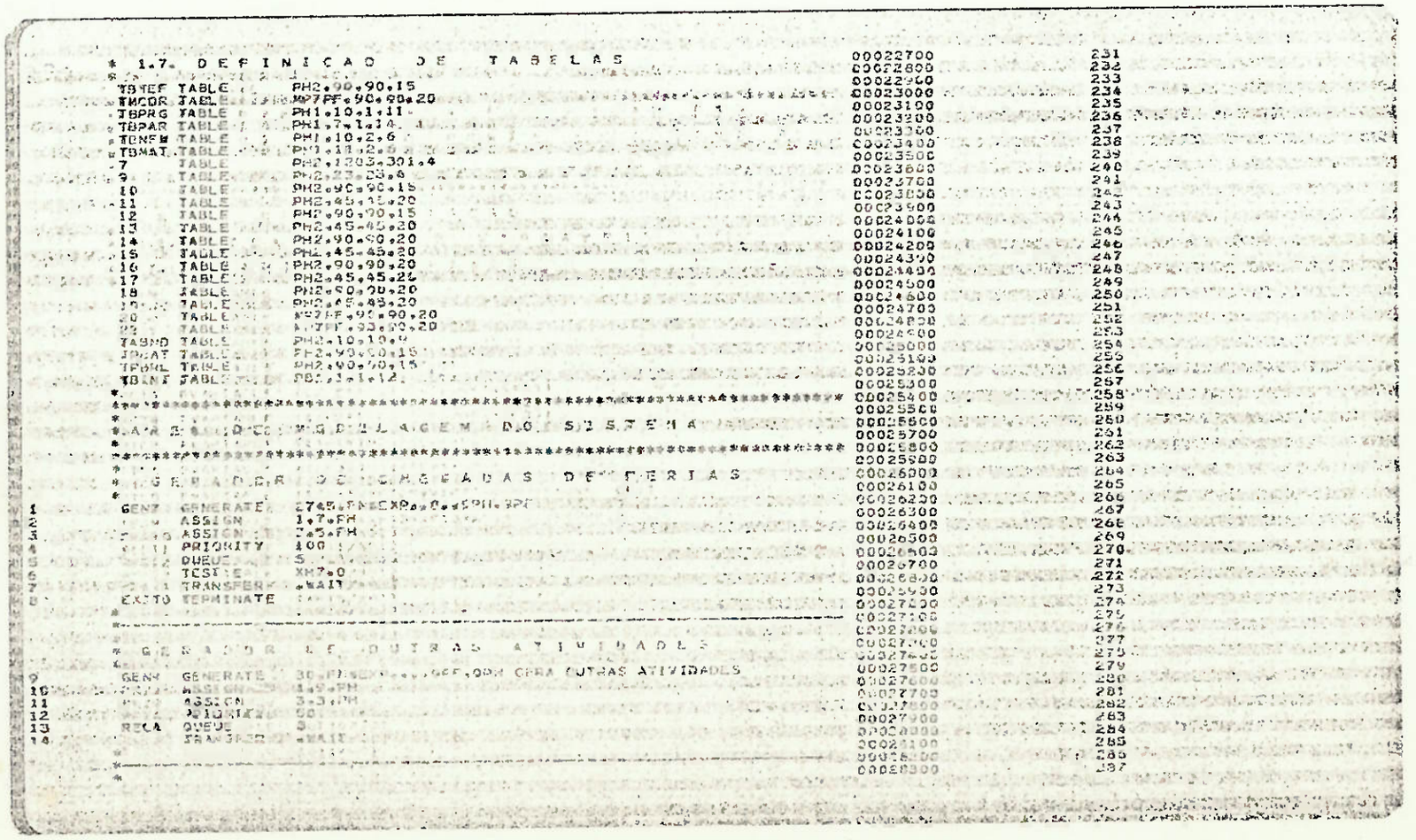
```
BLOCK
NUMBER  *LOC   OPERATION  A,B,C,D,E,F,G,H,I        COMMENTS                                                    STATEMENT
                                                                                                                NUMBER
               SIMULATE                                                                          00000300         3
        ********************************************************************************         00000400         4
        *    PROGRAMA  PARA  A  SIMULACAO  DAS                                                   00000500         5
        *  ATIVIDADES  DA  PROGRAMACAO  (GRUPO                                                   00000600         6
        *    MEDIO)  PARA  O  PERIODO  DE  1985  A                                               00000700         7
        *1987  COM  A  INTRODUCAO  DO  MEJA-COBOL                                               00000800         8
        *                                                                                       00000900         9
        *DESENVOLVIDO  POR:  DARIO  I.  MIYAKE                                                  00000901        10
        ********************************************************************************         00001000        11
        *                                                                                       00001100        12
        *1. AREA  DE  DECLARACAO                                                                00001200        13
        *                                                                                       00001300        14
        *  1.1. DECLARACAO  DE  VARIAVEIS                                                       00001400        15
        *                                                                                       00001500        16
        *    1.1.1. VARIAVEIS  REAIS                                                            00001600        17
        *                                                                                       00001700        18
         TPRN   FVARIABLE  (PH2*.65)                                                           00001800        19
         TPRNM  FVARIABLE  (PH2*.333)                                                          00001801        20
         MDPM   FVARIABLE  (XH22/XB1)                                                          00001900        21
         MDET   FVARIABLE  (XH23/XB1)                                                          00002000        22
         MDRC   FVARIABLE  (XH24/XB1)                                                          00002100        23
         TMCA   FVARIABLE  1-(C1/N$*CCA)/10)                                                   00002200        24
         TMCP   FVARIABLE  1-(C1/V$*CCP)/10)                                                   00002300        25
         TMCM   FVARIABLE  1-(C1/N$*CCM)/10)                                                   00002400        26
         PCTA   FVARIABLE  ((XF3*100)/(XF1+XF2+XF3))                                           00002500        27
         PCTP   FVARIABLE  ((XF1*100)/(XF1+XF2+XF3))                                           00002600        28
         PCTM   FVARIABLE  ((XF2*100)/(XF1+XF2+XF3))                                           00002700        29
         PCTD   FVARIABLE  (100-(V$PCTA+V$PCTP+V$PCTM))                                        00002800        30
         PTCP   FVARIABLE  ((XF1*100)/(C1*XB3))                                                00002900        31
         PTCM   FVARIABLE  ((XF2*100)/(C1*XB3))                                                00003000        32
         PTCA   FVARIABLE  ((XF3*100)/(C1*XB3))                                                00003100        33
         PTCF   FVARIABLE  ((XF3*100)/(C1*XB3))                                                00003200        34
         PTCO   FVARIABLE  (100-(V$PTCP+V$PTCM+V$PTCA+V$PTCF))                                 00003300        35
         MEDP   FVARIABLE  ((XF10+XF12+XF16+XF18)/XB1)                                         00003400        36
         MEDM   FVARIABLE  ((XF11+XF13+XF15+XF17+XF19)/XB1)                                    00003500        37
         MED10  FVARIABLE  (XF10/XB1)                                                          00003600        38
         MED11  FVARIABLE  (XF11/XB1)                                                          00003700        39
         MED12  FVARIABLE  (XF12/XB1)                                                          00003800        40
         MED13  FVARIABLE  (XF13/XB1)                                                          00003900        41
         MED14  FVARIABLE  (XF14/XB1)                                                          00004000        42
         MED15  FVARIABLE  (XF15/XB1)                                                          00004100        43
         MED16  FVARIABLE  (XF16/XB1)                                                          00004200        44
         MED17  FVARIABLE  (XF17/XB1)                                                          00004300        45
         MED18  FVARIABLE  (XF18/XB1)                                                          00004400        46
         MED19  FVARIABLE  (XF19/XB1)                                                          00004500        47
         MEAB   FVARIABLE  (N$EXTNC/XB1)                                                       00004600        48
         MEGR   FVARIABLE  (V$MEDM+V$MEDP)                                                     00004700        49
         PRZA   FVARIABLE  ((TB20+QT10)/90)                                                    00004800        50
         PRZR   FVARIABLE  ((TB21+QT12)/90)                                                    00004900        51
         PRZG   FVARIABLE  ((TB31+MCDR+QT10)/90)                                               00005000        52
         TQBA   FVARIABLE  (QT10/90)                                                           00005100        53
         TQBR   FVARIABLE  (QT12/90)                                                           00005200        54
         TQBG   FVARIABLE  (QT1/90)                                                            00005300        55
         TCBA   FVARIABLE  (TB23/90)                                                           00005400        56
         TCBR   FVARIABLE  (TB22/90)                                                           00005500        57
```

```
TCBG   FVARIABLE  (TB1IXCOR/90)
TEBA   FVARIABLE  (TB10/90)
TCBR   FVARIABLE  (TB12/90)
TEBG   FVARIABLE  (TB5*TBICF/90)
TMETA  FVARIABLE  (PH2*.7)
*
*-------------------------------------------------------------
*
*   1.1.2. V A R I A V E I S   I N T E I R A S
*
NEW    VARIABLE   (PH182)
TLAST  VARIABLE   (PH2-PH5*PH6)
DAYAF  VARIABLE   ((TC1+90)/90)*90-C1)
SOTQ   VARIABLE   (Q1+Q2+Q3)
PNEA   VARIABLE   (XH10+XH12+XH14+XH16+XH18)
PNEM   VARIABLE   (Q1+XH10+XH12+XH14+XH15+XH13+Q4+S$CONCT+S$DIGIT)
MPEA   VARIABLE   (XH11+XH13+XH15+XH17+XH19)
MPEM   VARIABLE   (C2+V$MPEA)
TATA   VARIABLE   (XH7+XH9+V$PNEA+V$MPEA)
TLEA   VARIABLE   (V$PNEA+V$MPEA)
TLEX   VARIABLE   (V$PNEM+V$MPEM)
TPEN   VARIABLE   (N$EXIT1+N$EXIT5)
TMEN   VARIABLE   (N$EXIT2+N$EXIT4)
PENT   VARIABLE   (N$EXIT1+N$EXIT5-XH5)
MENT   VARIABLE   (N$EXIT2+N$EXIT4-XH6)
TENT   VARIABLE   (V$PENT+V$MENT)
DEMP   VARIABLE   (V$RECP-XH20)
DEMA   VARIABLE   (N$RECM-XH21)
DEMT   VARIABLE   (V$DEMP+V$DEMA)
RECP   VARIABLE   (N$RCP10+N$RCP12+N$RCP14+N$RCP16+N$RCP18)
TBTC   VARIABLE   (PH1*10)
*
*-------------------------------------------------------------
*
*   1.1.3. V A R I A V E L   B O O L E A N A
*
META   BVARIABLE  PH1*E*10+PH1*E*12
NOPRO  BVARIABLE  PH1*E*7+PH1*E*9
PROCE  BVARIABLE  PH1*E*16+PH1*E*18
*
*-------------------------------------------------------------
*
*  1.2. D E C L A R A C A O   D E   R E G I S T R A D O R E S
*
*    1.2.1. R E G I S T R A D O R E S   T I P O   0
*
      INITIAL   X81*12          NUMERO DE MESES SIMULADOS
      INITIAL   X82*0           CONTADOR DE MESES
      INITIAL   X83*0           NUM. DE PROGRAMADORES
      INITIAL   X84-X81)*0
*
*-------------------------------------------------------------
*
*    1.2.2. R E G I S T R A D O R E S   T I P O   H
*
      INITIAL   XH5*0           GUARDA V$TPEN DO MES ANTERIOR
      INITIAL   XH6*0           "   V$TMEN  "  "   "
```

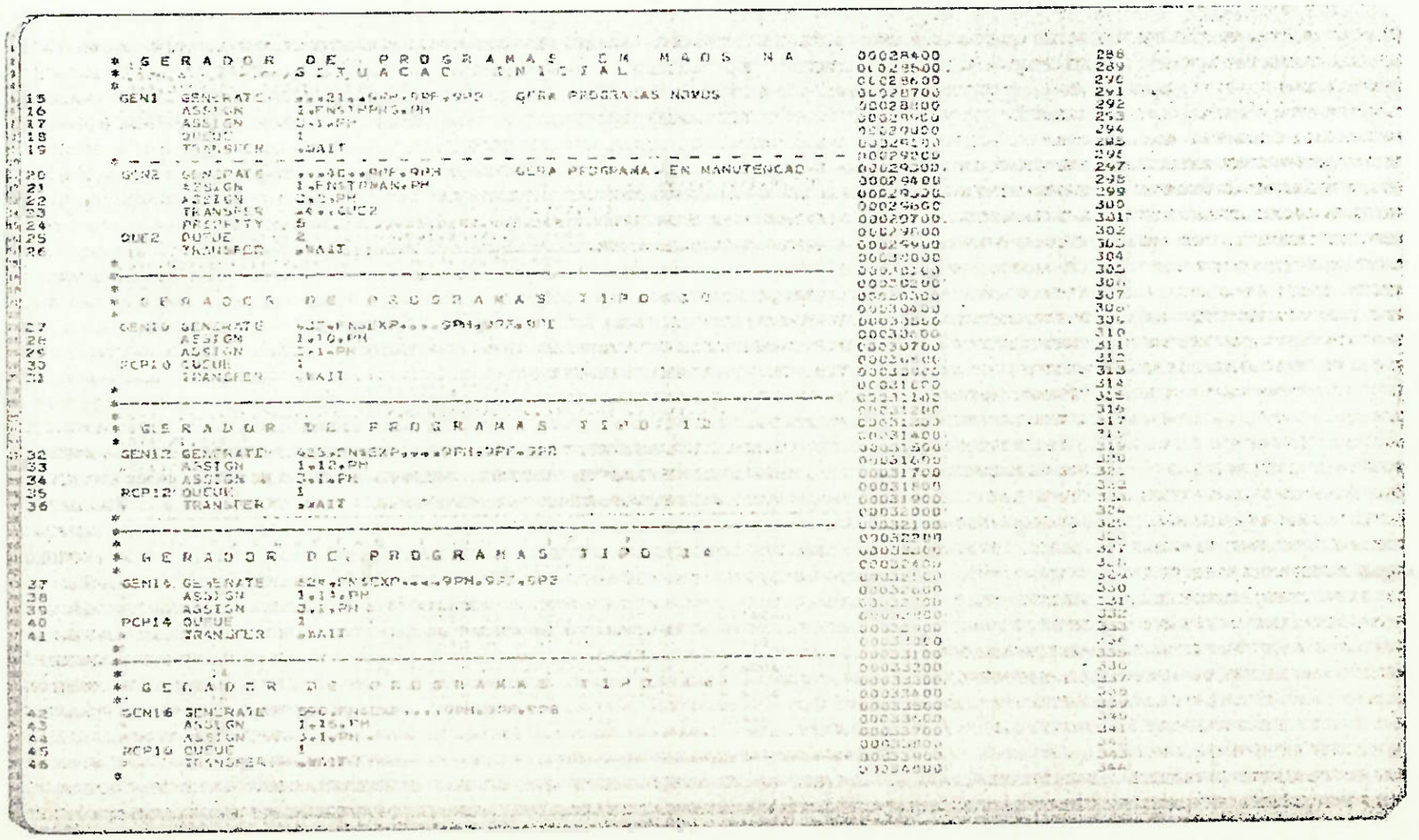
```
         INITIAL   XH7-XH19,0          NUM. DE ATIVIDADES EM ATENDI/   00011300
         INITIAL   XH20,0              GUARDA VSRECM DO MES ANTERIOR   00011400
         INITIAL   XH21,0                   "   VSRECM  "  "    "     00011500
         INITIAL   XH22,0              ACUMULA VSTLEM TODO MES         00011600
         INITIAL   XH23,0                   "   VSTENT  "  "    "     00011700
         INITIAL   XH24,0                   "   VSDEMT  "  "    "     00011800
*                                                                      00011900
*--------------------------------------------------------------------  00012000
*                                                                      00012100
*  1.2.3.  REGISTRADORES TIPO F                                        00012200
*                                                                      00012300
         INITIAL   XF1-XF5,0                                          00012400
         INITIAL   XF7-XF19,0          TOTAL DE HORAS POR ATIVIDADE   00012500
*                                                                      00012600
*--------------------------------------------------------------------  00012700
*                                                                      00012800
*  1.2.4.  REGISTRADORES TIPO L                                        00012900
*                                                                      00013000
         INITIAL   XL1-XL50,0          VALORES MEDIOS REAIS P/ RELAT.  00013100
*                                                                      00013200
*--------------------------------------------------------------------  00013300
*                                                                      00013400
*  1.3.  DECLARACAO DE MATRIZES                                        00013500
*                                                                      00013600
MAT1     MATRIX    MH,12,8           MATRIZ PARA RELATORIO            00013700
MAT2     MATRIX    MH,10,9           MATRIZ PARA RELATORIO            00013800
MAT3     MATRIX    MH,12,3           MATRIZ PARA RELATORIO            00013900
MAT4     MATRIX    MH,12,9           MATRIZ PARA RELATORIO            00014000
*                                                                      00014100
*--------------------------------------------------------------------  00014200
*                                                                      00014300
*  1.4.  DECLARACAO DE DEPOSITOS                                       00014400
*                                                                      00014500
         STORAGE   SSWORK,8          NUM. DE ELEMENTOS POR EQUIPE     00014600
         STORAGE   SSDIGIT,20        VAGAS EM DIGIT                   00014700
         STORAGE   SSCONCT,20        VAGAS EM CONCT                   00014800
*                                                                      00014900
*--------------------------------------------------------------------  00015000
*                                                                      00015100
*  1.5.  ESPECIFICACAO DE FUNCOES                                      00015200
*                                                                      00015300
 EXP      FUNCTION  RN1,C24                                           00015400
0,0,0,0/1,,104/2,,222/3,,355/4,,509/5,,69/6,,915                       00015500
0,7,1,2/2,75,1,28/3,,84,1,30/4,28,2,12/5,2,3                           00015600
0,92,2,52/4,94,5,81/6,93,8,92/9,96,12/16,97,25                         00015700
,98,39/,99,4,64/,995,5,3/,998,6,2/,999,7/,9997,8                       00015800
 TDMG     FUNCTION  RN1,D5                                            00015900
0,2830,10/,5650,12/,7610,14/,9150,16/1,303,18                          00016000
 TMAN     FUNCTION  RN1,D5                                            00016100
0,1810,11/,4120,13/,5220,15/,8570,17/1,0000,19                         00016200
 TABAN    FUNCTION  RN1,C2                                            00016300
0,0,10/1,0,70                                                          00016400
 7        FUNCTION  RN1,D3                                            00016500
0,33,1203/,69,1506/1,0,1805                                            00016600
 9        FUNCTION  RN1,D4                                            00016700
0,38,22/,85,85/,96,,67/1,00,90                                         00016800
 10       FUNCTION  RN1,D13                                           00016900
0,0930,90/,2310,180/,3970,270/,5420,360/,6830,450/,8150,540            00017000
,872,630/,902,720/,937,810/,958,900/,969,990/,984,1080/1,0,1170        00017100
```



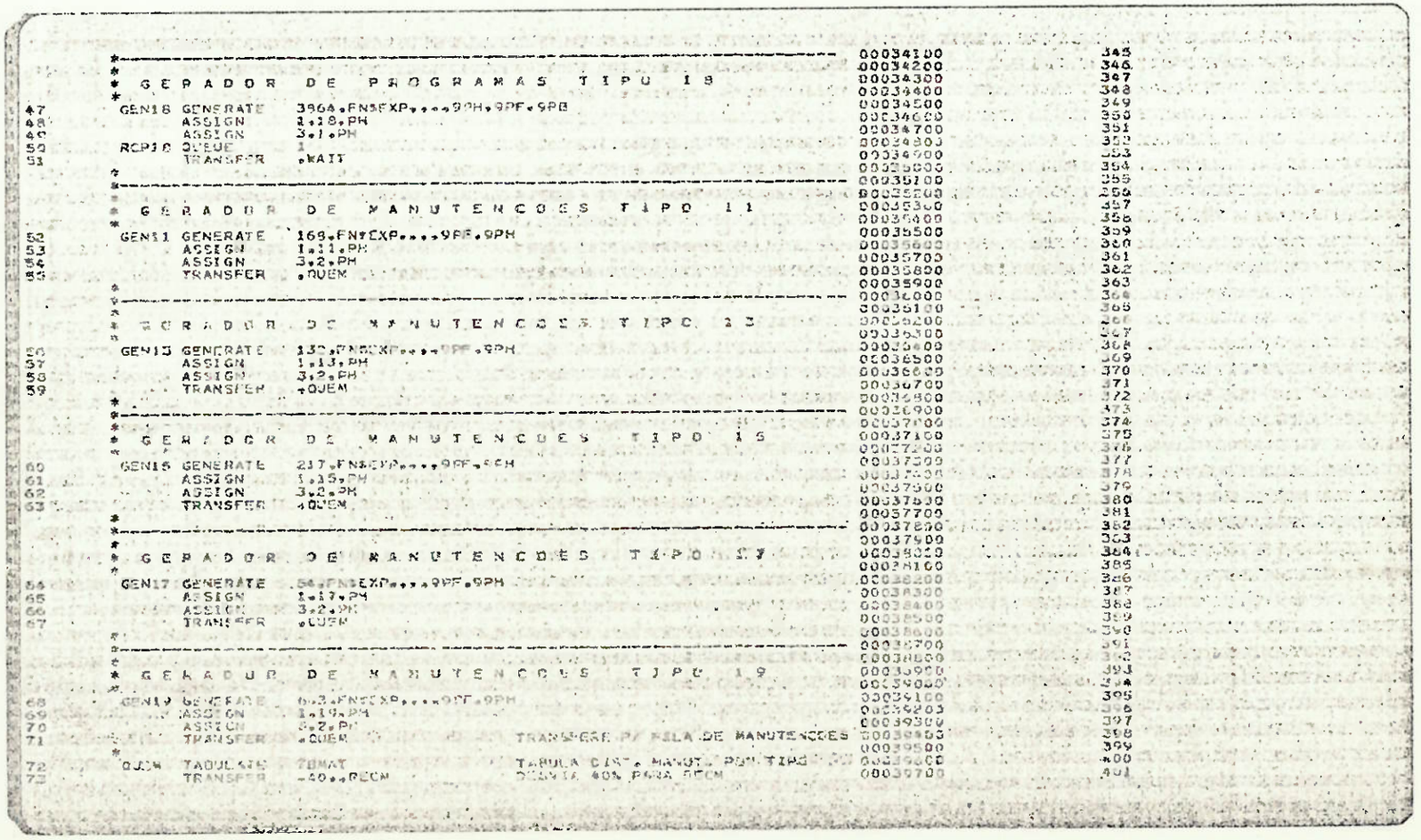
```
 11     FUNCTION    RN1,D9                                                00016500                174
0,278,45/,633,90/,852,135/,904,180/,939,225/,957,270                      00016600                175
,983,315/,991,360/1,000,415                                               00016700                176
 12     FUNCTION    RN1,D11                                               00016800                177
0,20,90/,40,180/,625,270/,75,360/,825,450                                 00016900                178
,875,540/,90,630/,925,720/,95,810/,975,900/1,000,990                      00017000                179
 13     FUNCTION    RN1,D10                                               00017100                180
0,322,45/,676,90/,833,135/,867,180/,9114,225                              00017200                181
0,922,270/,956,315/,973,360/,989,405/1,000,450                            00017300                182
 14     FUNCTION    RN1,D8                                                00017400                183
0,2,90/,488,180/,686,270/,809,360/,914,450/,943,540/,971,630/1,0,720      00017500                184
 15     FUNCTION    RN1,D16                                               00017600                185
0,096,22/,27,45/,443,68/,60,90/,696,112/,757,135                          00017700                186
,80,158/,852,180/,895,202/,930,225/,948,248                               00017800                187
,965,270/,974,292/,983,315/,991,338/1,00,360                              00017900                188
 16     FUNCTION    RN1,D11                                               00018000                189
,095,90/,191,180/,383,270/,638,360/,745,450                               00018100                190
,891,540/,894,630/,936,720/,957,810/,978,900/1,0,990                      00018200                191
 17     FUNCTION    RN1,D16                                               00018300                192
,24,22/,44,45/,60,69/,72,90/,77,112                                       00018400                193
,82,135/,89,158/,89,180/,90,202/,92,225/,94,248                           00018500                194
,96,270/,97,295/,98,315/,99,338/1,00,360                                  00018600                195
 18     FUNCTION    RN1,D8                                                00018700                196
,04,90/,28,180/,52,270/,84,360/,76,450                                    00018800                197
,84,540/,92,630/,96,720/1,00,810                                          00018900                198
 19     FUNCTION    RN1,D16                                               00019000                199
,231,22/,500,45/,628,68/,718,90/,760,112                                  00019100                200
,795,135/,859,158/,872,180/,891,202                                       00019200                201
,925,225/,936,248/,949,270/,961,292/,974,315/,987,338/1,00,360            00019300                202
 DIG   FUNCTION    RN1,C2                                                00020000                203
,0,45/1,0,135                                                             00020100                204
 TEMPO FUNCTION    RN1,C2                                                00020200                205
,0,10/1,0,60                                                              00020300                206
 TEMPM FUNCTION    RN1,C2                                                00020400                207
,0,05/1,0,30                                                              00020500                208
*                                                                         00020600                209
*-------------------------------------------------------------------      00020700                210
*                                                                         00020800                211
*  1.5-  P A R A M E T R O   I   D O   T I P O   H                        00020900                212
*                                                                         00021000                213
*     CONTEUDO                    SIGNIFICADO                             00021100                214
*                                                                         00021200                215
*        7          FERIAS                                              00021300                216
*        8          OUTRAS ATIVIDADES                                   00021400                217
*       10          DESENVOLVIMENTO DE PROGRAMAS BATCH/ATUALIZACAO      00021500                218
*       11          MANUTENCAO DE PROGRAMAS BATCH/ATUALIZACAO           00021600                219
*       12          DESENVOLVIMENTO DE PROGRAMAS BATCH/RELATORIO        00021700                220
*       13          MANUTENCAO DE PROGRAMAS BATCH/ATUALIZACAO           00021800                221
*       14          DESENVOLVIMENTO DE PROGRAMAS ON-LINE COM SCOA       00021900                222
*       15          MANUTENCAO DE PROGRAMAS ON-LINE COM SCOA            00022000                223
*       16          DESENVOLVIMENTO DE PROGRAMAS ON-LINE SEM SCOA       00022100                224
*       17          MANUTENCAO DE PROGRAMAS ON-LINE SEM SCOA            00022200                225
*       18          DESENVOLVIMENTO DE OUTROS TIPOS DE PROGRAMAS        00022300                226
*       19          MANUTENCAO DE OUTROS TIPOS DE PROGRAMAS             00022400                227
*                                                                         00022500                228
*-------------------------------------------------------------------      00022600                229
*                                                                         00022700                230
```



```
* 1.7. DEFINICAO DE TABELAS
TBTEF TABLE PH2,90,90,15
TMCOR TABLE M$7FF,90,90,20
TBPRG TABLE PH1,10,1,11
TBPAR TABLE PH1,7,1,14
TONFB TABLE PH1,10,2,6
TBMAT TABLE PH1,10,2,9
7 TABLE PH2,1205,301,4
9 TABLE PH2,23,23,6
10 TABLE PH2,90,90,15
11 TABLE PH2,45,45,20
12 TABLE PH2,90,90,15
13 TABLE PH2,45,45,20
14 TABLE PH2,90,90,20
15 TABLE PH2,45,45,20
16 TABLE PH2,90,90,20
17 TABLE PH2,45,45,20
18 TABLE PH2,90,90,20
19 TABLE PH2,45,45,20
20 TABLE M$7FF,90,90,20
22 TABLE M$7FF,90,90,20
TASND TABLE PH2,10,10,9
JDEAT TABLE PH2,90,90,15
TFORL TABLE PH2,90,90,15
TBANF TABLE PH1,1,1,12
*
*********************************************************************
*
* PARAMETRIZACAO DA MODELAGEM DO SISTEMA
*
*********************************************************************
*
* GERADOR DE CHEGADAS DE FERIAS
*
1 GENF GENERATE E7#5,FNEXP,0,0,,SPH,3PT
2 ASSIGN 1,7,FH
3 ASSIGN 2,5,FH
4 PRIORITY 100
5 QUEUE 5
6 TESTE XH7,0
7 TRANSFER ,WAIT
8 EXITO TERMINATE
*
*---------------------------------------------------------------------
*
* GERADOR DE OUTRAS ATIVIDADES
*
9 GENA GENERATE 30,FN$EXP,,,,SPH,GERA OUTRAS ATIVIDADES
10 ASSIGN 1,9,FH
11 ASSIGN 3,3,FH
12 PRIORITY 90
13 RECA QUEUE 9
14 TRANSFER ,WAIT
*
*---------------------------------------------------------------------
*
```

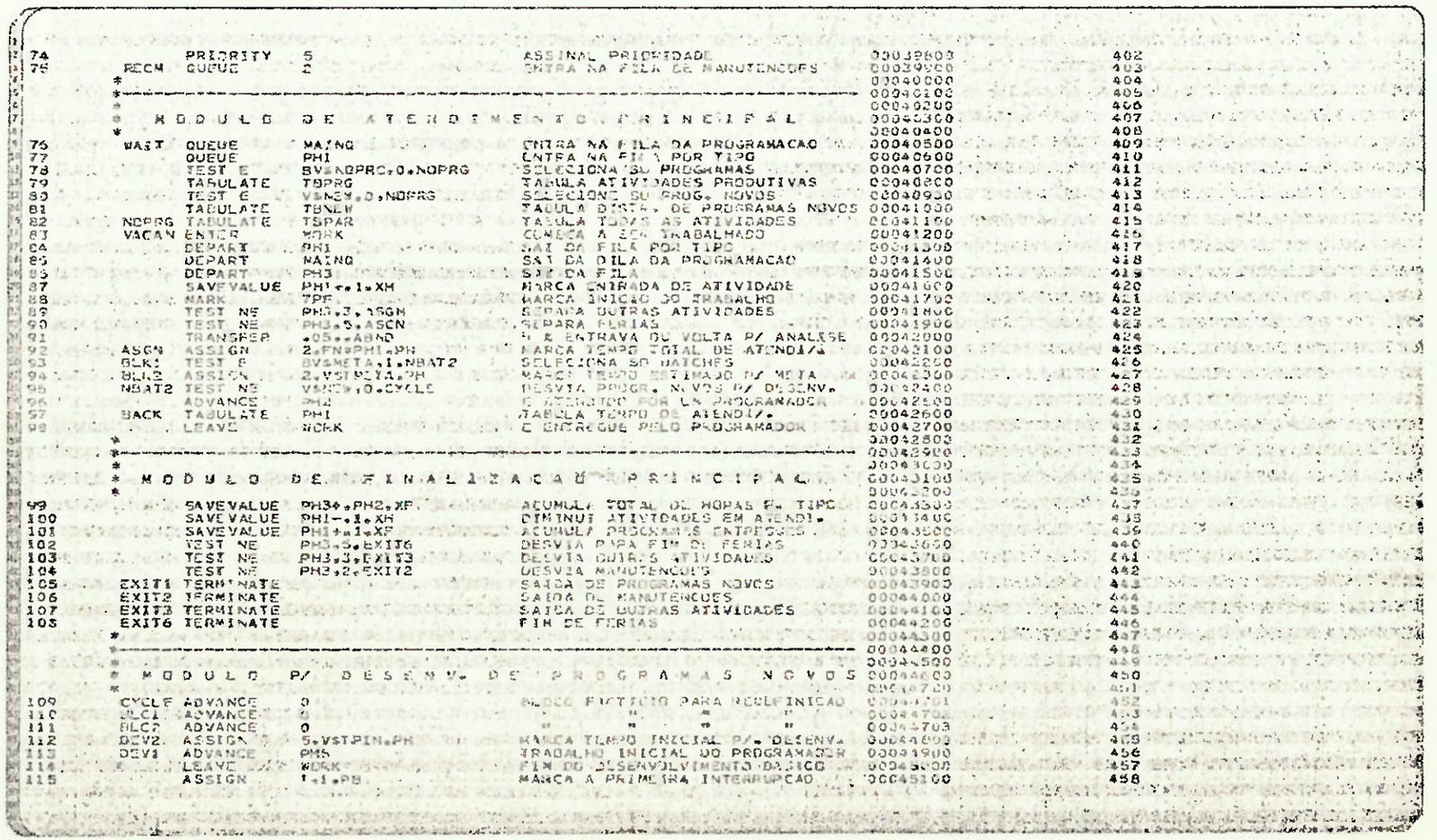


```
*  G E R A D O R   D E   P R O G R A M A S   C O M   M A O S   N A
*                  S I T U A C A O   I N I C I A L
*
GEN1   GENERATE   ***21***GPH.GPF.GPP      GERA PROGRAMAS NOVOS
       ASSIGN     1.FN$TPPNS.PH
       ASSIGN     2.1.PH
       QUEUE      1
       TRANSFER   .WAIT
*
GUN2   GENERATE   ***36***GPF.GPH          GERA PROGRAMAS EM MANUTENCAO
       ASSIGN     1.FN$TPMAN.PH
       ASSIGN     2.2.PH
       TRANSFER   .A**.QUE2
       PRIORITY   5
QUE2   QUEUE      2
       TRANSFER   .WAIT
*
*
*  G E R A D O R   D E   P R O G R A M A S   T I P O   1 0
*
GEN10  GENERATE   ***1*N$1XP****9PH.9PF.9P1
       ASSIGN     1.10.PH
       ASSIGN     2.1.PH
PCP10  QUEUE      1
       TRANSFER   .WAIT
*
*
*  G E R A D O R   D E   P R O G R A M A S   T I P O   1 2
*
GEN12  GENERATE   ***1*FN$EXP****9PH.9PF.9P2
       ASSIGN     1.12.PH
       ASSIGN     2.1.PH
PCP12  QUEUE      1
       TRANSFER   .WAIT
*
*
*  G E R A D O R   D E   P R O G R A M A S   T I P O   1 4
*
GEN14  GENERATE   ***1*FN$EXP****9PH.9PF.9P3
       ASSIGN     1.14.PH
       ASSIGN     2.1.PH
PCP14  QUEUE      1
       TRANSFER   .WAIT
*
*
*  G E R A D O R   D E   P R O G R A M A S   T I P O   1 6
*
GEN16  GENERATE   ***1*FN$EXP****9PH.9PF.9P8
       ASSIGN     1.16.PH
       ASSIGN     2.1.PH
PCP16  QUEUE      1
       TRANSFER   .WAIT
*
```

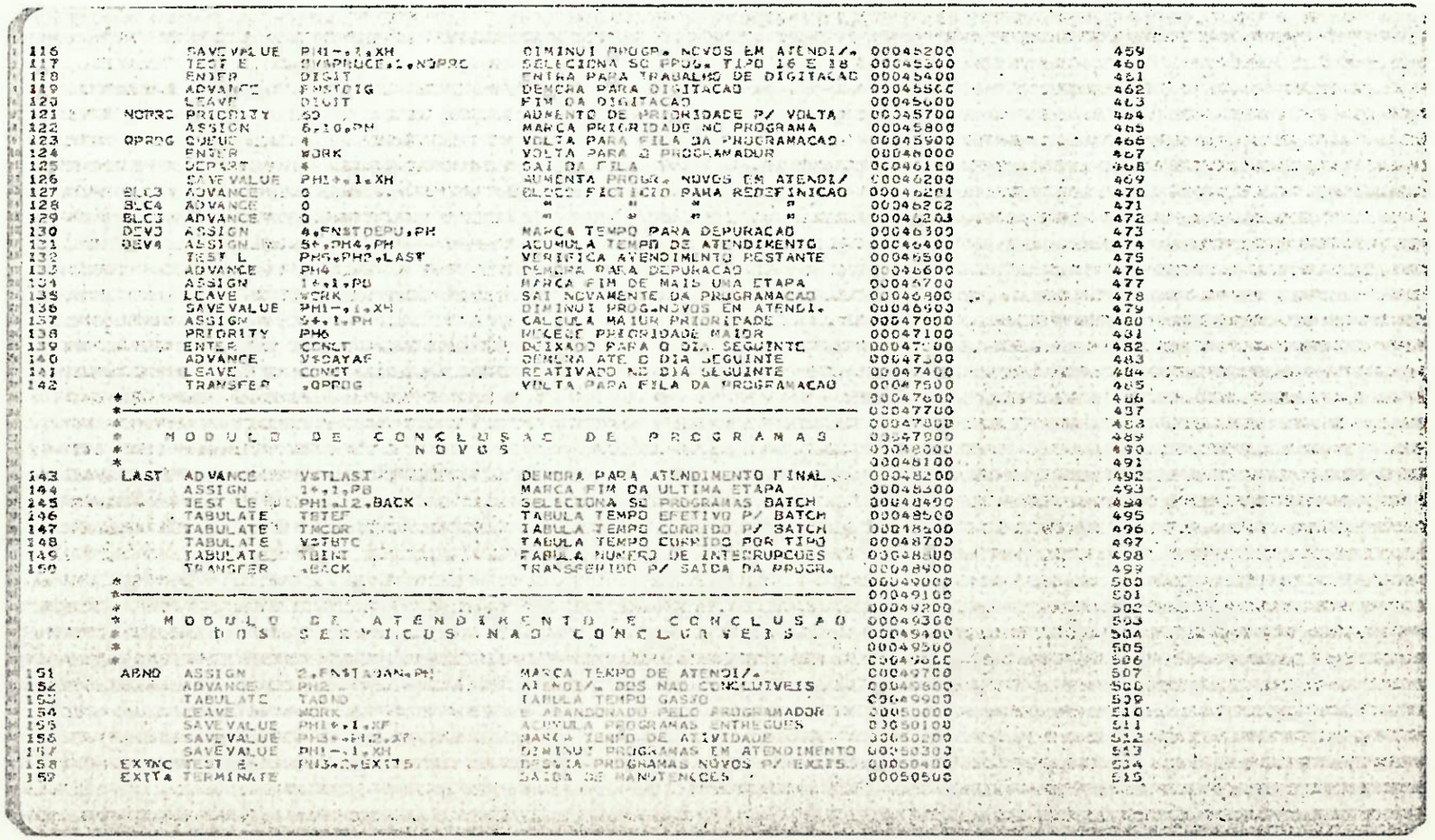


```
                *-----------------------------------------------------------------------
                *
                *  G E R A D O R   D E   P R O G R A M A S   T I P O   1 8
                *
47                GEN18 GENERATE   3964,FNSEXP,,,,9PH,9PF,9PB
48                      ASSIGN     1,18,PH
49                      ASSIGN     3,1,PH
50                RCP18 QUEUE      1
51                      TRANSFER   ,WAIT
                *
                *-----------------------------------------------------------------------
                *
                *  G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 1
                *
55                GEN11 GENERATE   168,FNTEXP,,,,9PF,9PH
56                      ASSIGN     1,11,PH
57                      ASSIGN     3,2,PH
58                      TRANSFER   ,QUEM
                *
                *-----------------------------------------------------------------------
                *
                *  G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 3
                *
60                GEN13 GENERATE   132,FNSEXP,,,,9PF,9PH
61                      ASSIGN     1,13,PH
62                      ASSIGN     3,2,PH
63                      TRANSFER   ,QUEM
                *
                *-----------------------------------------------------------------------
                *
                *  G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 5
                *
65                GEN15 GENERATE   217,FNSEXP,,,,9PF,9PH
66                      ASSIGN     1,15,PH
67                      ASSIGN     3,2,PH
68                      TRANSFER   ,QUEM
                *
                *-----------------------------------------------------------------------
                *
                *  G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 7
                *
70                GEN17 GENERATE   56,FNSEXP,,,,9PF,9PH
71                      ASSIGN     1,17,PH
72                      ASSIGN     3,2,PH
73                      TRANSFER   ,QUEM
                *
                *-----------------------------------------------------------------------
                *
                *  G E R A D O R   D E   M A N U T E N C O E S   T I P O   1 9
                *
75                GEN19 GENERATE   63,FNSEXP,,,,9PF,9PH
76                      ASSIGN     1,19,PH
77                      ASSIGN     3,2,PH
78                      TRANSFER   ,QUEM                    TRANSFERE P/ FILA DE MANUTENCOES
                *
79                QUEM  TABULATE   TMAT                    TABULA CIST. MANUT. POR TIPO
80                      TRANSFER   .40,,RECM                DESVIA 40% PARA RECM

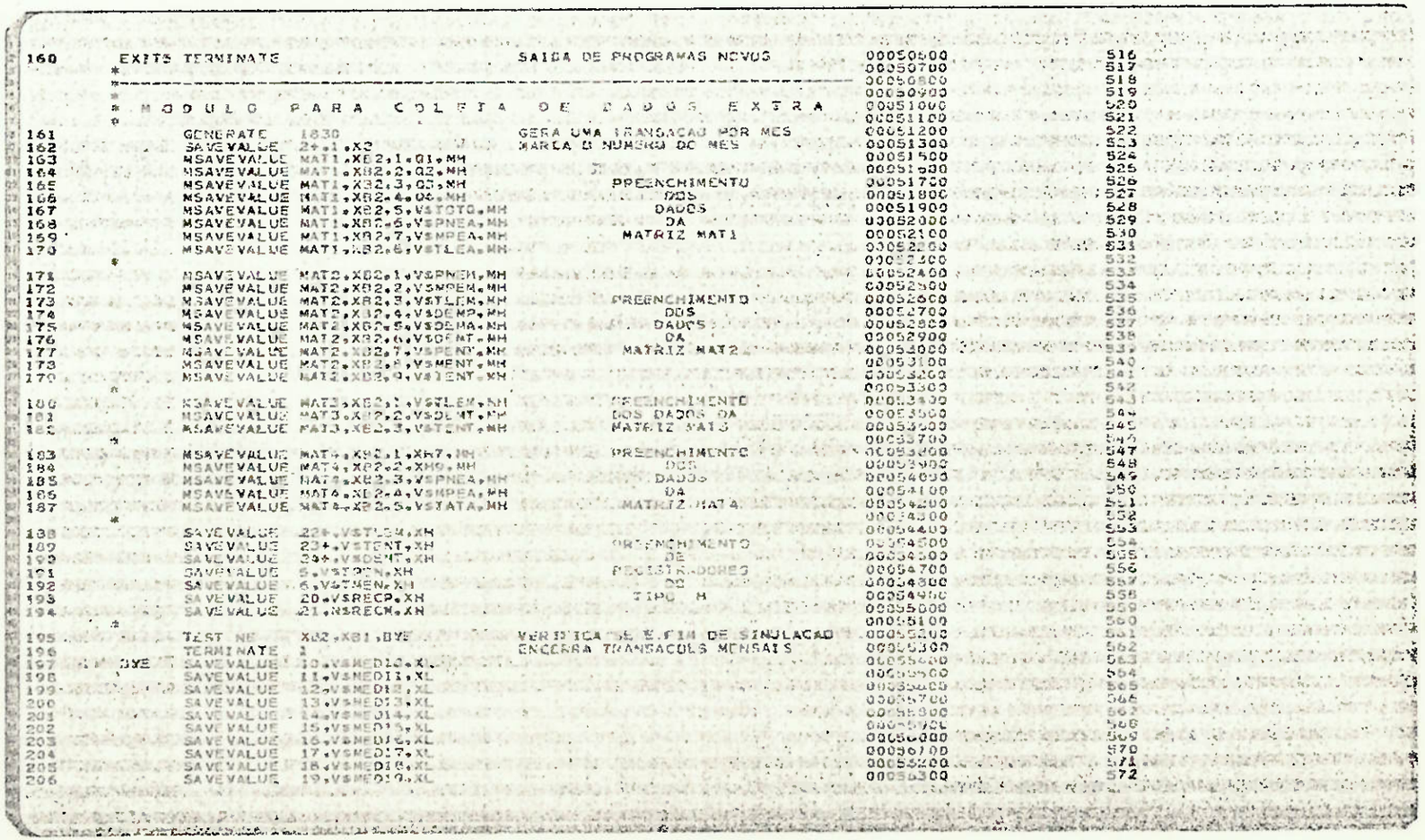
00034100                                                                              345
00034200                                                                              346
00034300                                                                              347
00034400                                                                              348
00034500                                                                              349
00034600                                                                              350
00034700                                                                              351
00034800                                                                              352
00034900                                                                              353
00035000                                                                              354
00035100                                                                              355
00035200                                                                              356
00035300                                                                              357
00035400                                                                              358
00035500                                                                              359
00035600                                                                              360
00035700                                                                              361
00035800                                                                              362
00035900                                                                              363
00036000                                                                              364
00036100                                                                              365
00036200                                                                              366
00036300                                                                              367
00036400                                                                              368
00036500                                                                              369
00036600                                                                              370
00036700                                                                              371
00036800                                                                              372
00036900                                                                              373
00037000                                                                              374
00037100                                                                              375
00037200                                                                              376
00037300                                                                              377
00037400                                                                              378
00037500                                                                              379
00037600                                                                              380
00037700                                                                              381
00037800                                                                              382
00037900                                                                              383
00038000                                                                              384
00038100                                                                              385
00038200                                                                              386
00038300                                                                              387
00038400                                                                              388
00038500                                                                              389
00038600                                                                              390
00038700                                                                              391
00038800                                                                              392
00038900                                                                              393
00039000                                                                              394*
00039100                                                                              395
00039200                                                                              396
00039300                                                                              397
00039400                                                                              398
00039500                                                                              399
00039600                                                                              400
00039700                                                                              401
```



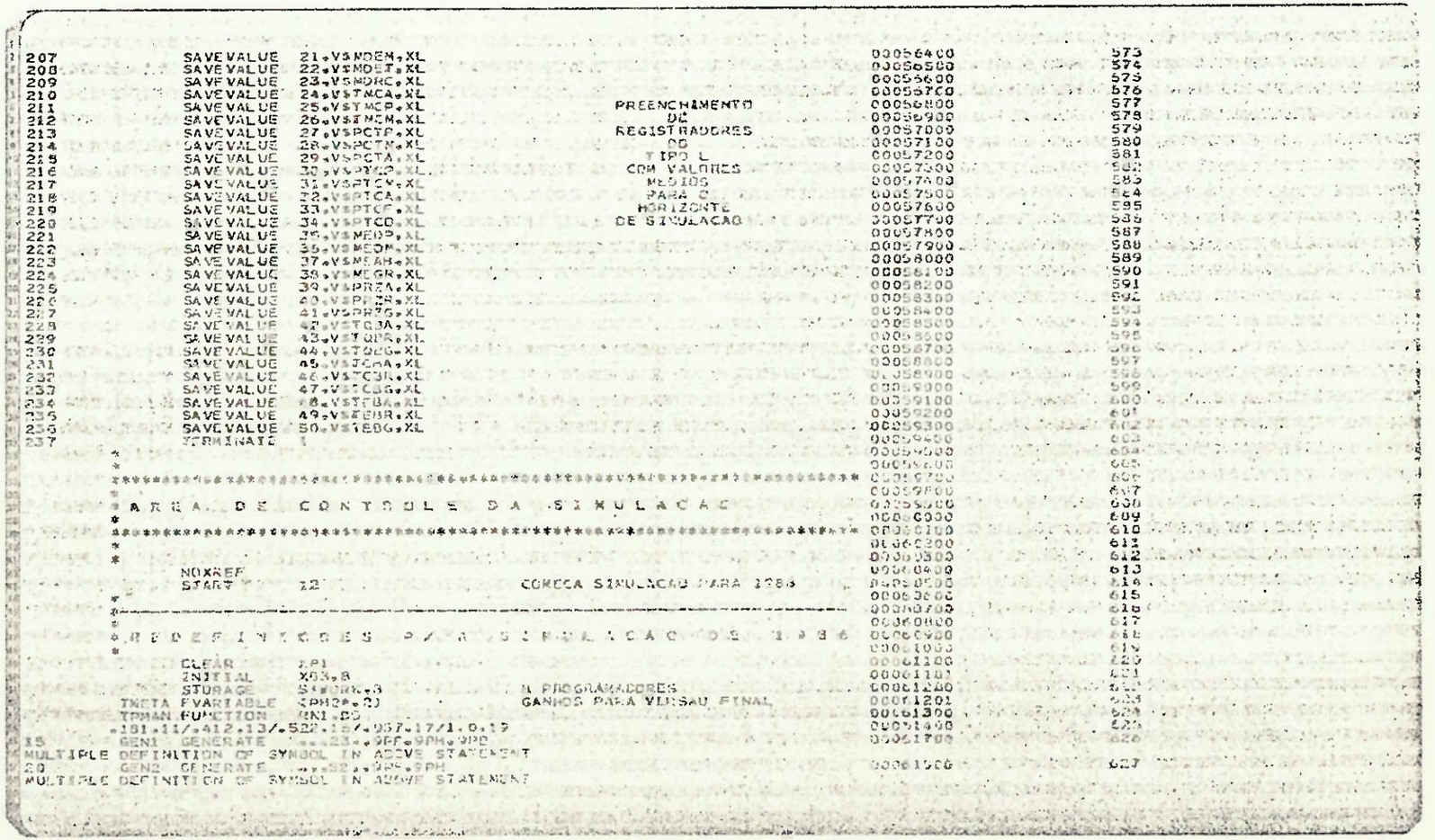
```
                PRIORITY    5                   ASSINAL PRIORIDADE
RECM            QUEUE       2                   ENTRA NA FILA DE MANUTENCOES
*
*-------------------------------------------------------------------
*
*   M O D U L O   D E   A T E N D I M E N T O   P R I N C I P A L
*
*
MAST    QUEUE       MAINQ                   ENTRA NA FILA DA PROGRAMACAO
        QUEUE       PH1                     ENTRA NA FILA POR TIPO
        TEST E      BV$ADPRG+0,NOPRG        SELECIONA SE PROGRAMAS
        TABULATE    TBPRG                   TABULA ATIVIDADES PRODUTIVAS
        TEST E      V$NEW,0,NOPRG           SELECIONE SE PROG. NOVOS
        TABULATE    TBNEW                   TABULA DIFER. DE PROGRAMAS NOVOS
NOPRG   TABULATE    TSPAR                   TABULA TODAS AS ATIVIDADES
VACAN   ENTER       WORK                    COMECA A SER TRABALHADO
        DEPART      PH1                     SAI DA FILA POR TIPO
        DEPART      MAINQ                   SAI DA FILA DA PROGRAMACAO
        DEPART      PH3                     SAI DA FILA
        SAVEVALUE   PH1+,1,XH               MARCA ENTRADA DE ATIVIDADE
        MARK        TPR                     MARCA INICIO DO TRABALHO
        TEST NE     PH3,3,NGEN              SEPARA OUTRAS ATIVIDADES
        TEST NE     PH3,5,ASGN              SEPARA FERIAS
        TRANSFER    .05,,ABND               5 % ENTRAVA DE VOLTA P/ ANALISE
ASGN    ASSIGN      2,FN$PH1,PH             MARCA TEMPO TOTAL DE ATENDIM.
BLK1    TEST E      BV$META,1,NBAT2         SELECIONA SE BATCHES
BLK2    ASSIGN      2,V$TPLA,PH             MARCA TEMPO ESTIMADO P/ META
NBAT2   TEST NE     V$NEW,0,CYCLE           DESVIA PROGR. NOVOS P/ DESENV.
        ADVANCE     PH2                     E ATENDIDO POR UM PROGRAMADOR
BACK    TABULATE    PH1                     TABELA TEMPO DE ATENDIM.
        LEAVE       WORK                    E ENTREGUE PELO PROGRAMADOR
*
*-------------------------------------------------------------------
*
*   M O D U L O   D E   F I N A L I Z A C A O   P R I N C I P A L
*
*
        SAVEVALUE   PH3+,PH2,XF             ACUMULA TOTAL DE HORAS P. TIPO
        SAVEVALUE   PH1-,1,XH               DIMINUI ATIVIDADES EM ATENDI.
        SAVEVALUE   PH1+,1,XF               ACUMULA PROGRAMAS ENTREGUES
        TEST NE     PH3,5,EXIT6             DESVIA PARA FIM DE FERIAS
        TEST NE     PH3,3,EXIT3             DESVIA OUTRAS ATIVIDADES
        TEST NE     PH3,2,EXIT2             DESVIA MANUTENCOES
EXIT1   TERMINATE                           SAIDA DE PROGRAMAS NOVOS
EXIT2   TERMINATE                           SAIDA DE MANUTENCOES
EXIT3   TERMINATE                           SAIDA DE OUTRAS ATIVIDADES
EXIT6   TERMINATE                           FIM DE FERIAS
*
*-------------------------------------------------------------------
*
*   M O D U L O   P/   D E S E N V .   D E   P R O G R A M A S   N O V O S
*
*
CYCLE   ADVANCE     0                       BLOCO FICTICIO PARA REPETIR
BLC1    ADVANCE     0                       "   "   "   "   "
BLC2    ADVANCE     0                       "   "   "   "   "
DEV2    ASSIGN      5,V$TPIN,PH             MARCA TEMPO INICIAL P/ DESENV.
DEV1    ADVANCE     PH5                     TRABALHO INICIAL DO PROGRAMADOR
        LEAVE       WORK                    FIM DO DESENVOLVIMENTO BASICO
        ASSIGN      1,1,PB                  MARCA A PRIMEIRA INTERRUPCAO
```



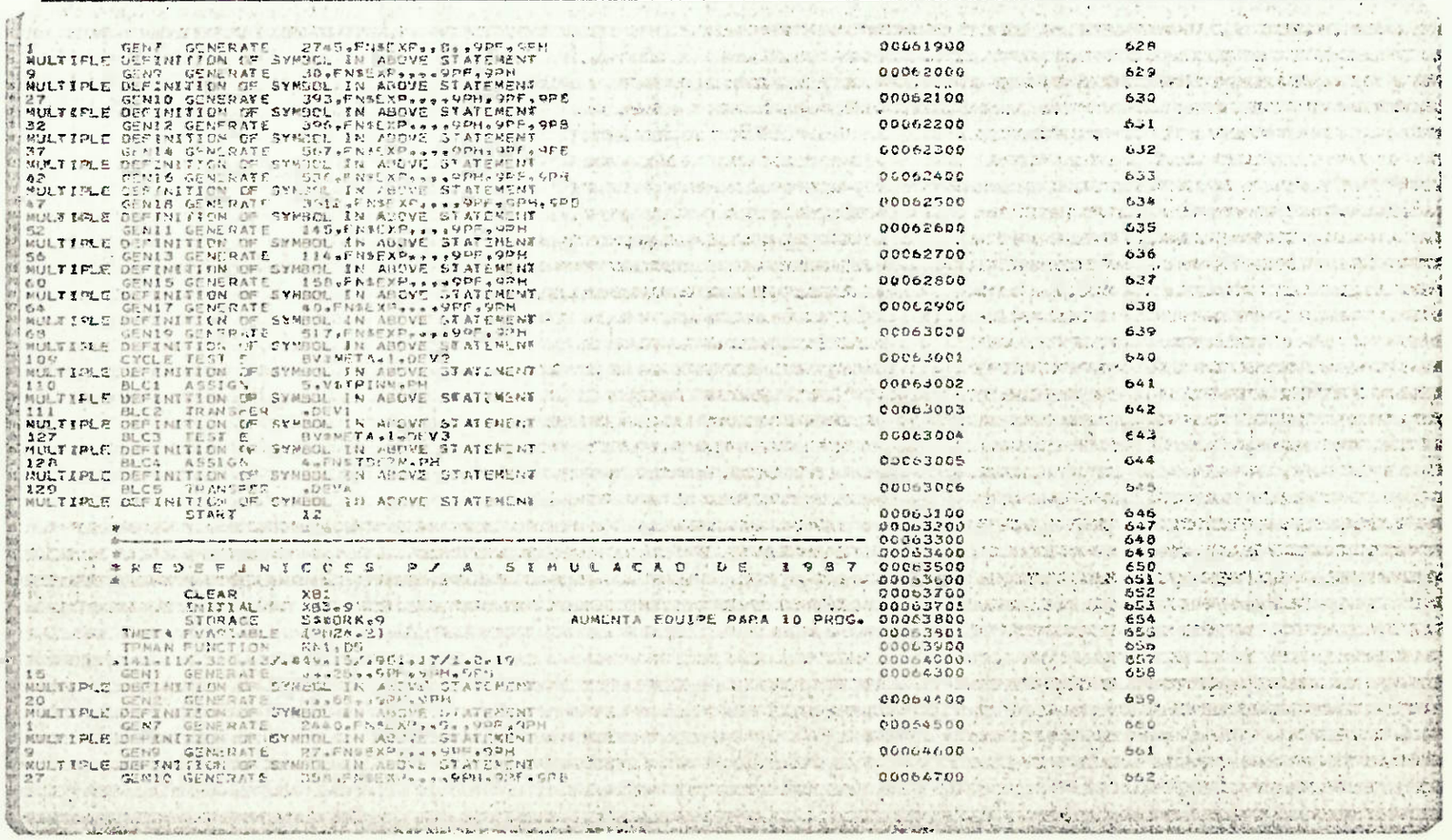
```
116                SAVEVALUE  PH1-,1,XH          DIMINUI PROGR. NOVOS EM ATENDI/.  00045200                459
117                TEST E     V$APROCE,1,NOPRC   SELECIONA SO PROG. TIPO 16 E 18    00045300                460
118                ENTER      DIGIT              ENTRA PARA TRABALHO DE DIGITACAO  00045400                461
119                ADVANCE    FNSTDIG            DEMORA PARA DIGITACAO             00045500                462
120                LEAVE      DIGIT              FIM DA DIGITACAO                  00045600                463
121          NOPRC PRIORITY   50                 AUMENTO DE PRIORIDADE P/ VOLTA    00045700                464
122                ASSIGN     6,10,PH            MARCA PRIORIDADE NO PROGRAMA      00045800                465
123          QPROG QUEUE      4                  VOLTA PARA FILA DA PROGRAMACAO    00045900                466
124                ENTER      WORK               VOLTA PARA O PROGRAMADOR          00046000                467
125                DEPART     4                  SAI DA FILA                       00046100                468
126                SAVEVALUE  PH1+,1,XH          AUMENTA PROGR. NOVOS EM ATENDI/   00046200                469
127          BLC3  ADVANCE    0                  BLOCO FICTICIO PARA REDEFINICAO   00046201                470
128          BLC4  ADVANCE    0                  "  "  "  "                       00046202                471
129          BLC5  ADVANCE    0                  "  "  "  "                       00046203                472
130          DEV3  ASSIGN     4,FNSTDEPU,PH      MARCA TEMPO PARA DEPURACAO        00046300                473
131          DEV4  ASSIGN     5+,PH4,PH          ACUMULA TEMPO DE ATENDIMENTO      00046400                474
132                TEST L     PH5,PH2,LAST       VERIFICA ATENDIMENTO RESTANTE     00046500                475
133                ADVANCE    PH4                DEMORA PARA DEPURACAO             00046600                476
134                ASSIGN     1+,1,P8            MARCA FIM DE MAIS UMA ETAPA       00046700                477
135                LEAVE      WORK               SAI NOVAMENTE DA PROGRAMACAO      00046800                478
136                SAVEVALUE  PH1-,1,XH          DIMINUI PROG.NOVOS EM ATENDI.     00046900                479
137                ASSIGN     5+,1,PH            CALCULA MAIOR PRIORIDADE          00047000                480
138                PRIORITY   PH6                MODELO PRIORIDADE MAIOR           00047100                481
139                ENTER      CONCT              DEIXADO PARA O DIA SEGUINTE       00047200                482
140                ADVANCE    VSDAYAF            DEMORA ATE O DIA SEGUINTE         00047300                483
141                LEAVE      CONCT              REATIVADO NO DIA SEGUINTE         00047400                484
142                TRANSFER   ,QPROG             VOLTA PARA FILA DA PROGRAMACAO    00047500                485
                   *                                                               00047600                486
                   *-------------------------------------------------------------  00047700                487
                   *                                                               00047800                488
                   *    MODULO DE CONCLUSAO DE PROGRAMAS                           00047900                489
                   *          NOVOS                                                00048000                490
                   *                                                               00048100                491
143          LAST  ADVANCE    VSTLAST            DEMORA PARA ATENDIMENTO FINAL     00048200                492
144                ASSIGN     1+,1,P8            MARCA FIM DA ULTIMA ETAPA         00048300                493
145                TEST LE    PH1,12,BACK        SELECIONA SE PROGRAMAS BATCH      00048400                494
146                TABULATE   TBTEF              TABULA TEMPO EFETIVO P/ BATCH     00048500                495
147                TABULATE   TMCOR              TABULA TEMPO CORRIDO P/ BATCH     00048600                496
148                TABULATE   V$TBTC             TABULA TEMPO CORRIDO POR TIPO     00048700                497
149                TABULATE   TBINT             TABULA NUMERO DE INTERRUPCOES     00048800                498
150                TRANSFER   ,BACK             TRANSFERIDO P/ SAIDA DA PROGR.    00048900                499
                   *                                                               00049000                500
                   *-------------------------------------------------------------  00049100                501
                   *                                                               00049200                502
                   *    MODULO DE ATENDIMENTO E CONCLUSAO                          00049300                503
                   *       DOS SERVICOS NAO CONCLUSIVEIS                           00049400                504
                   *                                                               00049500                505
                   *                                                               00049600                506
151          ABND  ASSIGN     2,FNSTABAN,PH      MARCA TEMPO DE ATENDI/.           00049700                507
152                ADVANCE    PH2                ATENDI/. DOS NAO CONCLUSIVEIS     00049800                508
153                TABULATE   TABND              TABULA TEMPO GASTO                00049900                509
154                LEAVE      WORK               E ABANDONADO PELO PROGRAMADOR     00050000                510
155                SAVEVALUE  PH1+,1,XF          ACUMULA PROGRAMAS ENTREGUES       00050100                511
156                SAVEVALUE  PH3+,PH2,XF        MARCA TEMPO DE ATIVIDADE          00050200                512
157                SAVEVALUE  PH1-,1,XH          DIMINUI PROGRAMAS EM ATENDIMENTO  00050300                513
158          EXTNC TEST E     PH3,2,EXITS        DESVIA PROGRAMAS NOVOS P/ EXITS   00050400                514
159          EXITS TERMINATE                     SAIDA DE MANUTENCOES              00050500                515
```



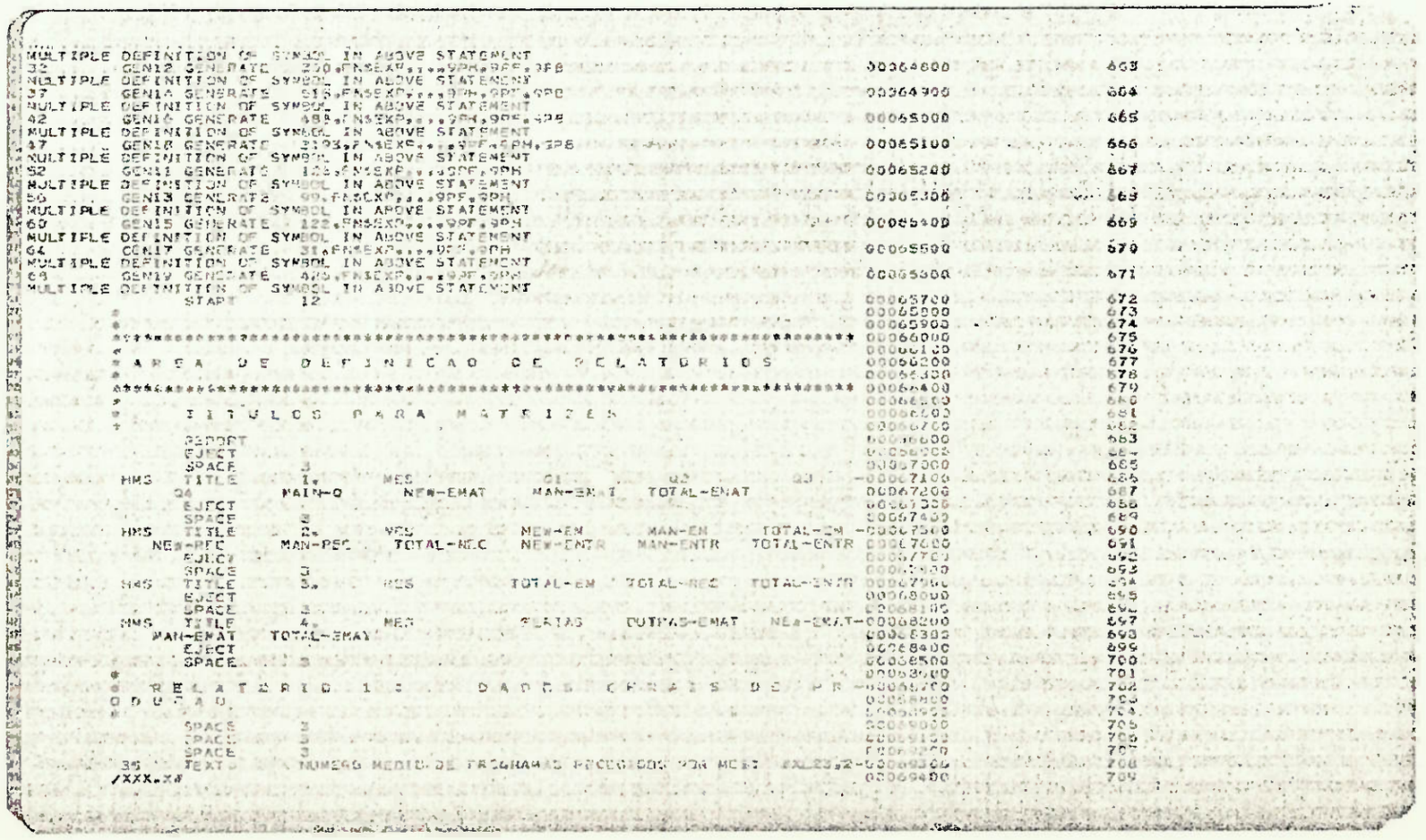
```
160      EXITS TERMINATE                          SAIDA DE PROGRAMAS NOVOS
         *
         *-------------------------------------------------------------
         *
         * MODULO PARA COLETA DE DADOS EXTRA
         *
161              GENERATE  1830                   GERA UMA TRANSACAO POR MES
162              SAVEVALUE 2+,1,X2                MARCA O NUMERO DO MES
163              MSAVEVALUE MAT1,XB2,1,01,MH
164              MSAVEVALUE MAT1,XB2,2,02,MH
165              MSAVEVALUE MAT1,XB2,3,03,MH          PREENCHIMENTO
166              MSAVEVALUE MAT1,XB2,4,04,MH             DOS
167              MSAVEVALUE MAT1,XB2,5,VSTOTG,MH       DADOS
168              MSAVEVALUE MAT1,XB2,6,VSPNEA,MH         DA
169              MSAVEVALUE MAT1,XB2,7,VSMPEA,MH     MATRIZ MAT1
170              MSAVEVALUE MAT1,XB2,8,VSTLEA,MH
         *
171              MSAVEVALUE MAT2,XB2,1,VSPNEM,MH
172              MSAVEVALUE MAT2,XB2,2,VSMPEM,MH
173              MSAVEVALUE MAT2,XB2,3,VSTLEM,MH      PREENCHIMENTO
174              MSAVEVALUE MAT2,XB2,4,VSDEM2,MH         DOS
175              MSAVEVALUE MAT2,XB2,5,VSDEMA,MH       DADOS
176              MSAVEVALUE MAT2,XB2,6,VSDEMT,MH         DA
177              MSAVEVALUE MAT2,XB2,7,VSPEND,MH     MATRIZ MAT2
178              MSAVEVALUE MAT2,XB2,8,VSMENT,MH
179              MSAVEVALUE MAT3,XB2,9,VSTENT,MH
         *
180              MSAVEVALUE MAT3,XB2,1,VSTLEM,MH      PREENCHIMENTO
181              MSAVEVALUE MAT3,XB2,2,VSDEMT,MH      DOS DADOS DA
182              MSAVEVALUE MAT3,XB2,3,VSTENT,MH      MATRIZ MAT3
         *
183              MSAVEVALUE MAT4,XB2,1,XH7,MH         PREENCHIMENTO
184              MSAVEVALUE MAT4,XB2,2,XH9,MH            DOS
185              MSAVEVALUE MAT4,XB2,3,VSPNEA,MH       DADOS
186              MSAVEVALUE MAT4,XB2,4,VSMPEA,MH         DA
187              MSAVEVALUE MAT4,XB2,5,VSTATA,MH      MATRIZ MAT4
         *
188              SAVEVALUE  22+,VSTLEM,XH
189              SAVEVALUE  23+,VSTENT,XH             ORIENCHIMENTO
190              SAVEVALUE  24+,VSDENT,XH                DE
191              SAVEVALUE  5,VSTOEN,XH               REGISTRADORES
192              SAVEVALUE  6,VSTMEN,XH                  DO
193              SAVEVALUE  20,VSRECP,XH                TIPO H
194              SAVEVALUE  21,NSRECM,XH
         *
195              TEST NE    X82,X81,BYE               VERIFICA SE E FIM DE SIMULACAO
196              TERMINATE  1                         ENCERRA TRANSACOES MENSAIS
197      BYE     SAVEVALUE  10,VSMED10,XL
198              SAVEVALUE  11,VSMED11,XL
199              SAVEVALUE  12,VSMED12,XL
200              SAVEVALUE  13,VSMED13,XL
201              SAVEVALUE  14,VSMED14,XL
202              SAVEVALUE  15,VSMED15,XL
203              SAVEVALUE  16,VSMED16,XL
204              SAVEVALUE  17,VSMED17,XL
205              SAVEVALUE  18,VSMED18,XL
206              SAVEVALUE  19,VSMED19,XL
```



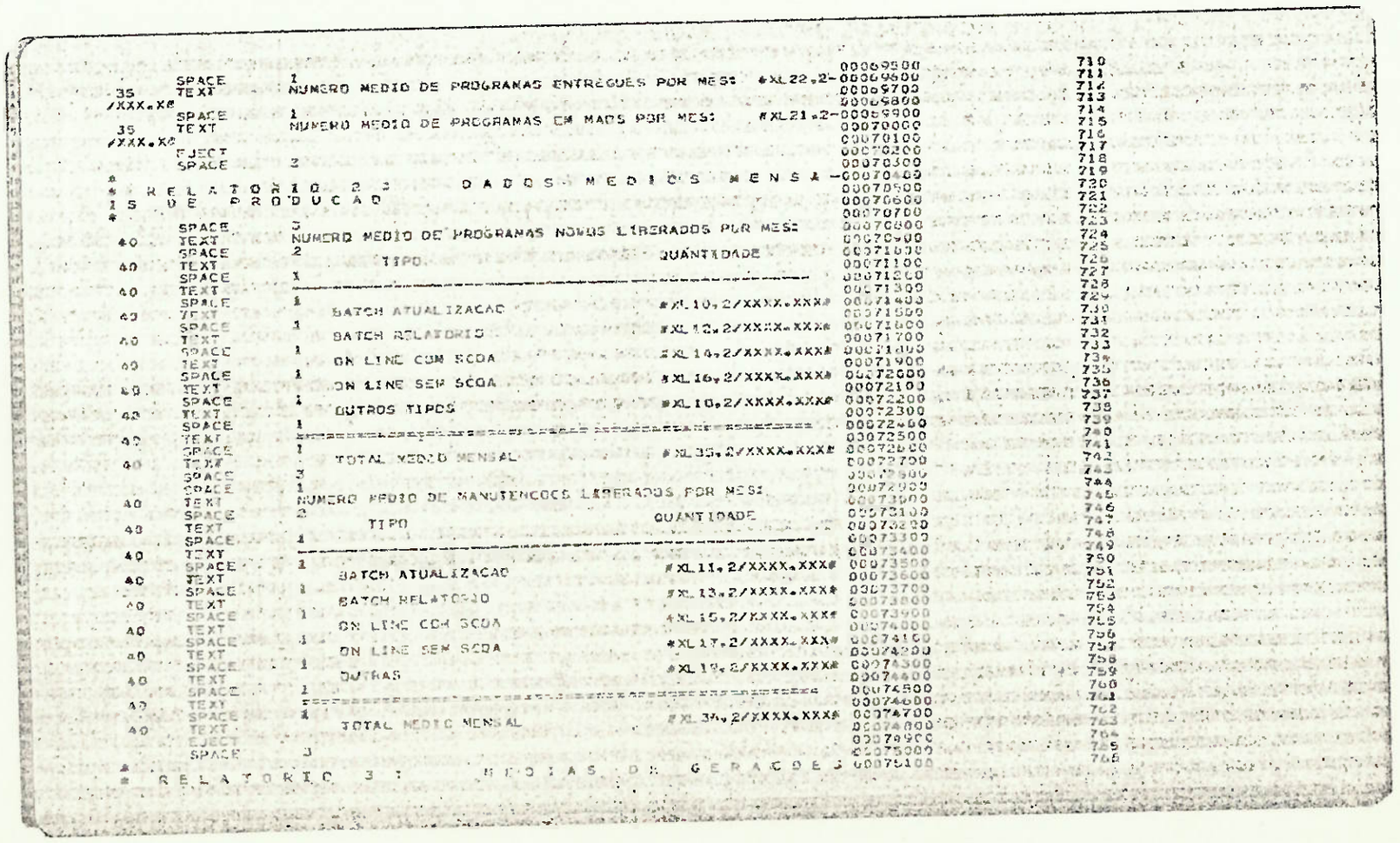
```
207                SAVEVALUE  21,VSNDEM,XL                                  00056400      573
208                SAVEVALUE  22,VSMDET,XL                                  00056500      574
209                SAVEVALUE  23,VSMDRC,XL                                  00056600      575
210                SAVEVALUE  24,VSTMCA,XL                                  00056700      576
211                SAVEVALUE  25,VSTMCP,XL                PREENCHIMENTO     00056800      577
212                SAVEVALUE  26,VSTMCM,XL                    DE            00056900      578
213                SAVEVALUE  27,VSPCTP,XL              REGISTRADORES       00057000      579
214                SAVEVALUE  28,VSPCTM,XL                    DO            00057100      580
215                SAVEVALUE  29,VSPCTA,XL                 TIPO L           00057200      581
216                SAVEVALUE  30,VSPYLD,XL              COM VALORES         00057300      582
217                SAVEVALUE  31,VSPTCV,XL                 MEDIOS           00057400      583
218                SAVEVALUE  32,VSPTCA,XL                 PARA O           00057500      584
219                SAVEVALUE  33,VSPTCF,XL               HORIZONTE          00057600      585
220                SAVEVALUE  34,VSPTCD,XL            DE SIMULACAO          00057700      586
221                SAVEVALUE  35,VSMEDP,XL                                  00057800      587
222                SAVEVALUE  36,VSMEDM,XL                                  00057900      588
223                SAVEVALUE  37,VSMEAH,XL                                  00058000      589
224                SAVEVALUE  38,VSMEGR,XL                                  00058100      590
225                SAVEVALUE  39,VSPRZA,XL                                  00058200      591
226                SAVEVALUE  40,VSPLZR,XL                                  00058300      592
227                SAVEVALUE  41,VSPRZG,XL                                  00058400      593
228                SAVEVALUE  42,VSTC3A,XL                                  00058500      594
229                SAVEVALUE  43,VSTQPR,XL                                  00058600      595
230                SAVEVALUE  44,VSTQDG,XL                                  00058700      596
231                SAVEVALUE  45,VSTCHA,XL                                  00058800      597
232                SAVEVALUE  46,VSTC3R,XL                                  00058900      598
233                SAVEVALUE  47,VSTCDG,XL                                  00059000      599
234                SAVEVALUE  48,VSTCBA,XL                                  00059100      600
235                SAVEVALUE  49,VSTBHR,XL                                  00059200      601
236                SAVEVALUE  50,VSTEDG,XL                                  00059300      602
237                TERMINATE  1                                             00059400      603
                   *                                                        00059500      604
                   *                                                        00059600      605
                   ************************************************************** 00059700      606
                   *                                                        00059800      607
                   *  A R E A   D E   C O N T R O L E   D A   S I M U L A C A C  00059900      608
                   *                                                        00060000      609
                   ************************************************************** 00060100      610
                   *                                                        00060200      611
                   *                                                        00060300      612
                   NOXREF                                                   00060400      613
                   START      12                  COMECA SIMULACAO PARA 1986 00060500      614
                   *                                                        00060600      615
                   *-------------------------------------------------------00060700      616
                   *                                                        00060800      617
                   *  R E D E F I N I C O E S   P / A   S I M U L A C A O   D E   1 9 8 6  00060900      618
                   *                                                        00061000      619
                   CLEAR      XPI                                           00061100      620
                   INITIAL    X05,8                                         00061101      621
                   STORAGE    S(GWK),3          A PROGRAMADORES             00061200      622
                   INETA FVARIABLE  XPH2*.2D    GANHOS PARA VENSAU FINAL    00061201      623
                   TPMAN FUNCTION   GN1,DC                                  00061300      624
                   .181,11/.412,13/.522,15/.957,17/1.0,19                   00061400      625
IS       GEN1  GENERATE    ...23+,9PF+,9PH+,9PD                           00061700      626
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
20       GEN2  GENERATE    ...52+,9PF+,9PH                                00061800      627
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
```



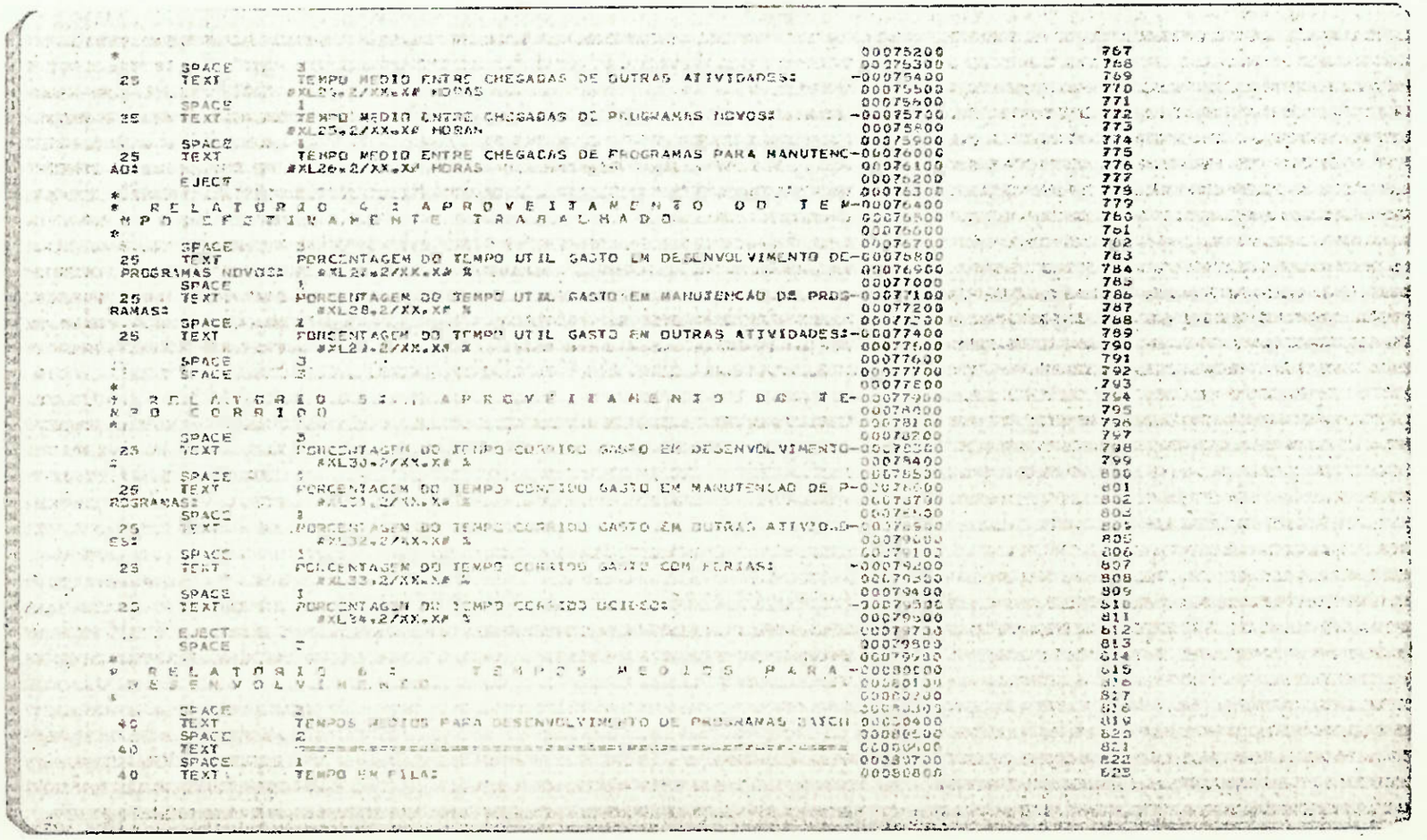
```
1          GEN7   GENERATE   2745,FNSEXP,,8,,9PF,9PH                                00061900                628
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
9          GEN8   GENERATE   30,FNSEXP,,,,9PF,9PH                                  00062000                629
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
27         GEN10  GENERATE   393,FNSEXP,,,,9PH,9PF,9PB                            00062100                630
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
32         GEN12  GENERATE   396,FNSEXP,,,,9PH,9PF,9PB                            00062200                631
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
37         GEN14  GENERATE   567,FNSEXP,,,,9PH,9PF,9PE                            00062300                632
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
42         GEN16  GENERATE   585,FNSEXP,,,,9PH,9PF,9PH                            00062400                633
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
47         GEN18  GENERATE   3611,FNSEXP,,,,9PF,9PH,9PB                           00062500                634
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
52         GEN11  GENERATE   145,FNSEXP,,,,9PF,9PH                               00062600                635
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
56         GEN13  GENERATE   114,FNSEXP,,,,9PF,9PH                               00062700                636
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
60         GEN15  GENERATE   150,FNSEXP,,,,9PF,9PH                               00062800                637
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
64         GEN17  GENERATE   40,FNSEXP,,,,9PF,9PH                                00062900                638
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
68         GEN19  GENERATE   517,FNSEXP,,,,9PF,9PH                               00063000                639
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
109        CYCLE  TEST E     BV$META+1,DEV3                                       00063001                640
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
110        BLC1   ASSIGN     5,V$TPINN,PR                                         00063002                641
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
111        BLC2   TRANSFER   ,DEV1                                                00063003                642
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
127        BLC3   TEST E     BV$META+1,DEV3                                       00063004                643
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
128        BLC4   ASSIGN     4,FNSTDEP,PH                                         00063005                644
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
129        BLC5   TRANSFER   ,DEV4                                                00063006                645
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
           START     X2                                                           00063100                646
*          ------------------------------------------------------------------   00063200                647
*                                                                               00063300                648
*                                                                               00063400                649
*          R E D E F I N I C O E S   P / A   S I M U L A C A O   D E   1 9 8 7    00063500                650
*                                                                               00063600                651
           CLEAR     X81                                                        00063700                652
           INITIAL   X83,9                                                      00063701                653
           STORAGE   S$WORK,9                  AUMENTA EQUIPE PARA 10 PROG.     00063800                654
           THETA EVALUABLE  (PN2A,2)                                            00063801                655
           IPMAN FUNCTION   RN1,D6                                              00063900                656
           .141-11/.355-13/.849+13/.901,17/1,0,19                               00064000                657
15         GEN1   GENERATE   ,,,20,,,9PF,,9PH,9PB                               00064300                658
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
20         GEN2   GENERATE   ,,,9B,,,9PF,9PH                                    00064400                659
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
1          GEN7   GENERATE   2440,FNSEXP,,,8,,,9PF,9PH                          00064500                660
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
9          GEN8   GENERATE   27,FNSEXP,,,,,9PF,9PH                              00064600                661
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
27         GEN10  GENERATE   358,FNSEXP,,,,,9PH,9PF,9PB                         00064700                662
```



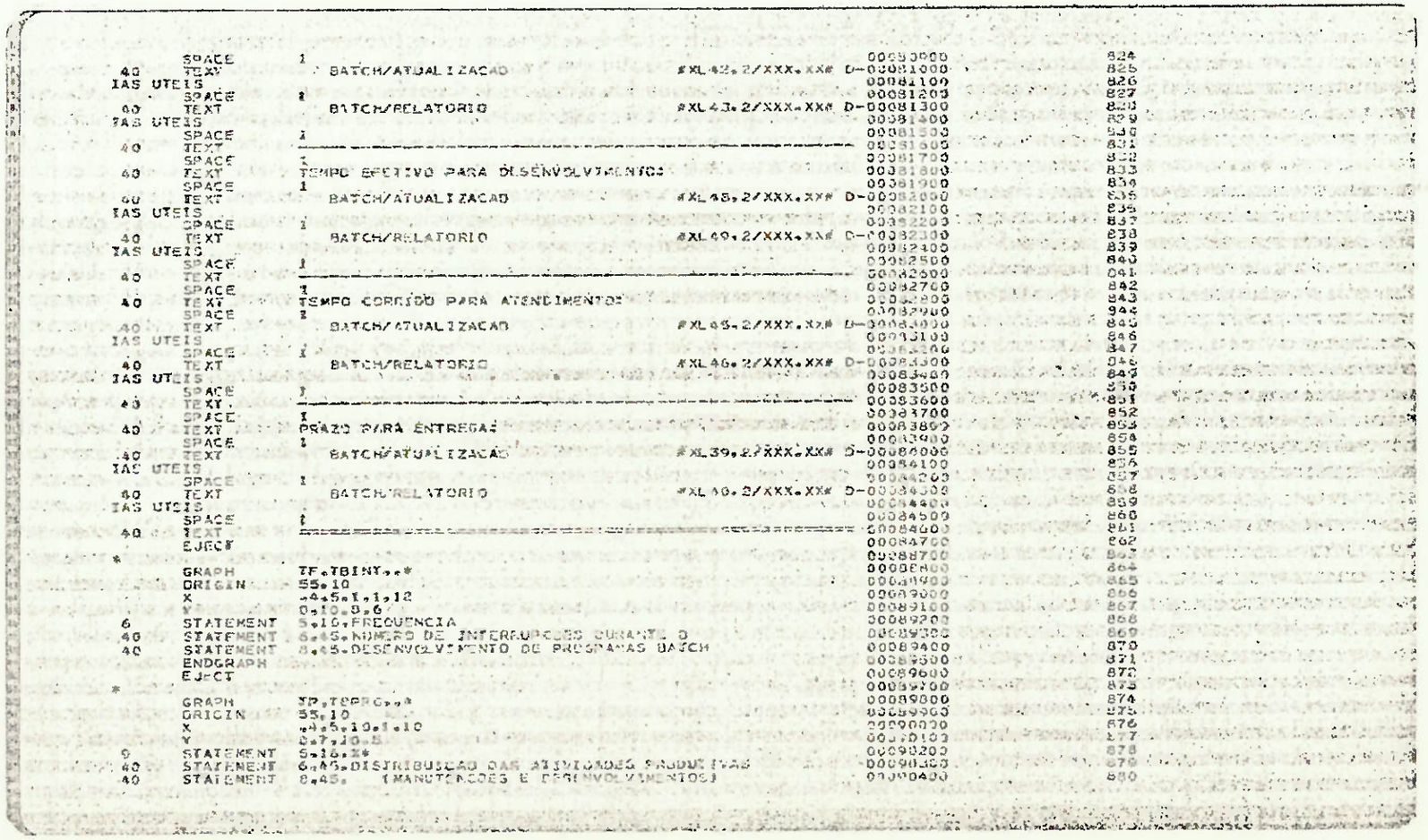
```
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
32        GEN12 GENERATE    230,FNSEXP,,,,9PH,9PF,3P8                    00064800                663
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
37        GEN14 GENERATE    313,FNSEXP,,,,9PH,9PF,9P8                    00064900                664
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
42        GEN16 GENERATE    488,FNSEXP,,,,9PH,9PF,4P8                    00065000                665
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
47        GEN18 GENERATE    2193,FNSEXP,,,,9PF,9PH,3P8                   00065100                666
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
52        GEN11 GENERATE    126,FNSEXP,,,,9PF,9PH                        00065200                667
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
56        GEN13 GENERATE    90,FNSEXP,,,,9PF,9PH                         00065300                668
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
60        GEN15 GENERATE    120,FNSEXP,,,,9PF,9PH                        00065400                669
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
64        GEN17 GENERATE    31,FNSEXP,,,,9PF,9PH                         00065500                670
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
68        GEN19 GENERATE    470,FNSEXP,,,,9PF,9PH                        00065600                671
MULTIPLE DEFINITION OF SYMBOL IN ABOVE STATEMENT
          START   12                                                     00065700                672
*                                                                        00065800                673
*                                                                        00065900                674
*************************************************************************00066000                675
*                                                                       *00066100                676
* A R E A   D E   D E F I N I C A O   D E   R E L A T O R I O S         *00066200                677
*                                                                       *00066300                678
*************************************************************************00066400                679
*                                                                        00066500                680
*     T I T U L O S   P A R A   M A T R I Z E S                          00066600                681
*                                                                        00066700                682
          REPORT                                                       00066800                683
          EJECT                                                        00066900                684
          SPACE   3                                                    00067000                685
    HMS   TITLE   1,    MES          01        02        03   -00067100                686
          Q4              MAIN-Q      NEW-EMAT    MAN-ENAT   TOTAL-EMAT  00067200                687
          EJECT                                                        00067300                688
          SPACE   3                                                    00067400                689
    HMS   TITLE   2,    YES          NEW-EN      MAN-EN     TOTAL-EN -00067500                690
       NEW-PEC    MAN-PEC    TOTAL-REC   NEW-ENTR    MAN-ENTR   TOTAL-ENTR 00067600                691
          EJECT                                                        00067700                692
          SPACE   3                                                    00067800                693
    HMS   TITLE   3,    REG          TOTAL-EM   TOTAL-REC   TOTAL-ENTR 00067900                694
          EJECT                                                        00068000                695
          SPACE   3                                                    00068100                696
    HMS   TITLE   4,    MES          SLATAS     OUTPAS-EMAT   NEW-EMAT-00068200                697
       MAN-EMAT   TOTAL-EMAT                                           00068300                698
          EJECT                                                        00068400                699
          SPACE   3                                                    00068500                700
*                                                                      00068600                701
*  R E L A T O R I O   1 :   D A D O S   G E R A I S   D E   P R -00068700                702
O D U T A O                                                            00068800                703
*                                                                      00068900                704
          SPACE   3                                                    00069000                705
          SPACE   3                                                    00069100                706
          SPACE   3                                                    00069200                707
    35    TEXT    'NUMERO MEDIO DE PROGRAMAS PROCESSADOS POR MES  EXEC-00069300                708
/XXX.X#                                                                00069400                709
```



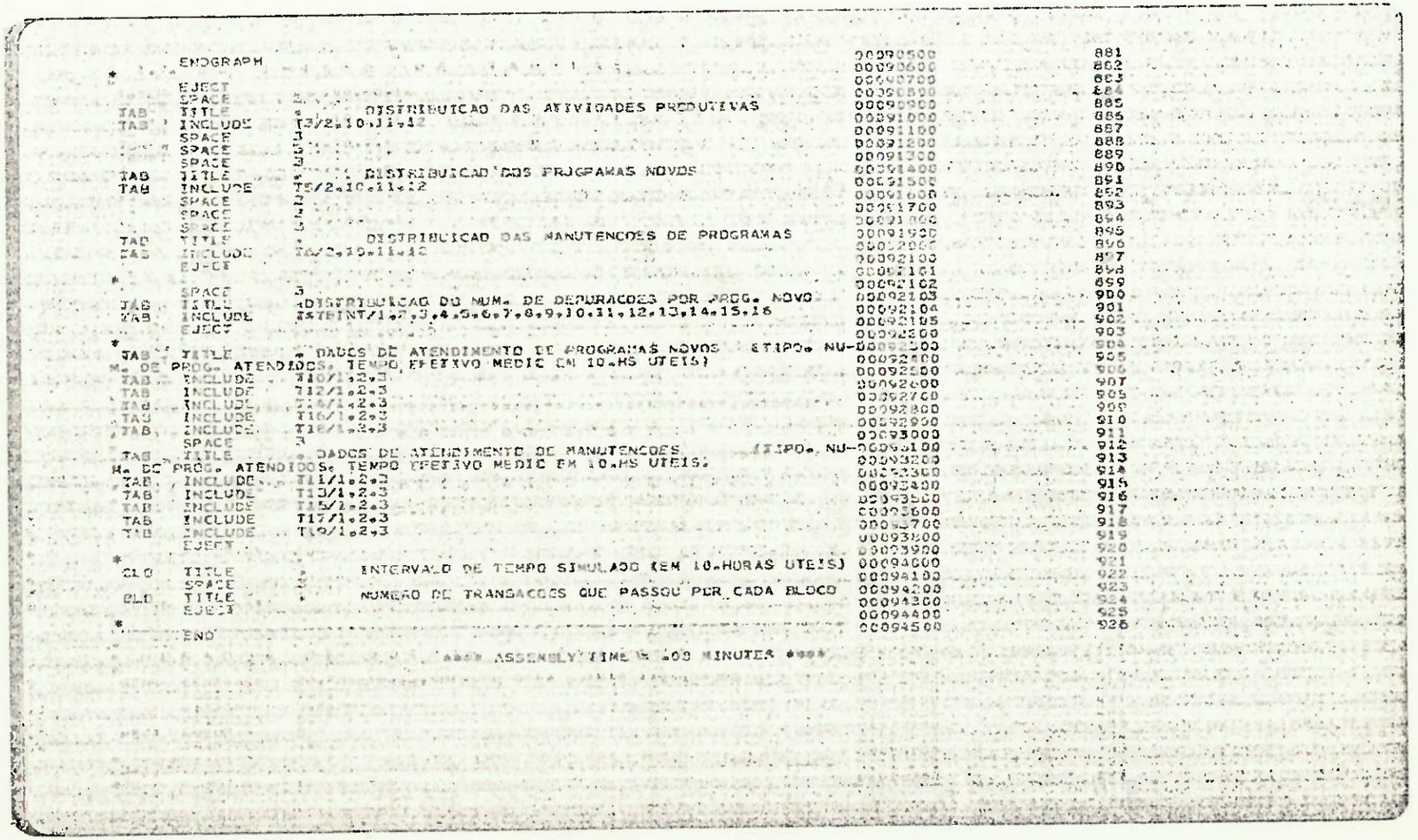
```
         SPACE   1
         TEXT    NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES:  #XL22.2-
35       TEXT
/XXX.X#
         SPACE   1
         TEXT    NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES:    #XL21.2-
35       TEXT
/XXX.X#
         EJECT
         SPACE   2
#
# RELATORIO 2.3      DADOS MEDIOS MENSAIS
1S DE PRODUCAO
#
         SPACE   3
40       TEXT    NUMERO MEDIO DE PROGRAMAS NOVOS LIBERADOS POR MES:
         SPACE   2
40       TEXT          TIPO                        QUANTIDADE
         SPACE   1
40       TEXT    ------------------------------------------------
         SPACE   1
40       TEXT       BATCH ATUALIZACAO             #XL10.2/XXXX.XXX#
         SPACE   1
40       TEXT       BATCH RELATORIO               #XL12.2/XXXX.XXX#
         SPACE   1
40       TEXT       ON LINE COM SCDA              #XL14.2/XXXX.XXX#
         SPACE   1
40       TEXT       ON LINE SEM SCDA              #XL16.2/XXXX.XXX#
         SPACE   1
40       TEXT       OUTROS TIPOS                  #XL10.2/XXXX.XXX#
         SPACE   1
40       TEXT    ================================================
         SPACE   1
40       TEXT       TOTAL MEDIO MENSAL            #XL35.2/XXXX.XXX#
         SPACE   3
         SPACE   1
40       TEXT    NUMERO MEDIO DE MANUTENCOES LIBERADOS POR MES:
         SPACE   2
40       TEXT          TIPO                        QUANTIDADE
         SPACE   1
40       TEXT    ------------------------------------------------
         SPACE   1
40       TEXT       BATCH ATUALIZACAO             #XL11.2/XXXX.XXX#
         SPACE   1
40       TEXT       BATCH RELATORIO               #XL13.2/XXXX.XXX#
         SPACE   1
40       TEXT       ON LINE COM SCDA              #XL15.2/XXXX.XXX#
         SPACE   1
40       TEXT       ON LINE SEM SCDA              #XL17.2/XXXX.XXX#
         SPACE   1
40       TEXT       OUTRAS                        #XL19.2/XXXX.XXX#
         SPACE   1
40       TEXT    ================================================
         SPACE   1
40       TEXT       TOTAL MEDIO MENSAL            #XL36.2/XXXX.XXX#
         EJECT
         SPACE   3
#
# RELATORIO 3.1      MEDIAS DE GERACOES
```



```
*
         SPACE   3                                                      00075200
         TEXT    TEMPO MEDIO ENTRE CHEGADAS DE OUTRAS ATIVIDADES:       00075300
           #XL24,2/XX,X# HORAS                                          00075400
         SPACE   1                                                      00075500
         TEXT    TEMPO MEDIO ENTRE CHEGADAS DE PROGRAMAS NOVOS:         00075600
           #XL25,2/XX,X# HORAS                                          00075700
         SPACE   1                                                      00075800
         TEXT    TEMPO MEDIO ENTRE CHEGADAS DE PROGRAMAS PARA MANUTENC- 00075900
AO:                                                                     00076000
         EJECT                                                          00076100
*                                                                       00076200
*  R E L A T O R I O   4 :   A P R O V E I T A M E N T O   D O   T E M-  00076300
*  P O   E F E T I V A M E N T E   T R A B A L H A D O                   00076400
*                                                                       00076500
         SPACE   3                                                      00076600
         TEXT    PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE- 00076700
PROGRAMAS NOVOS:                                                        00076800
           #XL27,2/XX,X# %                                              00076900
         SPACE   1                                                      00077000
         TEXT    PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROG- 00077100
RAMAS:                                                                  00077200
           #XL28,2/XX,X# %                                              00077300
         SPACE   2                                                      00077400
         TEXT    PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:- 00077500
           #XL29,2/XX,X# %                                              00077600
         SPACE   3                                                      00077700
         SPACE   3                                                      00077800
*                                                                       00077900
*  R E L A T O R I O   5 :   A P R O V E I T A M E N T O   D O   T E-  00078000
*  M P O   C O R R I D O                                                00078100
*                                                                       00078200
         SPACE   3                                                      00078300
         TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO- 00078400
           #XL30,2/XX,X# %                                              00078500
         SPACE   1                                                      00078600
         TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE P- 00078700
ROGRAMAS:                                                               00078800
           #XL31,2/XX,X# %                                              00078900
         SPACE   1                                                      00079000
         TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDA-  00079100
DES:                                                                    00079200
           #XL32,2/XX,X# %                                              00079300
         SPACE   1                                                      00079400
         TEXT    PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:         00079500
           #XL33,2/XX,X# %                                              00079600
         SPACE   1                                                      00079700
         TEXT    PORCENTAGEM DO TEMPO CORRIDO OCIOSO:                   00079800
           #XL34,2/XX,X# %                                              00079900
         EJECT                                                          00080000
         SPACE   2                                                      00080100
*                                                                       00080200
*  R E L A T O R I O   6 :   T E M P O S   M E D I O S   P A R A -      00080300
*  D E S E N V O L V I M E N T O                                        00080400
*                                                                       00080500
         SPACE   3                                                      00080600
         TEXT    TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH  00080700
         SPACE   2                                                      00080800
         TEXT    ---------- ---------- ---------- ----------            00080900
         SPACE   1                                                      00081000
         TEXT    TEMPO EM FILA:                                         00081100
```



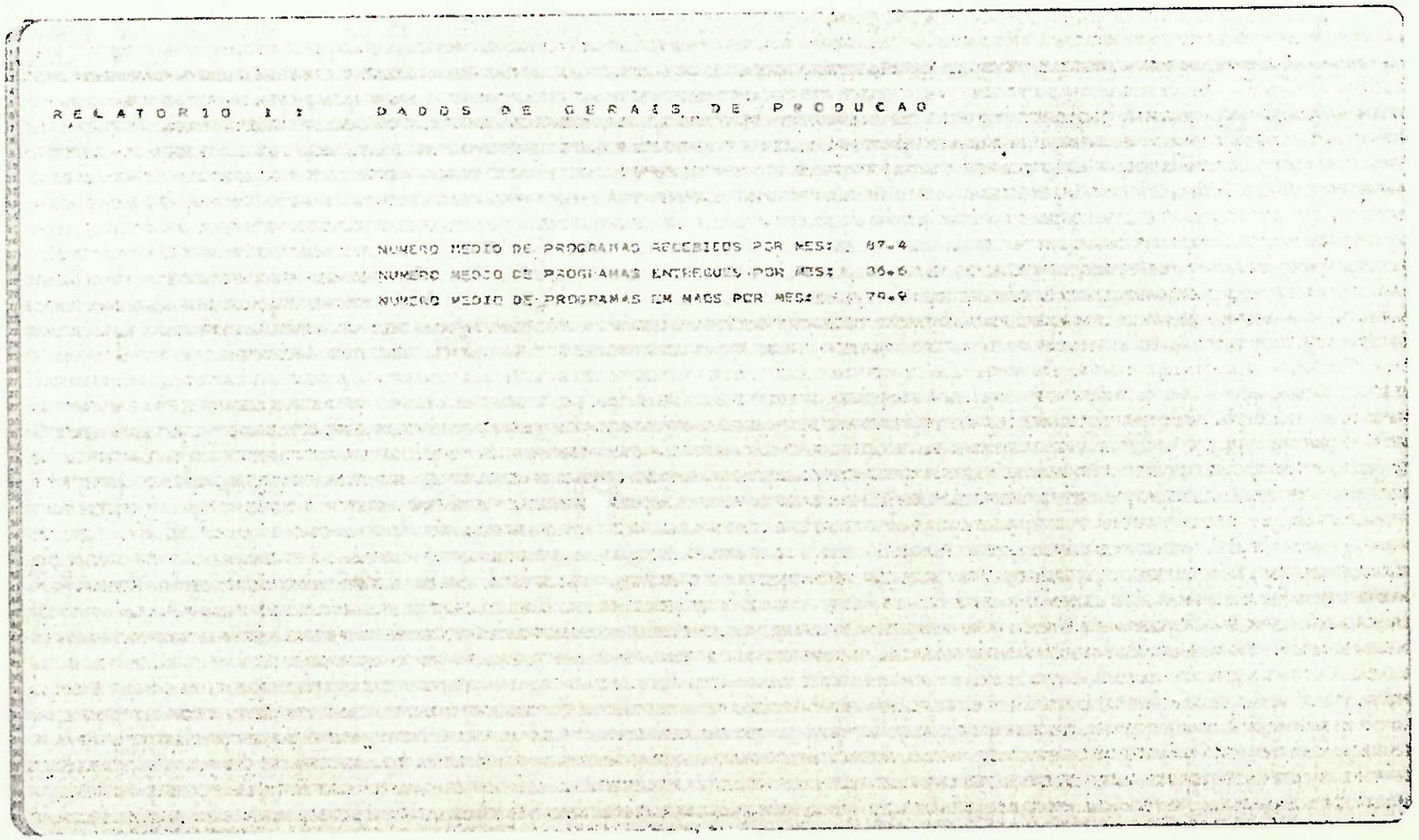
```
         SPACE   1
 40      TEXT      BATCH/ATUALIZACAO                #XL42.2/XXX.XX# D-00081000
IAS UTEIS
         SPACE   1
 40      TEXT      BATCH/RELATORIO                  #XL43.2/XXX.XX# D-00081300
IAS UTEIS
         SPACE   1
 40      TEXT    ------------------------------------------------------------
         SPACE   1
 40      TEXT    TEMPO EFETIVO PARA DESENVOLVIMENTOS:
         SPACE   1
 40      TEXT      BATCH/ATUALIZACAO                #XL48.2/XXX.XX# D-00082000
IAS UTEIS
         SPACE   1
 40      TEXT      BATCH/RELATORIO                  #XL49.2/XXX.XX# D-00082300
IAS UTEIS
         SPACE   1
 40      TEXT    ------------------------------------------------------------
         SPACE   1
 40      TEXT    TEMPO CORRIDO PARA ATENDIMENTOS
         SPACE   1
 40      TEXT      BATCH/ATUALIZACAO                #XL45.2/XXX.XX# D-00083000
IAS UTEIS
         SPACE   1
 40      TEXT      BATCH/RELATORIO                  #XL46.2/XXX.XX# D-00083300
IAS UTEIS
         SPACE   1
 40      TEXT    ------------------------------------------------------------
         SPACE   1
 40      TEXT    PRAZO PARA ENTREGAS
         SPACE   1
 40      TEXT      BATCH/ATUALIZACAO                #XL39.2/XXX.XX# D-00084000
IAS UTEIS
         SPACE   1
 40      TEXT      BATCH/RELATORIO                  #XL40.2/XXX.XX# D-00084300
IAS UTEIS
         SPACE   1
 40      TEXT    ------------------------------------------------------------
         EJECT
*
         GRAPH   TF,TBINY,,*
         ORIGIN  55,10
         X       -4,5,1,1,12
         Y       0,10,3,6
 6       STATEMENT 5,10,FREQUENCIA
 40      STATEMENT 6,45,NUMERO DE INTERRUPCOES DURANTE O
 40      STATEMENT 8,45,DESENVOLVIMENTO DE PROGRAMAS BATCH
         ENDGRAPH
         EJECT
*
         GRAPH   TP,TBPRG,,*
         ORIGIN  55,10
         X       -3,5,10,1,10
         Y       3,7,10,5
 0       STATEMENT 5,16,%
 40      STATEMENT 6,45,DISTRIBUICAO DAS ATIVIDADES PRODUTIVAS
 40      STATEMENT 8,45,    (MANUTENCOES E DESENVOLVIMENTOS)
```



```
         ENDGRAPH                                                                00000500    881
*                                                                                00000600    882
*                                                                                00000700    883
         EJECT                                                                   00000800    884
         SPACE    3                                                              00000900    885
TAB      TITLE    '     DISTRIBUICAO DAS ATIVIDADES PRODUTIVAS                   00001000    886
TAB      INCLUDE  T3/2,10,11,12                                                  00001100    887
         SPACE    3                                                              00001200    888
         SPACE    3                                                              00001300    889
         SPACE    3                                                              00001400    890
TAB      TITLE    '     DISTRIBUICAO DOS PROGRAMAS NOVOS                         00001500    891
TAB      INCLUDE  T5/2,10,11,12                                                  00001600    892
         SPACE    3                                                              00001700    893
         SPACE    3                                                              00001800    894
         SPACE    3                                                              00001900    895
TAB      TITLE    '     DISTRIBUICAO DAS MANUTENCOES DE PROGRAMAS                00002000    896
TAB      INCLUDE  T6/2,10,11,12                                                  00002100    897
         EJECT                                                                   00002101    898
*                                                                                00002102    899
         SPACE    3                                                              00002103    900
TAB      TITLE    'DISTRIBUICAO DO NUM. DE DEPURACOES POR PROG. NOVO             00002104    901
TAB      INCLUDE  T*TBINT/1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16                 00002105    902
         EJECT                                                                   00002200    903
*                                                                                00002300    904
TAB      TITLE    '  DADOS DE ATENDIMENTO DE PROGRAMAS NOVOS   &TIPO. NU-       00002400    905
M. DE PROG. ATENDIDOS, TEMPO EFETIVO MEDIO EM 10.HS UTEIS)                       00002500    906
TAB      INCLUDE  T10/1,2,3                                                      00002600    907
TAB      INCLUDE  T12/1,2,3                                                      00002700    908
TAB      INCLUDE  T14/1,2,3                                                      00002800    909
TAB      INCLUDE  T16/1,2,3                                                      00002900    910
TAB      INCLUDE  T18/1,2,3                                                      00003000    911
         SPACE    3                                                              00003100    912
TAB      TITLE    '  DADOS DE ATENDIMENTO DE MANUTENCOES      (TIPO. NU-       00003200    913
M. DE PROG. ATENDIDOS, TEMPO EFETIVO MEDIO EM 10.HS UTEIS)                       00003300    914
TAB      INCLUDE  T11/1,2,3                                                      00003400    915
TAB      INCLUDE  T13/1,2,3                                                      00003500    916
TAB      INCLUDE  T15/1,2,3                                                      00003600    917
TAB      INCLUDE  T17/1,2,3                                                      00003700    918
TAB      INCLUDE  T19/1,2,3                                                      00003800    919
         EJECT                                                                   00003900    920
*                                                                                00004000    921
CLO      TITLE    '     INTERVALO DE TEMPO SIMULADO (EM 10.HORAS UTEIS)          00004100    922
         SPACE    2                                                              00004200    923
CLO      TITLE    '     NUMERO DE TRANSACOES QUE PASSOU POR CADA BLOCO           00004300    924
         EJECT                                                                   00004400    925
*                                                                                00004500    926
         END

                    **** ASSEMBLY TIME = .008 MINUTES ****
```

ANEXO VI - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO PERÍODO DE 1985 a 1989 COM O META-COBOL



ANEXO VI.A - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1985 COM O META-COBOL(GRUPO MÉDIO DE 8 PROGRAMADORES)

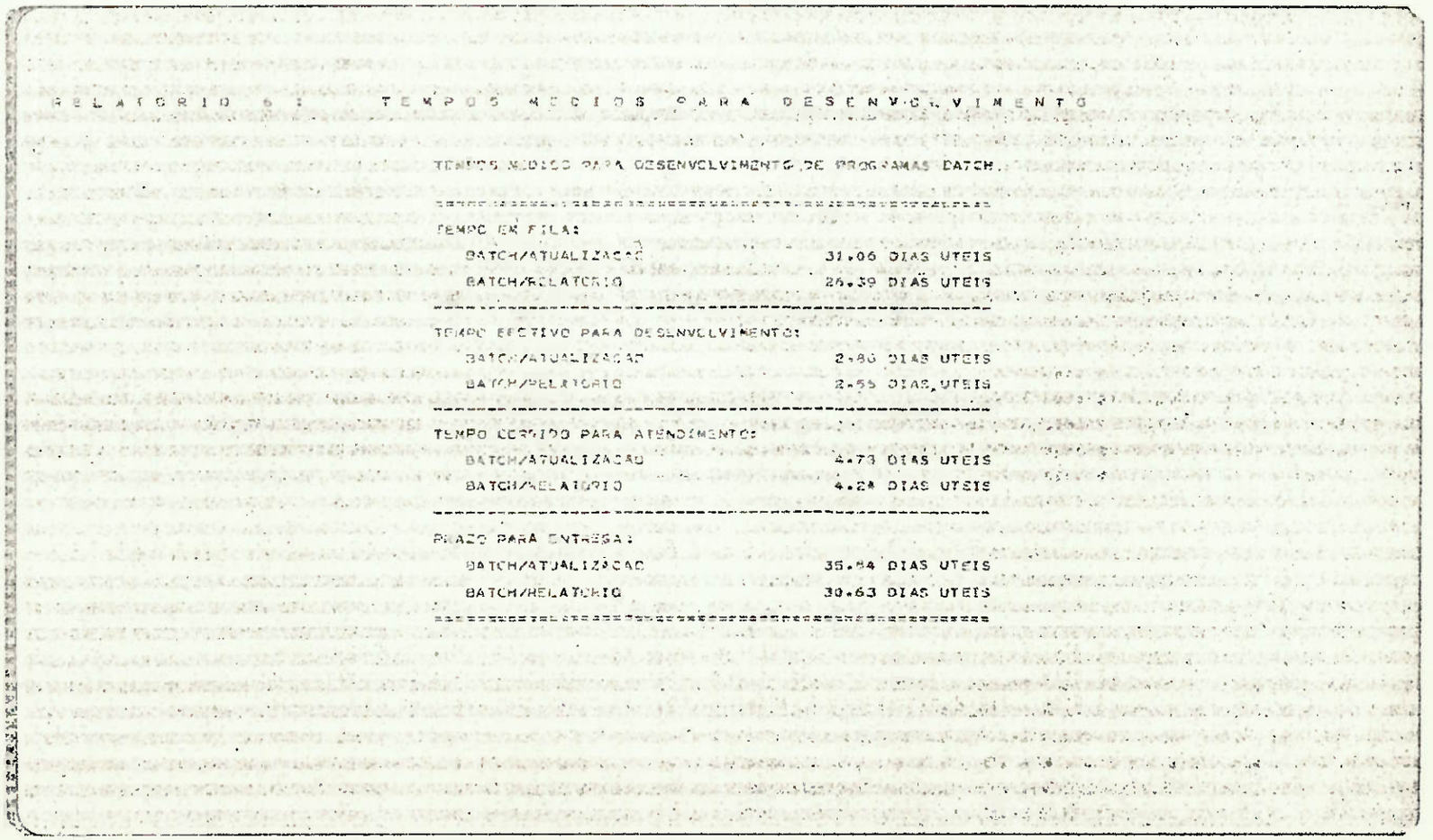
RELATORIO A : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	29.5 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	51.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	18.8 %

RELATORIO B: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	27.5 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	48.0 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	17.5 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	5.1 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	0.5 %

CONTINUAÇÃO DO ANEXO VI.A

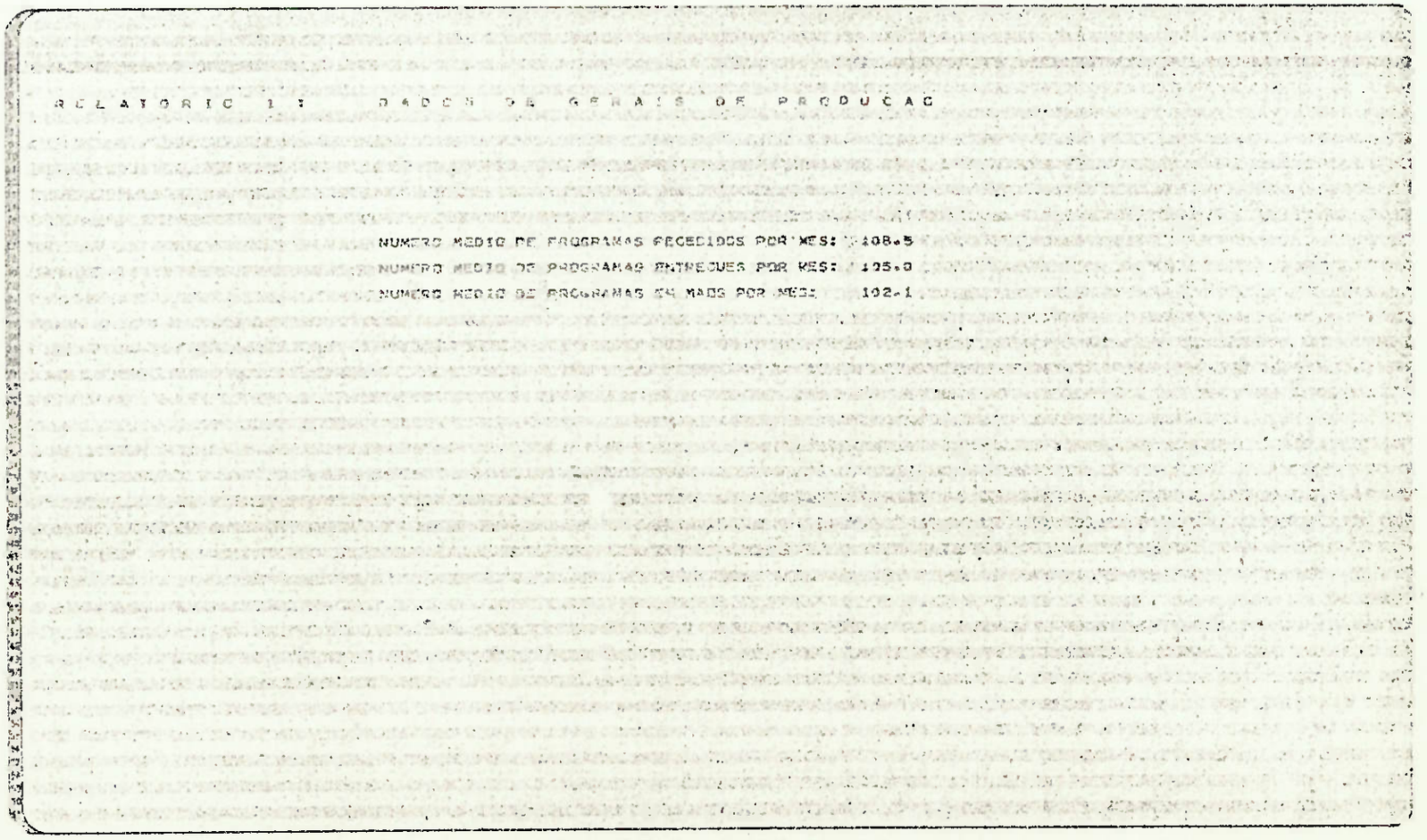


RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH.

TEMPO EM FILA:	
BATCH/ATUALIZACAO	31.06 DIAS UTEIS
BATCH/RELATORIO	26.39 DIAS UTEIS
TEMPO EFECTIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	2.80 DIAS UTEIS
BATCH/RELATORIO	2.55 DIAS UTEIS
TEMPO DECORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	4.78 DIAS UTEIS
BATCH/RELATORIO	4.24 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	35.84 DIAS UTEIS
BATCH/RELATORIO	30.63 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VI.A



RELATORIO 1 : DADOS DE GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 108.5
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 125.0
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 102.1

ANEXO VI.B - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1986 COM O META-COBOL (GRUPO MÉDIO DE 8 PROGRAMADORES)

RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	20.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	60.8 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	18.4 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	19.7 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	57.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	17.6 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	3.4 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	1.2 %

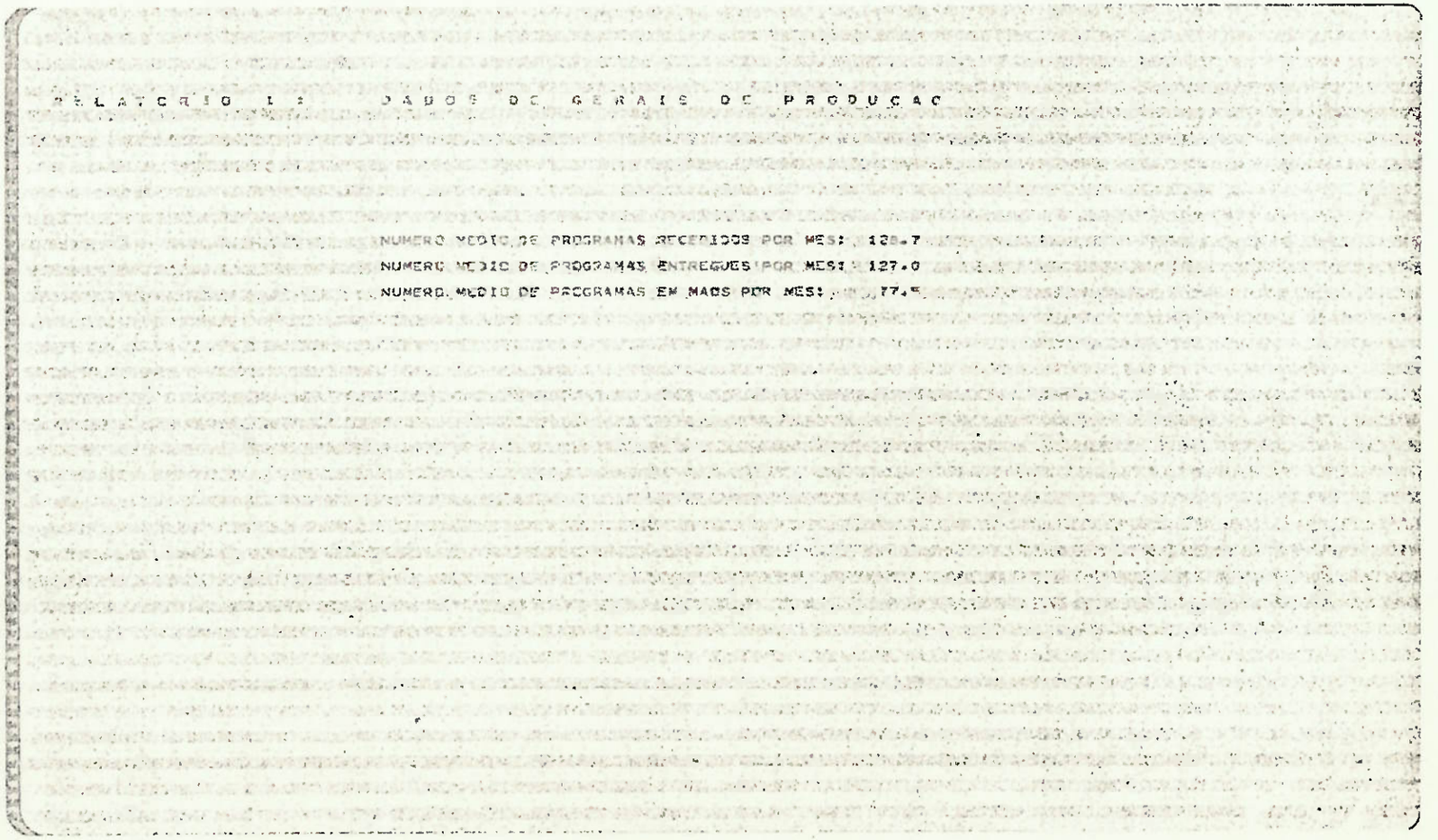
CONTINUAÇÃO DO ANEXO VI.B

RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	30.14 DIAS UTEIS
BATCH/RELATORIO	28.05 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	0.92 DIAS UTEIS
BATCH/RELATORIO	0.75 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	1.36 DIAS UTEIS
BATCH/RELATORIO	1.08 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	31.50 DIAS UTEIS
BATCH/RELATORIO	30.03 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VI.B



RELATORIO 1 : DADOS DE GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 128.7
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 127.0
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 77.5

ANEXO VI.C - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1987 COM O META-COBOL (GRUPO MÉDIO DE 10 PROGRAMADORES)

RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	17.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	62.7 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.6 %

RELATORIO 5 : APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	16.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	59.2 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	18.2 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.3 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	0.8 %

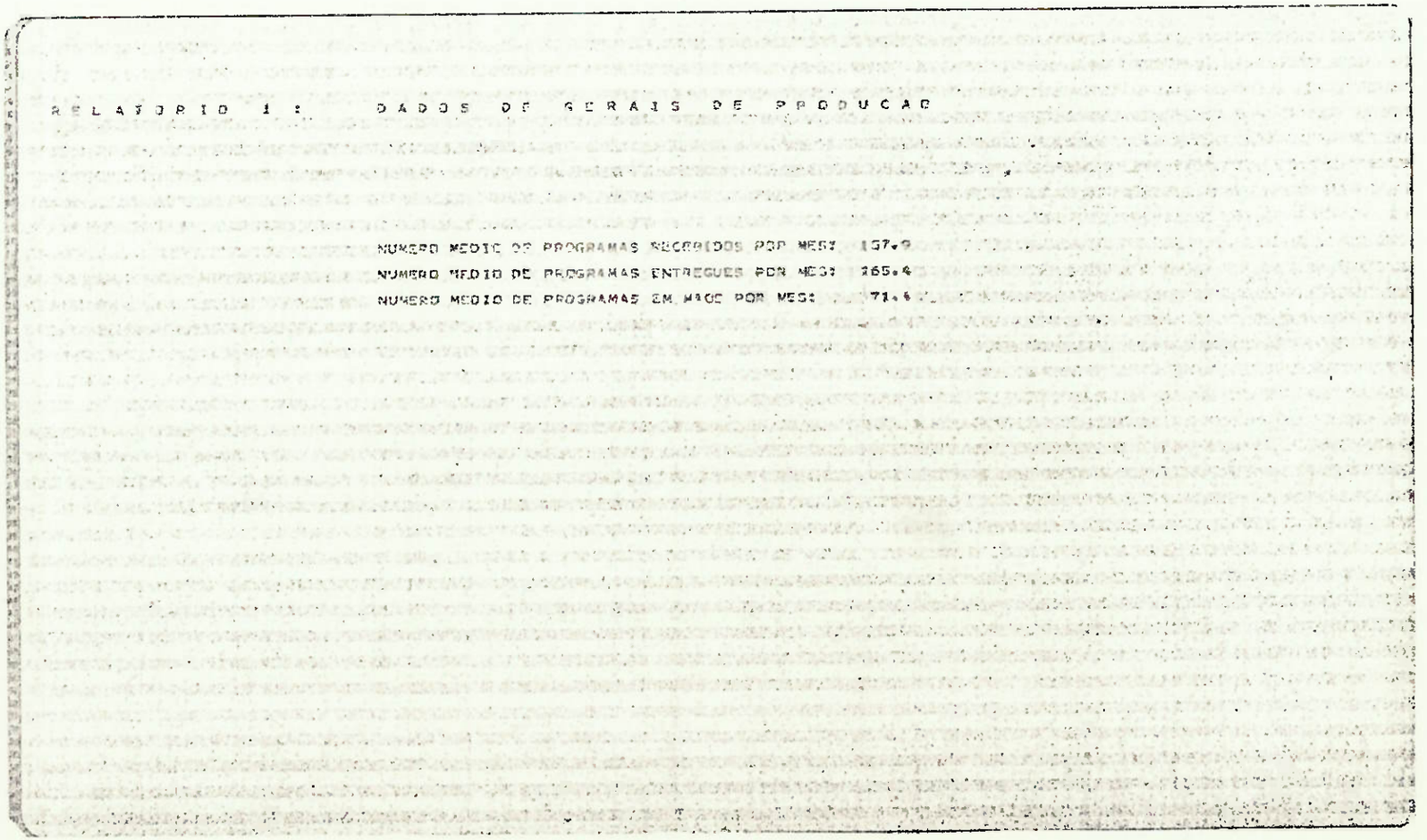
CONTINUAÇÃO DO ANEXO VI.C

RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	20.85 DIAS UTEIS
BATCH/RELATORIO	20.92 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	0.66 DIAS UTEIS
BATCH/RELATORIO	0.66 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	1.15 DIAS UTEIS
BATCH/RELATORIO	0.90 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	22.01 DIAS UTEIS
BATCH/RELATORIO	21.82 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VI.C



RELATORIO 1 : DADOS DE GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS ESCRITOS POR MES: 137.9
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 265.4
NUMERO MEDIO DE PROGRAMAS EM MANUT POR MES: 71.6

ANEXO VI.D - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1988 COM O META-COBOL (GRUPO MÉDIO DE 13 PROGRAMADORES)

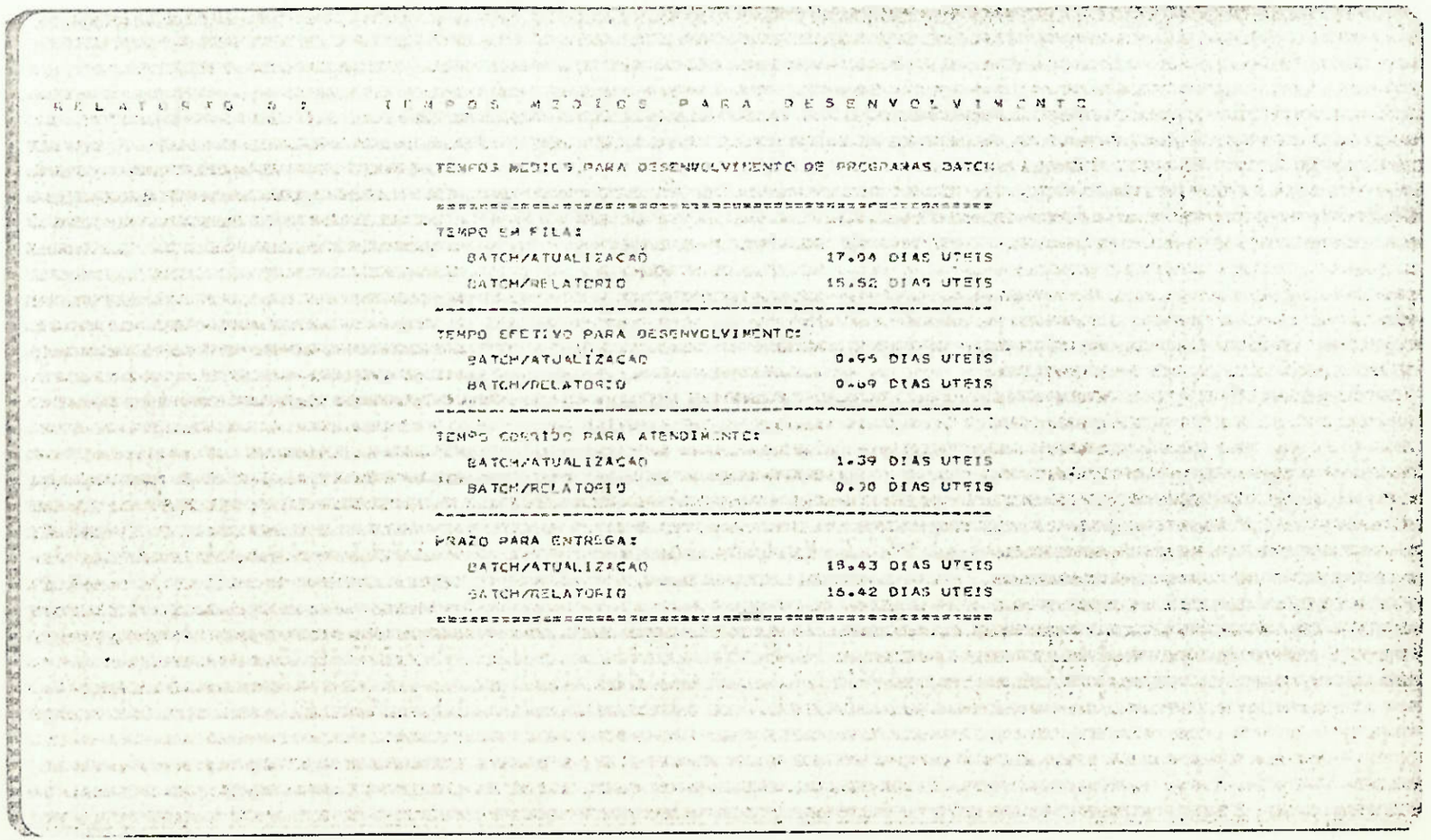
RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	16.8 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	64.1 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.0 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	15.6 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	59.5 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	17.6 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.6 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	0.8 %

CONTINUAÇÃO DO ANEXO VI.D



```
RELATORIO 5 :     TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

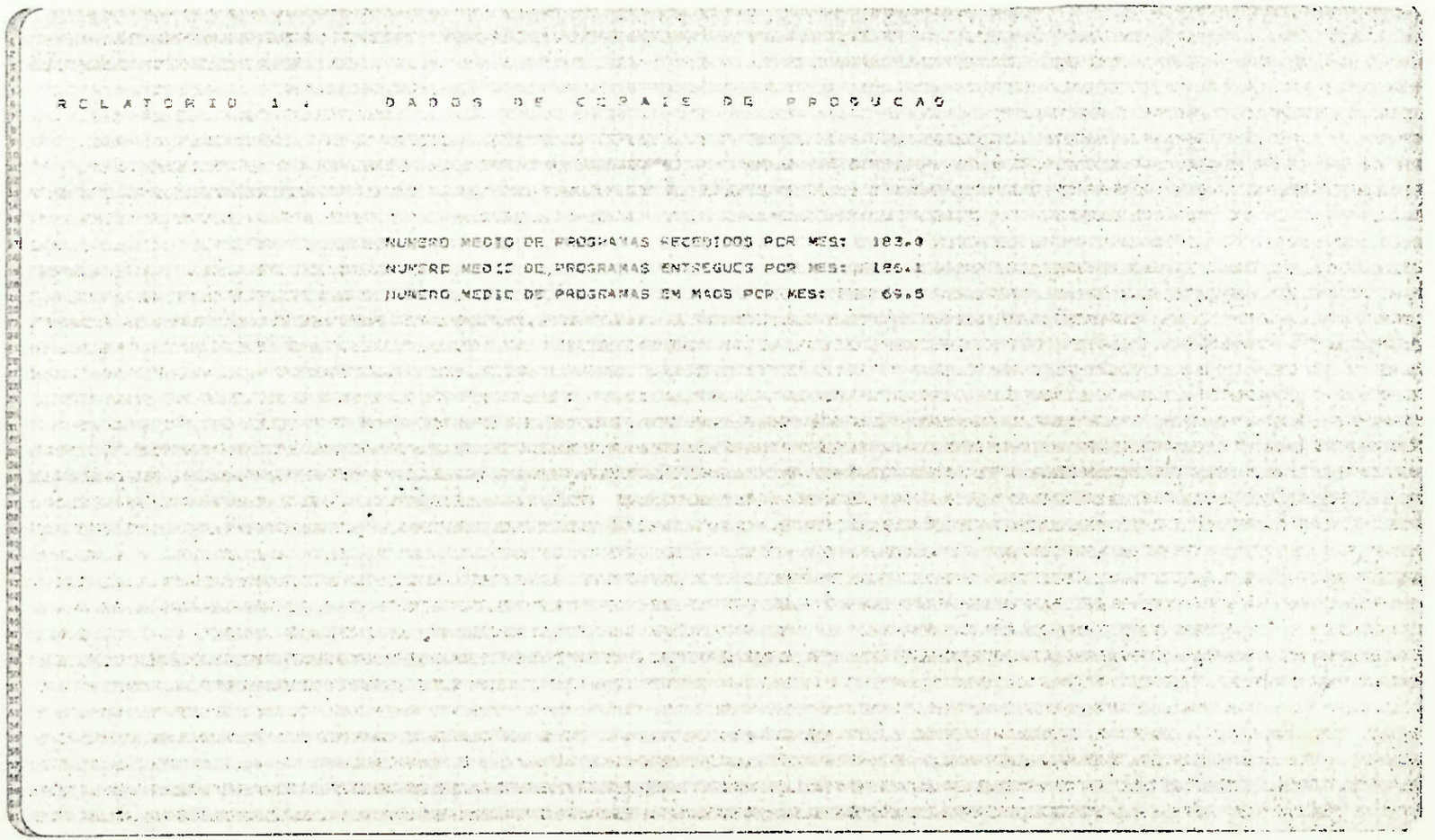
TEMPO EM FILA:
   BATCH/ATUALIZACAO                    17.04 DIAS UTEIS
   BATCH/RELATORIO                      15.52 DIAS UTEIS

TEMPO EFETIVO PARA DESENVOLVIMENTO:
   BATCH/ATUALIZACAO                     0.95 DIAS UTEIS
   BATCH/RELATORIO                       0.89 DIAS UTEIS

TEMPO DECORRIDO PARA ATENDIMENTO:
   BATCH/ATUALIZACAO                     1.39 DIAS UTEIS
   BATCH/RELATORIO                       0.90 DIAS UTEIS

PRAZO PARA ENTREGA:
   BATCH/ATUALIZACAO                    18.43 DIAS UTEIS
   BATCH/RELATORIO                      16.42 DIAS UTEIS
```

CONTINUAÇÃO DO ANEXO VI.D



RELATORIO 1 . DADOS DE CONTROLE DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 183.0
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 185.1
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 69.5

ANEXO VI.E - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1989 COM O META-COBOL (GRUPO MÉDIO DE 14 PROGRAMADORES)

RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	16.8 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	64.8 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	18.6 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	15.5 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	59.8 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	17.2 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.5 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	0.7 %

CONTINUAÇÃO DO ANEXO VI.E

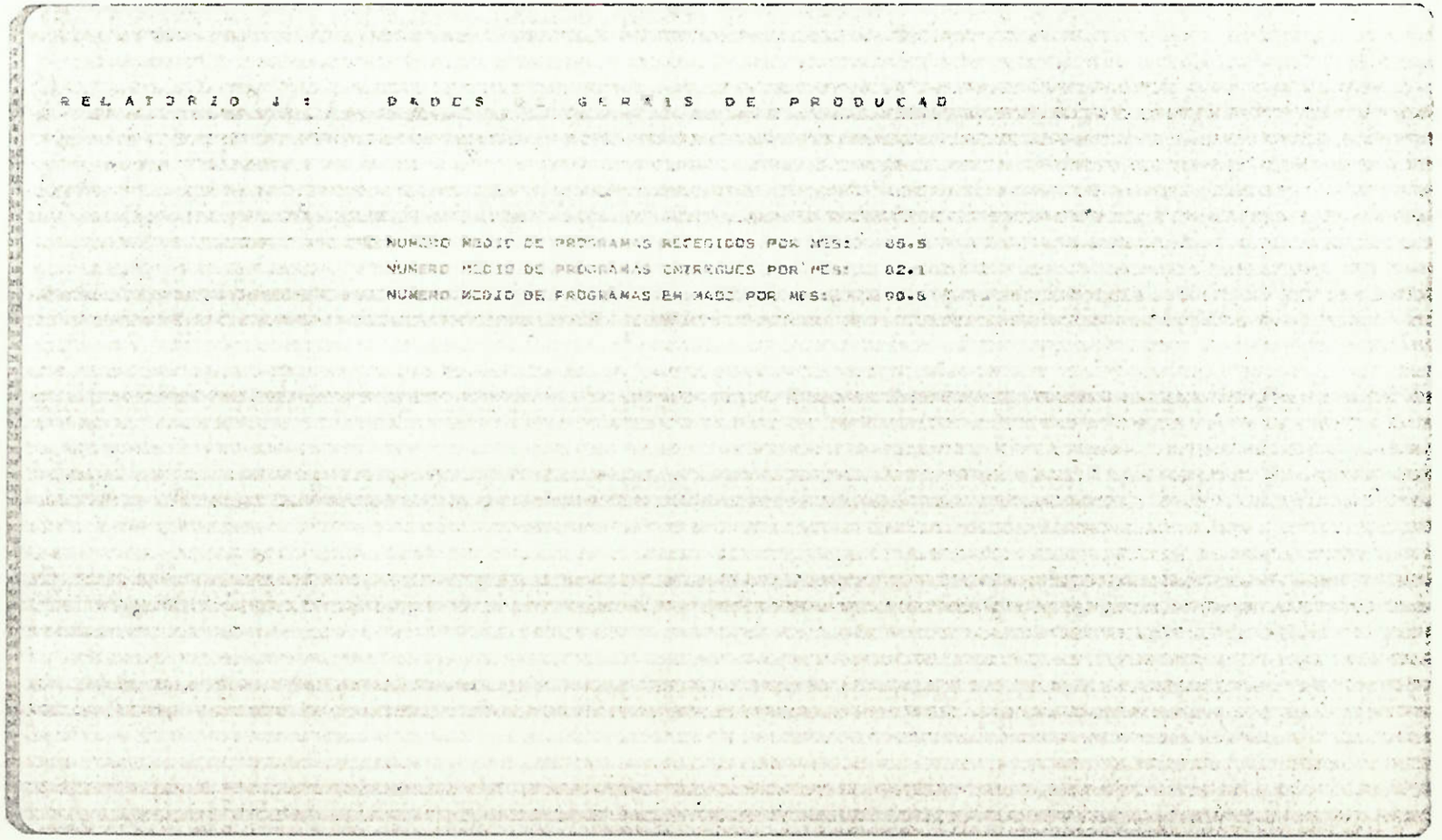
RELATORIO A 2 TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	12.44 DIAS UTEIS
BATCH/RELATORIO	13.24 DIAS UTEIS
TEMPO GASTO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	0.81 DIAS UTEIS
BATCH/RELATORIO	0.71 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	1.09 DIAS UTEIS
BATCH/RELATORIO	0.99 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	13.54 DIAS UTEIS
BATCH/RELATORIO	14.23 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VI.E

ANEXO VII - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO PERÍODO DE 1985 a 1989 SEM O META-COBOL



RELATORIO J : DADOS GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RETEGIDOS POR MES: 85.5
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 82.1
NUMERO MEDIO DE PROGRAMAS EM MANUT POR MES: 90.8

ANEXO VII.A - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1985 SEM O META-COBOL (GRUPO MÉDIO DE 8 PROGRAMADORES)

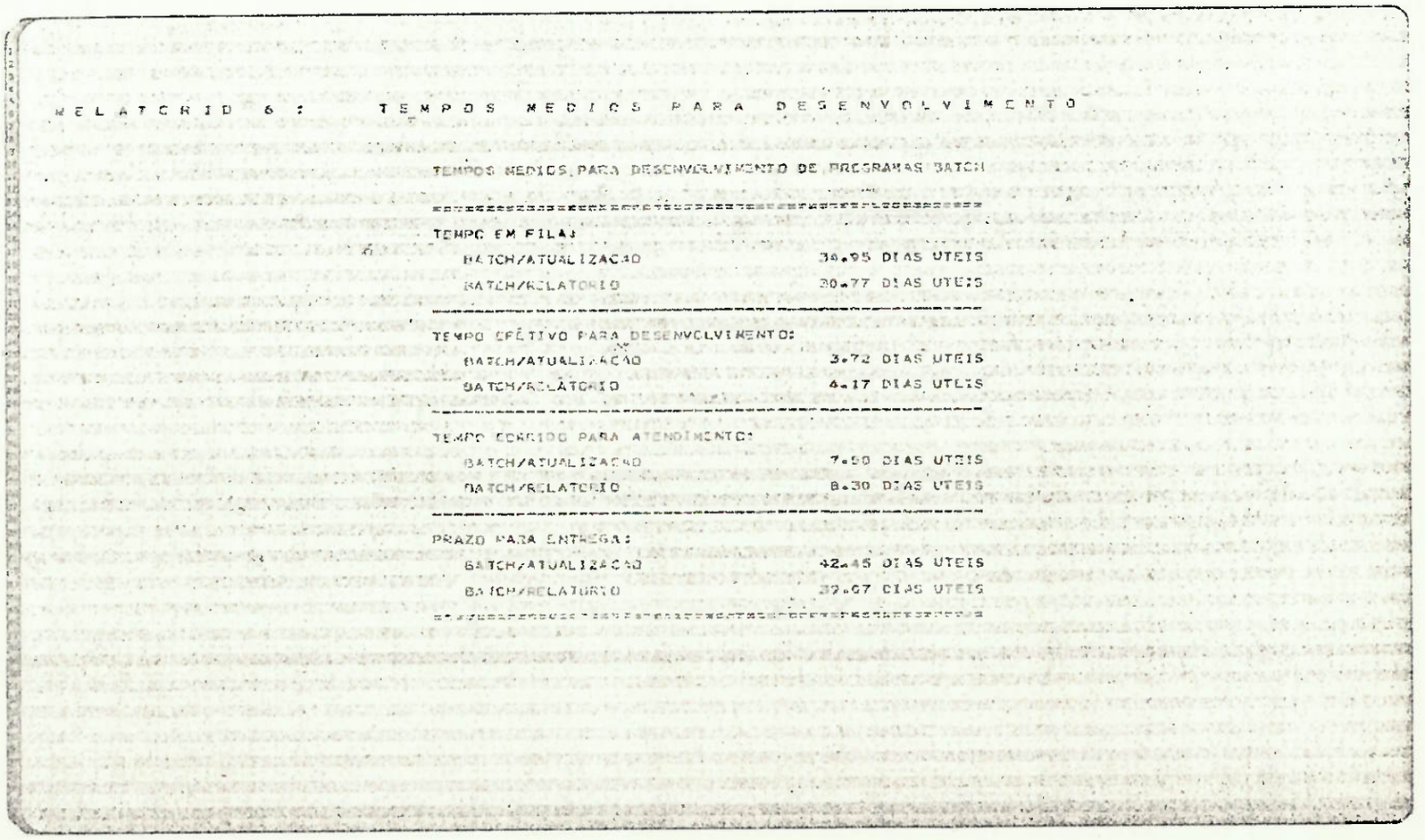
RELATORIO 4: APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS: 31.9 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS: 49.2 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES: 18.8 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO: 20.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS: 40.1 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES: 17.3 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS: 6.6 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO: 9.8 %

CONTINUAÇÃO DO ANEXO VII.A

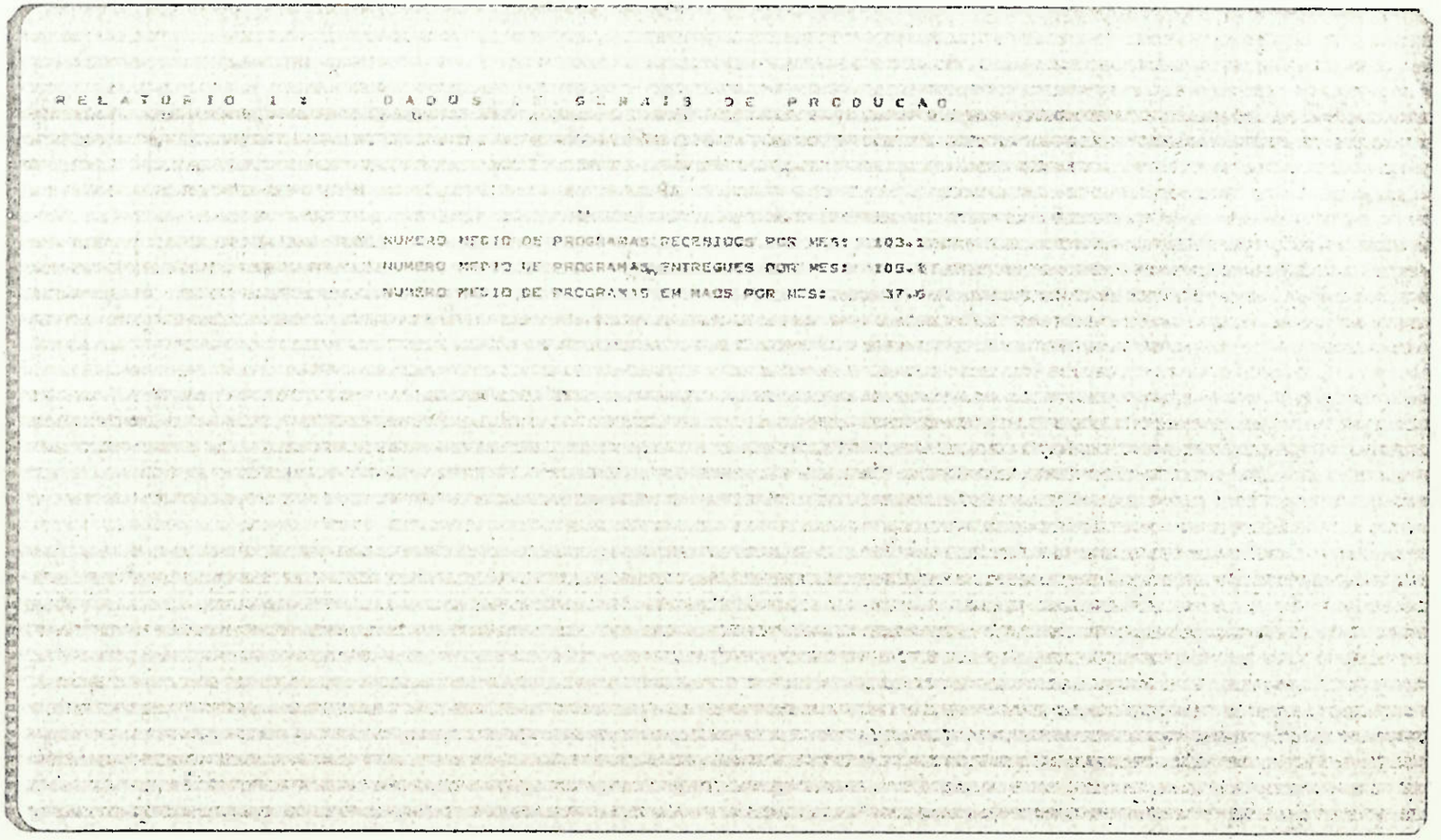


RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	34.95 DIAS UTEIS
BATCH/RELATORIO	20.77 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	3.72 DIAS UTEIS
BATCH/RELATORIO	4.17 DIAS UTEIS
TEMPO GASTO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	7.50 DIAS UTEIS
BATCH/RELATORIO	8.30 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	42.45 DIAS UTEIS
BATCH/RELATORIO	32.67 DIAS UTEIS

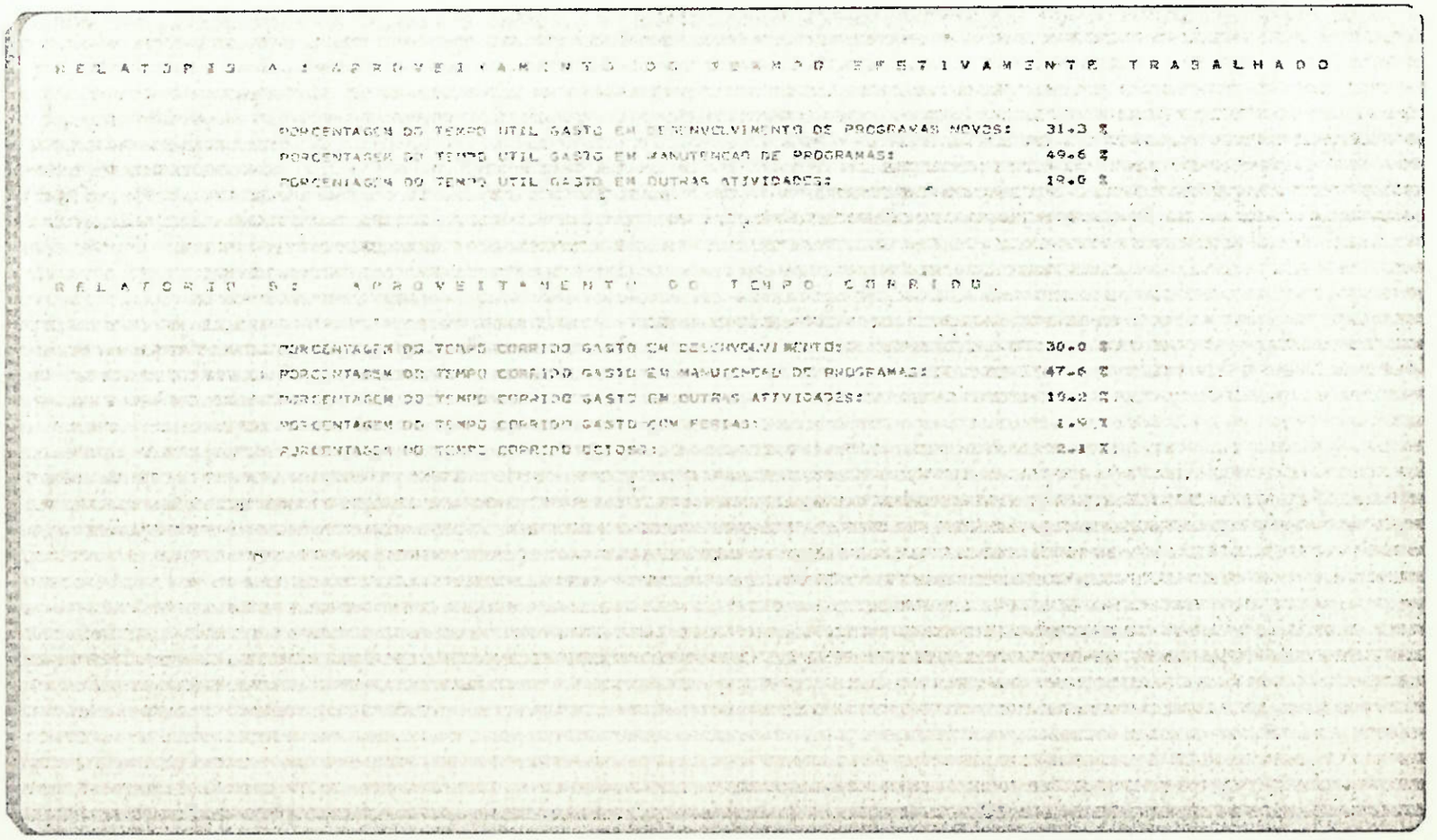
CONTINUAÇÃO DO ANEXO VII.A



RELATORIO 1 : DADOS GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 103.1
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 105.6
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 37.5

ANEXO VII.B - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1986 SEM O META-COBOL (GRUPO MÉDIO DE 10 PROGRAMADORES)



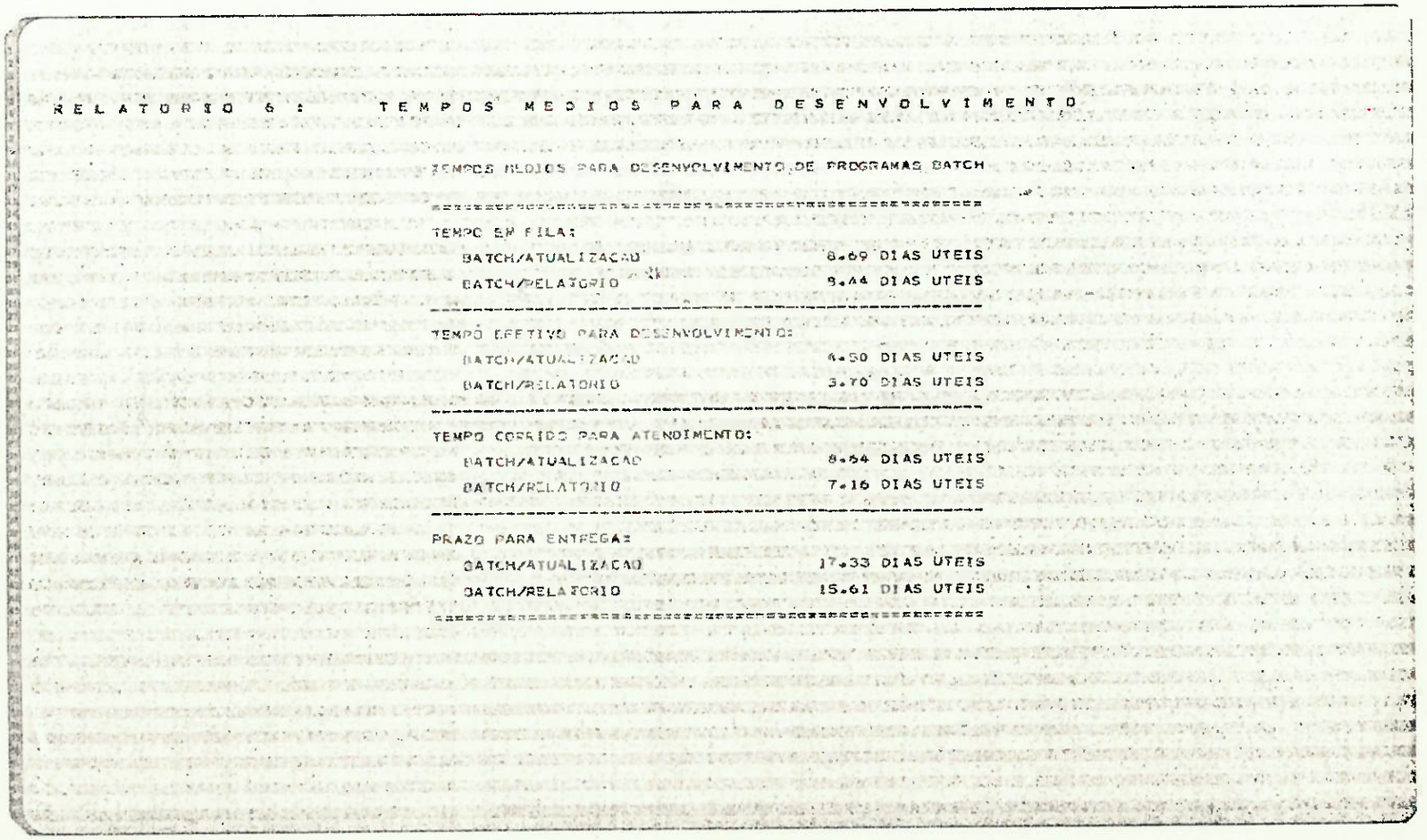
RELATORIO A : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	31.3 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	49.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.0 %

RELATORIO B: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	30.0 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	47.6 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	19.2 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	1.9 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	2.3 %

CONTINUAÇÃO DO ANEXO VII.B

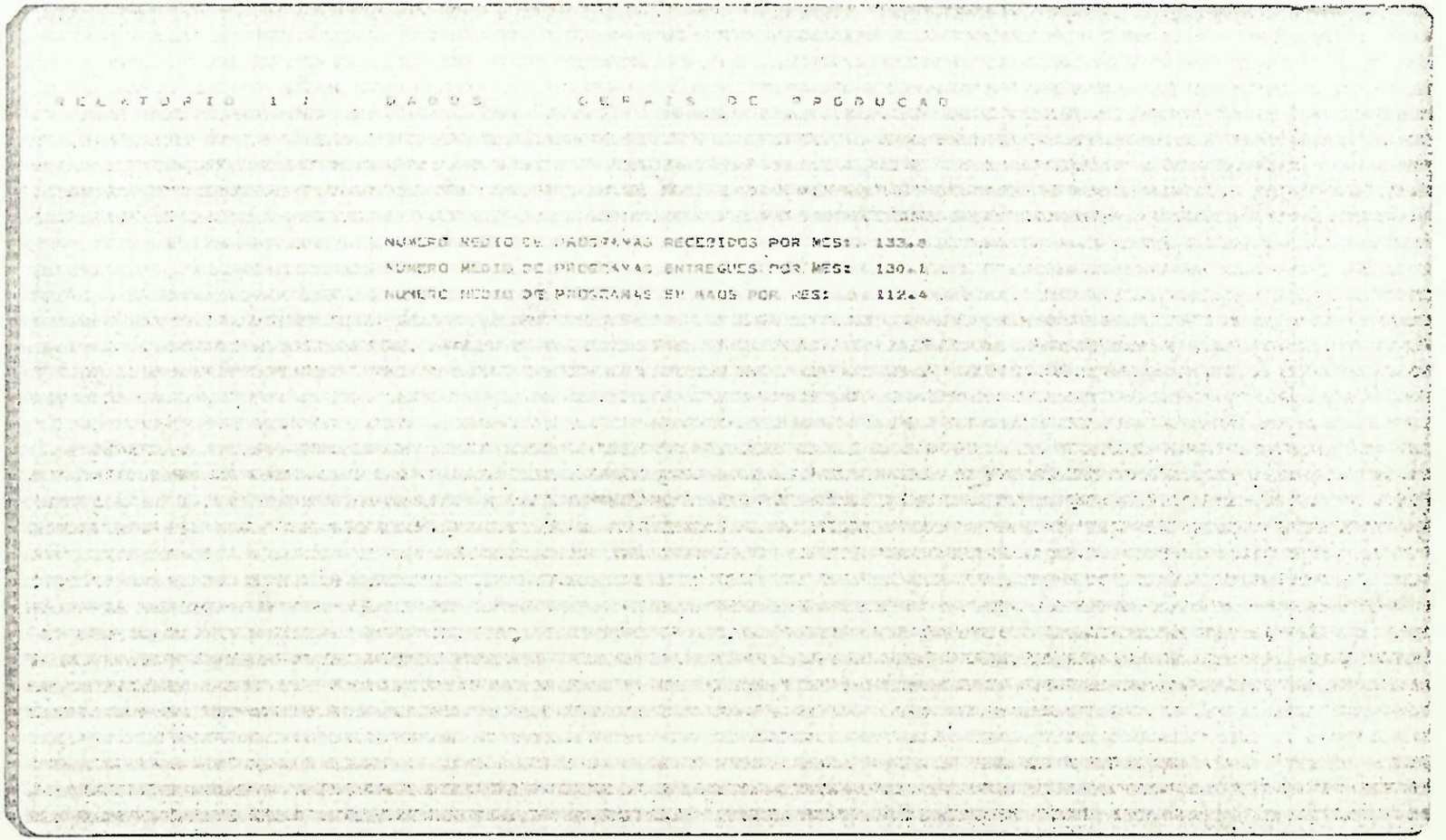


RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	8.69 DIAS UTEIS
BATCH/RELATORIO	9.44 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	4.50 DIAS UTEIS
BATCH/RELATORIO	3.70 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	8.54 DIAS UTEIS
BATCH/RELATORIO	7.16 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	17.33 DIAS UTEIS
BATCH/RELATORIO	15.61 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VII.B



ANEXO VII.C - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1987 SEM O META-COBOL (GRUPO MÉDIO DE 12 PROGRAMADORES)

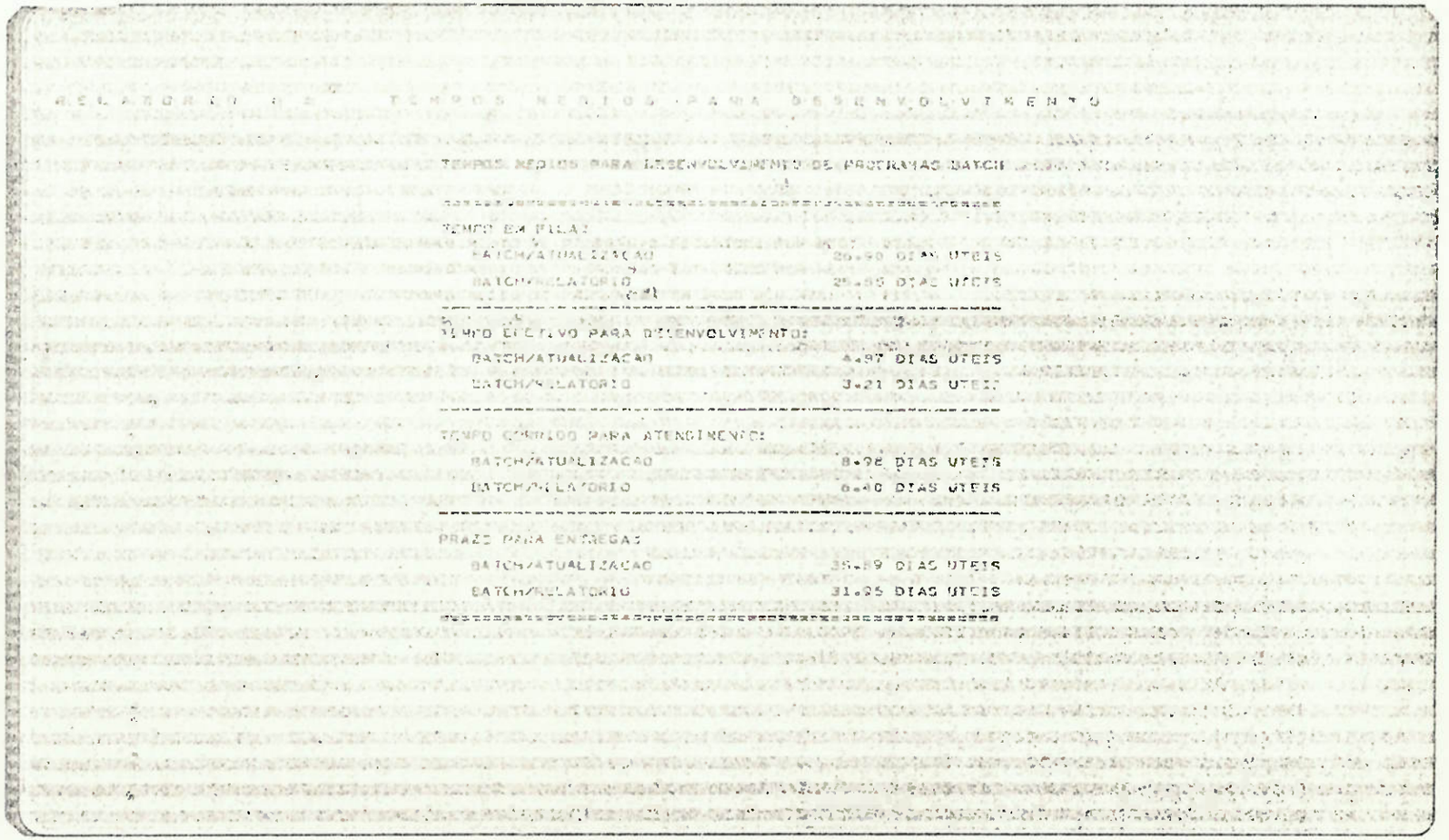
RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	28.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	53.0 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	18.3 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	26.3 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	48.8 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	16.8 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.3 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	1.5 %

CONTINUAÇÃO DO ANEXO VII.C



RELATORIO DE TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:

BATCH/ATUALIZACAO	20.50 DIAS UTEIS
BATCH/RELATORIO	25.85 DIAS UTEIS

TEMPO EFETIVO PARA DESENVOLVIMENTO:

BATCH/ATUALIZACAO	4.97 DIAS UTEIS
BATCH/RELATORIO	3.21 DIAS UTEIS

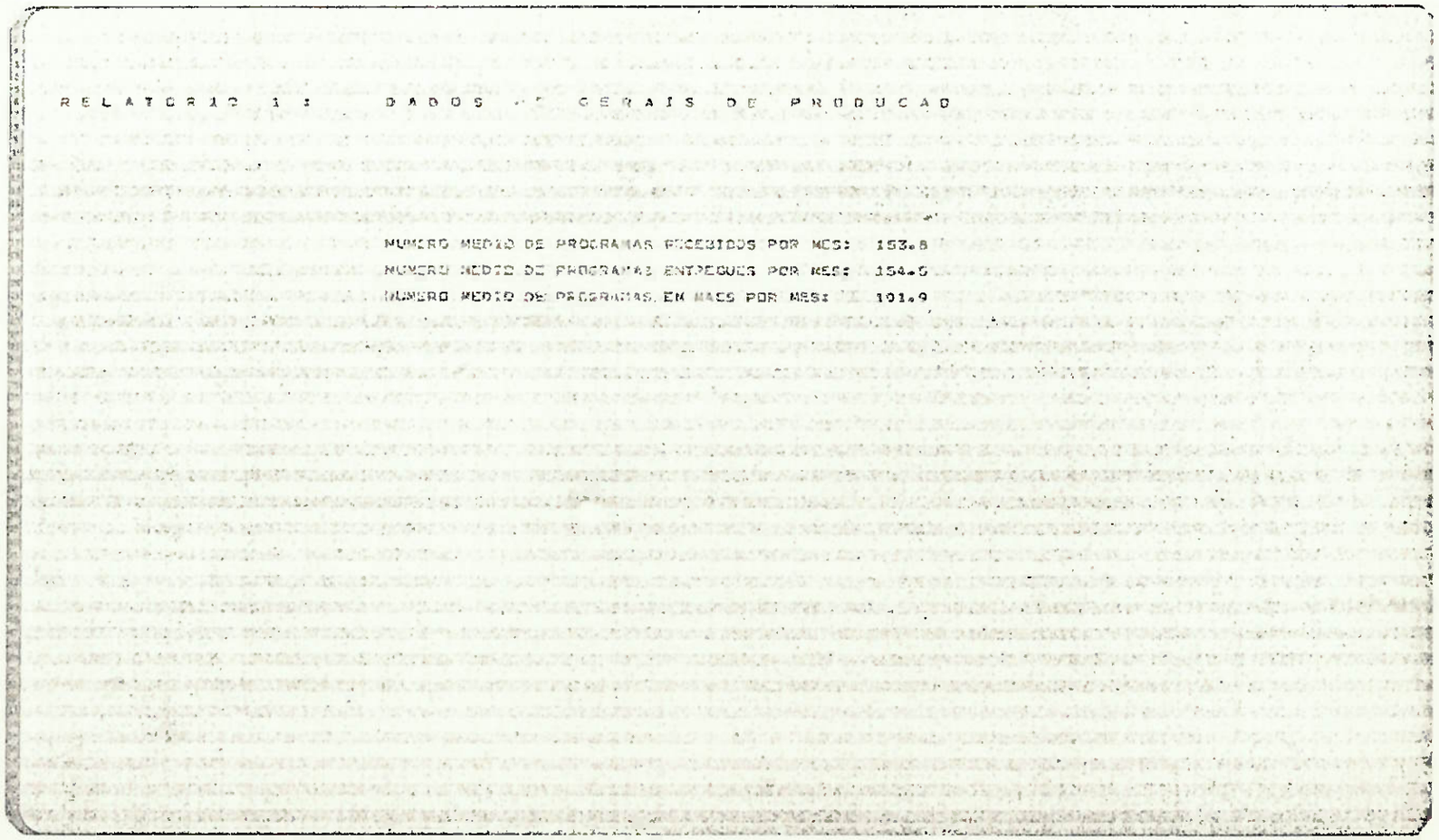
TEMPO GASTO PARA ATENDIMENTO:

BATCH/ATUALIZACAO	8.98 DIAS UTEIS
BATCH/RELATORIO	6.40 DIAS UTEIS

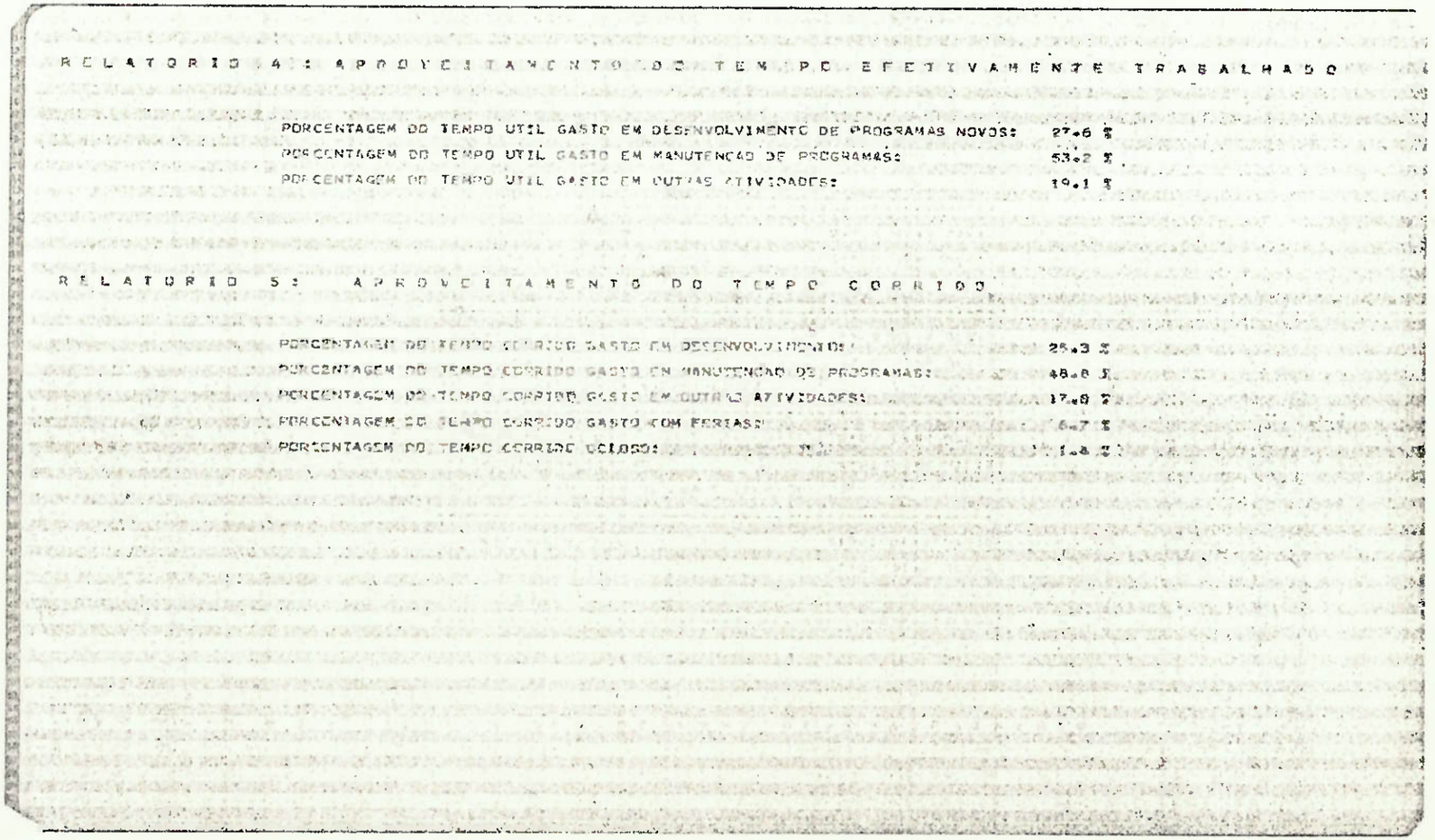
PRAZO PARA ENTREGAS:

BATCH/ATUALIZACAO	35.19 DIAS UTEIS
BATCH/RELATORIO	31.85 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VII.C



ANEXO VII.D - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1988 SEM O META-COBOL (GRUPO MÉDIO DE 14 PROGRAMADORES)



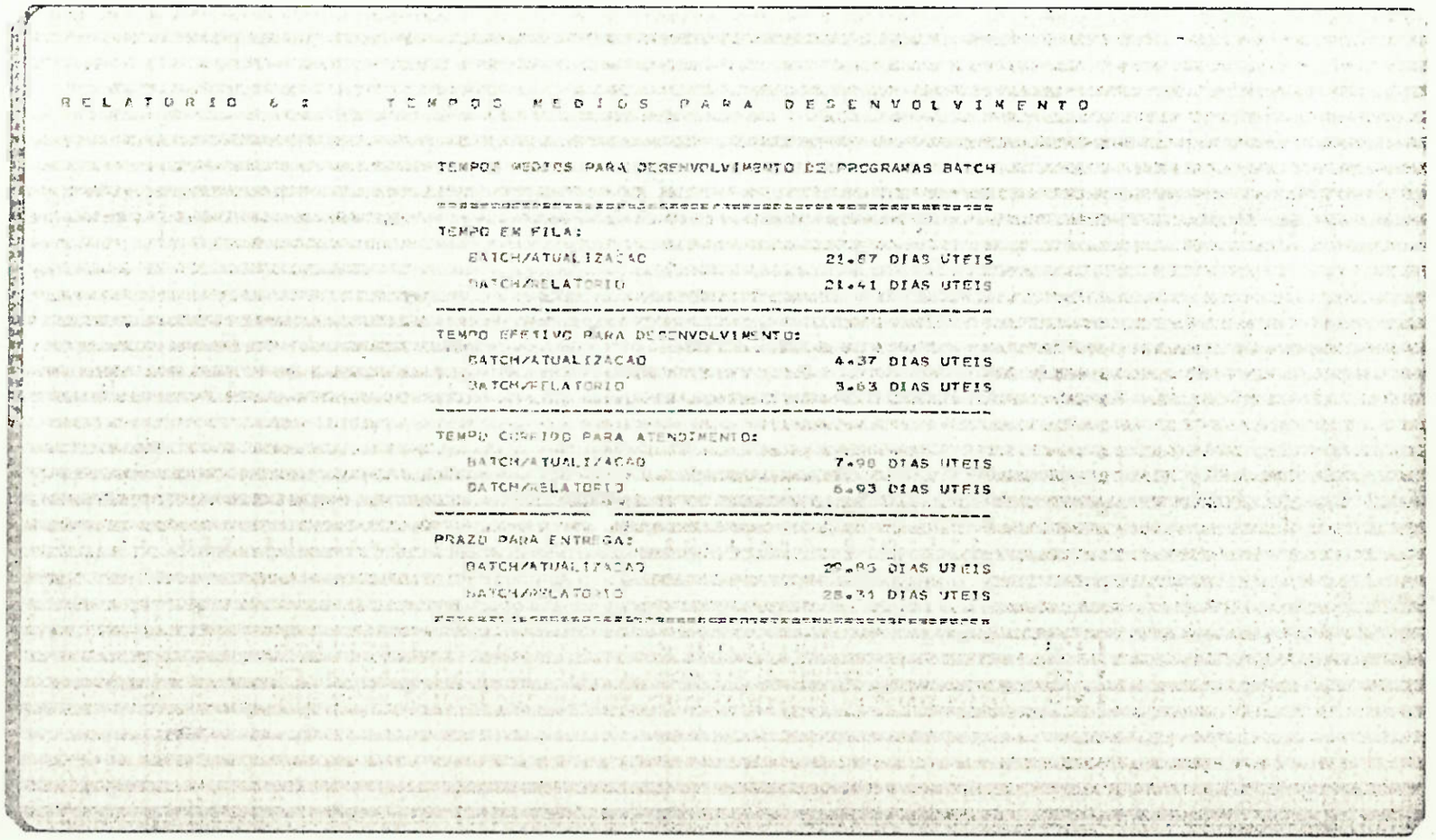
RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	27.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	53.2 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.1 %

RELATORIO 5 : APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	25.3 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	48.8 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	17.8 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	6.7 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	1.4 %

CONTINUAÇÃO DO ANEXO VII.D

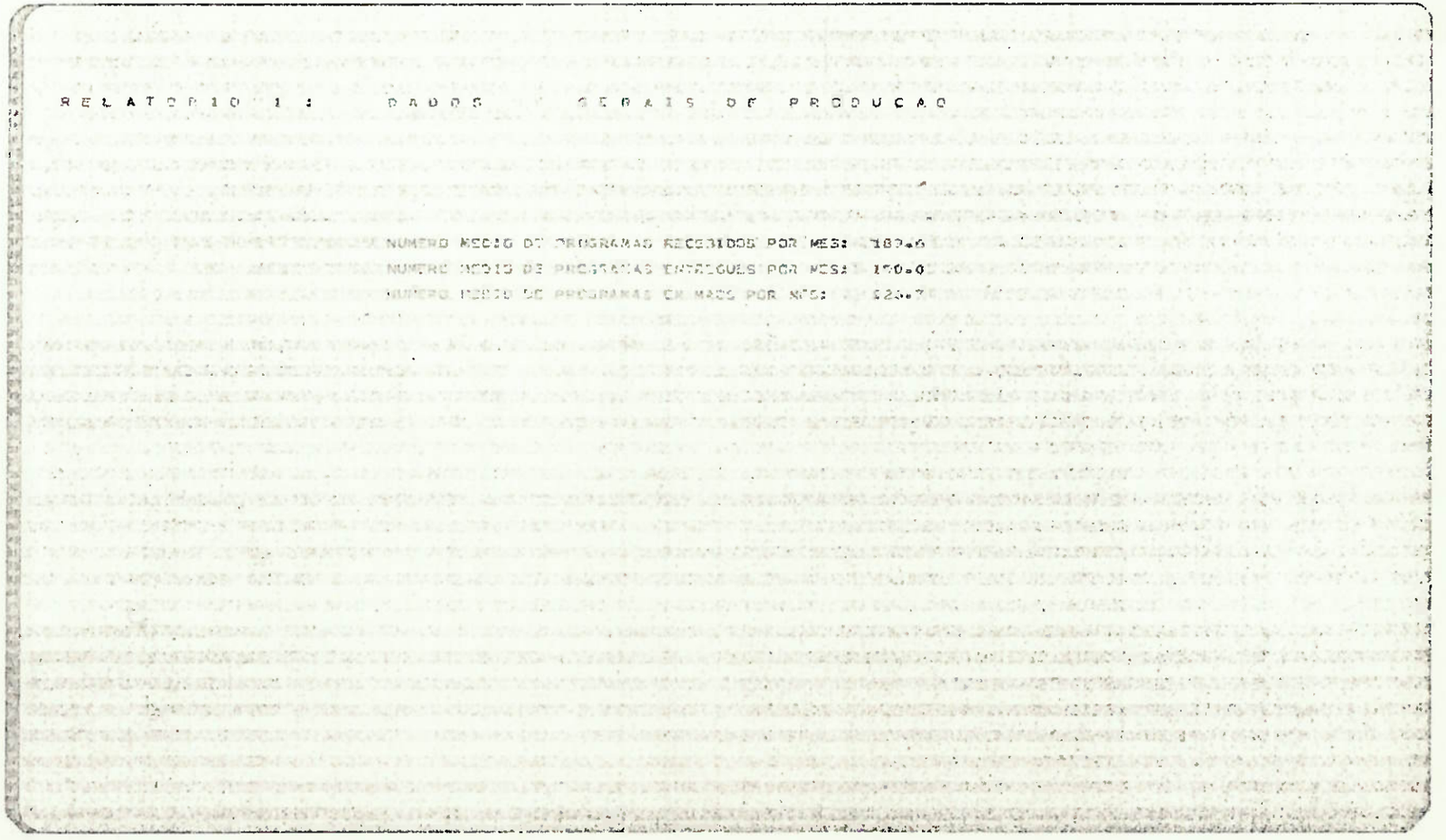


RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	21.67 DIAS UTEIS
BATCH/RELATORIO	21.41 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	4.37 DIAS UTEIS
BATCH/RELATORIO	3.63 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	7.98 DIAS UTEIS
BATCH/RELATORIO	6.93 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	29.85 DIAS UTEIS
BATCH/RELATORIO	28.34 DIAS UTEIS

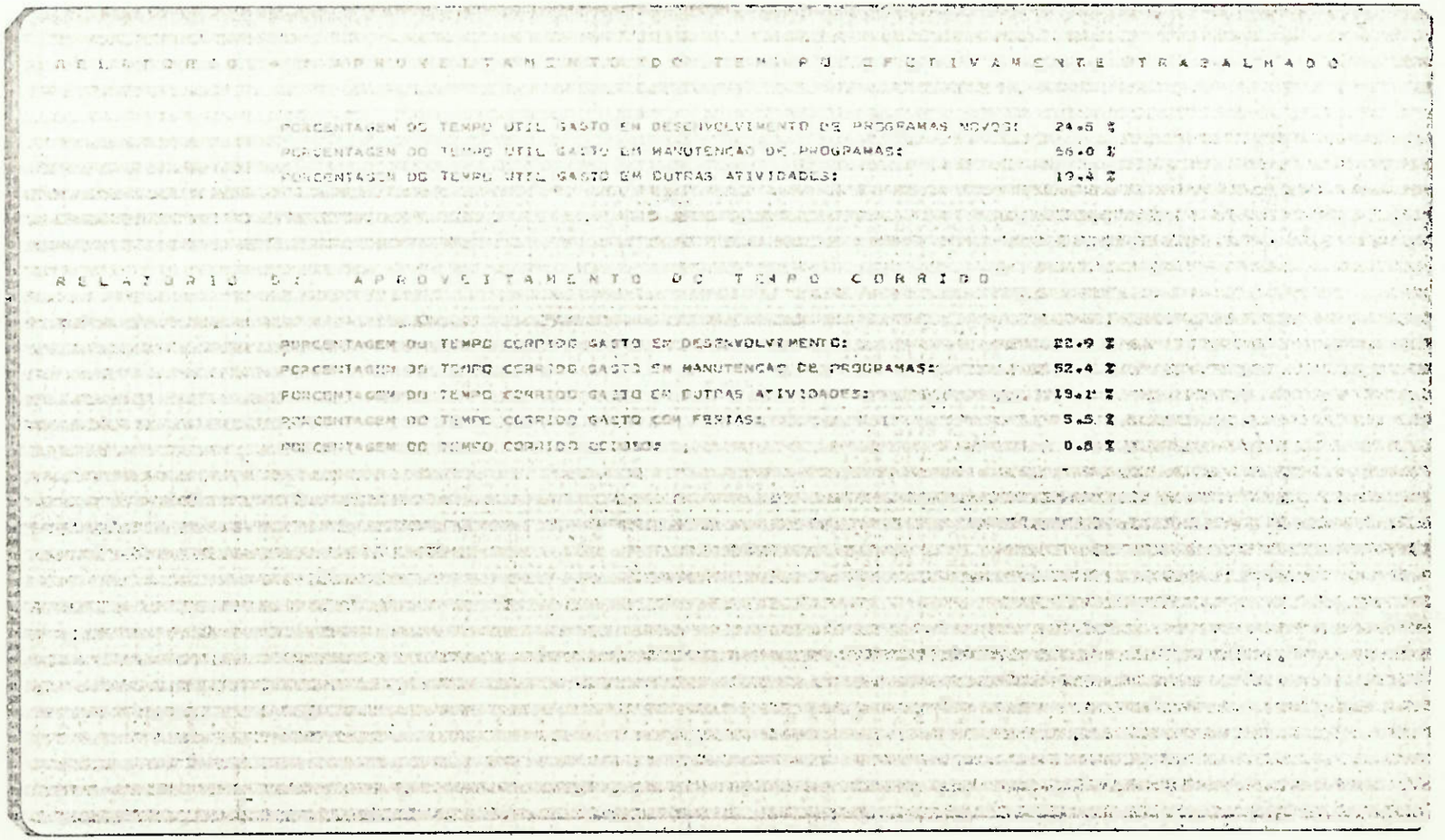
CONTINUAÇÃO DO ANEXO VII.D



RELATORIO 1 : DADOS GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 183.6
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 150.0
NUMERO MEDIO DE PROGRAMAS EM MANU POR MES: 525.7

ANEXO VII.E - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1989 SEM O META-COBOL (GRUPO MÉDIO DE 17 PROGRAMADORES)



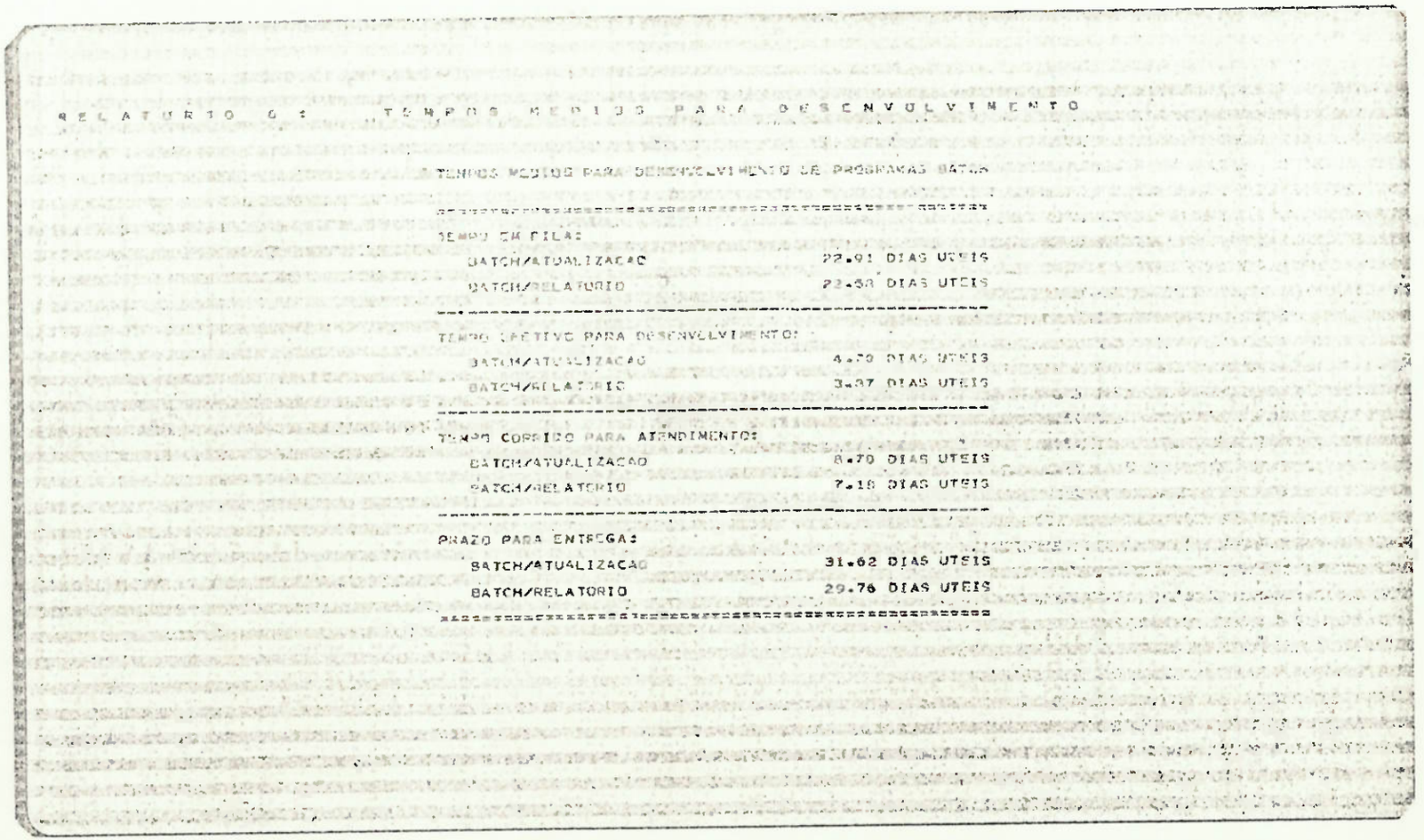
RELATORIO 4 - APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	24.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	56.0 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.4 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	22.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	52.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	19.1 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	5.5 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	0.0 %

CONTINUAÇÃO DO ANEXO VII.E



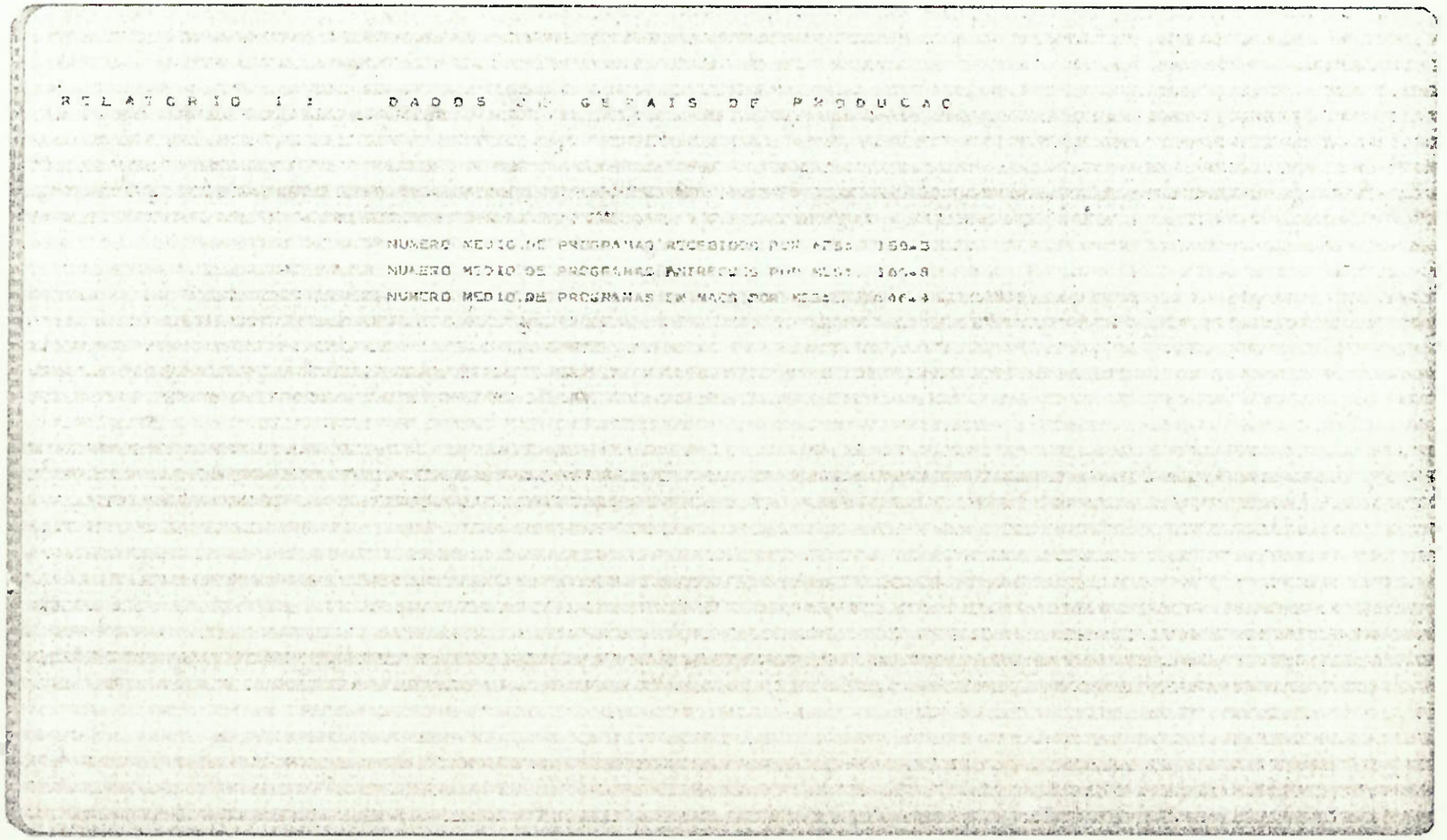
RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	22.91 DIAS UTEIS
BATCH/RELATORIO	22.58 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	4.79 DIAS UTEIS
BATCH/RELATORIO	3.87 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	8.70 DIAS UTEIS
BATCH/RELATORIO	7.18 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	31.62 DIAS UTEIS
BATCH/RELATORIO	29.76 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VII.E

ANEXO VIII - PRINCIPAIS RELATÓRIOS DAS SIMULAÇÕES PROCESSADAS PARA ANÁLISE DE SENSIBILIDADE QUANTO A VARIAÇÕES DO GANHO DE PRODUTIVIDADE PROPORCIONÁVEL PELO META-COBOL (ANO DE 1988 COM GRUPO MÉDIO DE 13 PROGRAMADORES)



ANEXO VIII.A - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DA REDUÇÃO À 10% DOS ATUAIS TEMPOS EFETIVOS PARA O DESENVOLVIMENTO DE PROGRAMAS "BATCH"

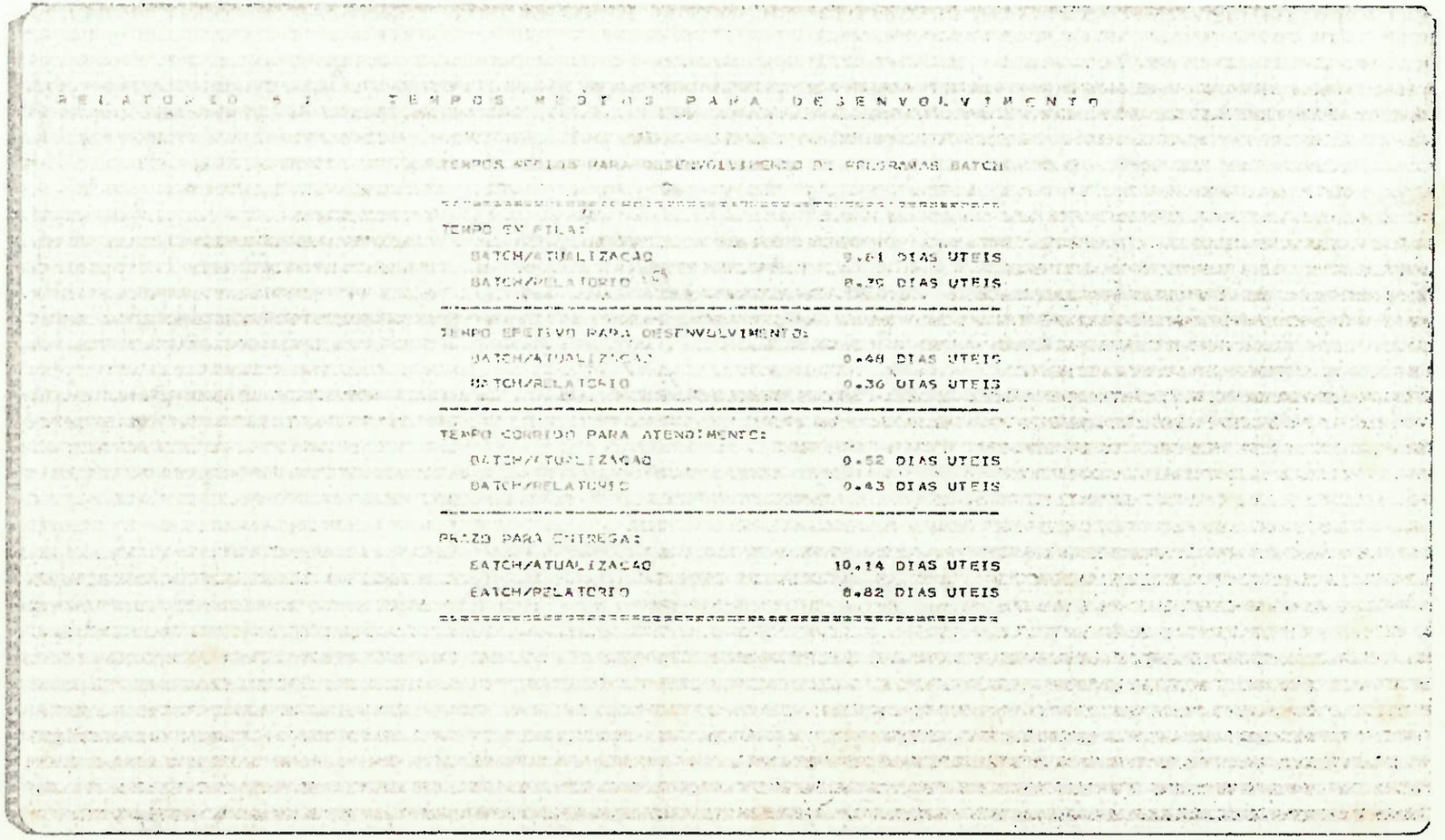
RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	16.2 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	63.6 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	20.1 %

RELATORIO 5 : APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	14.9 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	58.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	18.5 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	5.1 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	3.0 %

CONTINUAÇÃO DO ANEXO VIII.A



RELATORIO 5.2 TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS APLICAVEIS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:

BATCH/ATUALIZACAO	9.61 DIAS UTEIS
BATCH/RELATORIO	8.79 DIAS UTEIS

TEMPO EFETIVO PARA DESENVOLVIMENTO:

BATCH/ATUALIZACAO	0.48 DIAS UTEIS
BATCH/RELATORIO	0.36 DIAS UTEIS

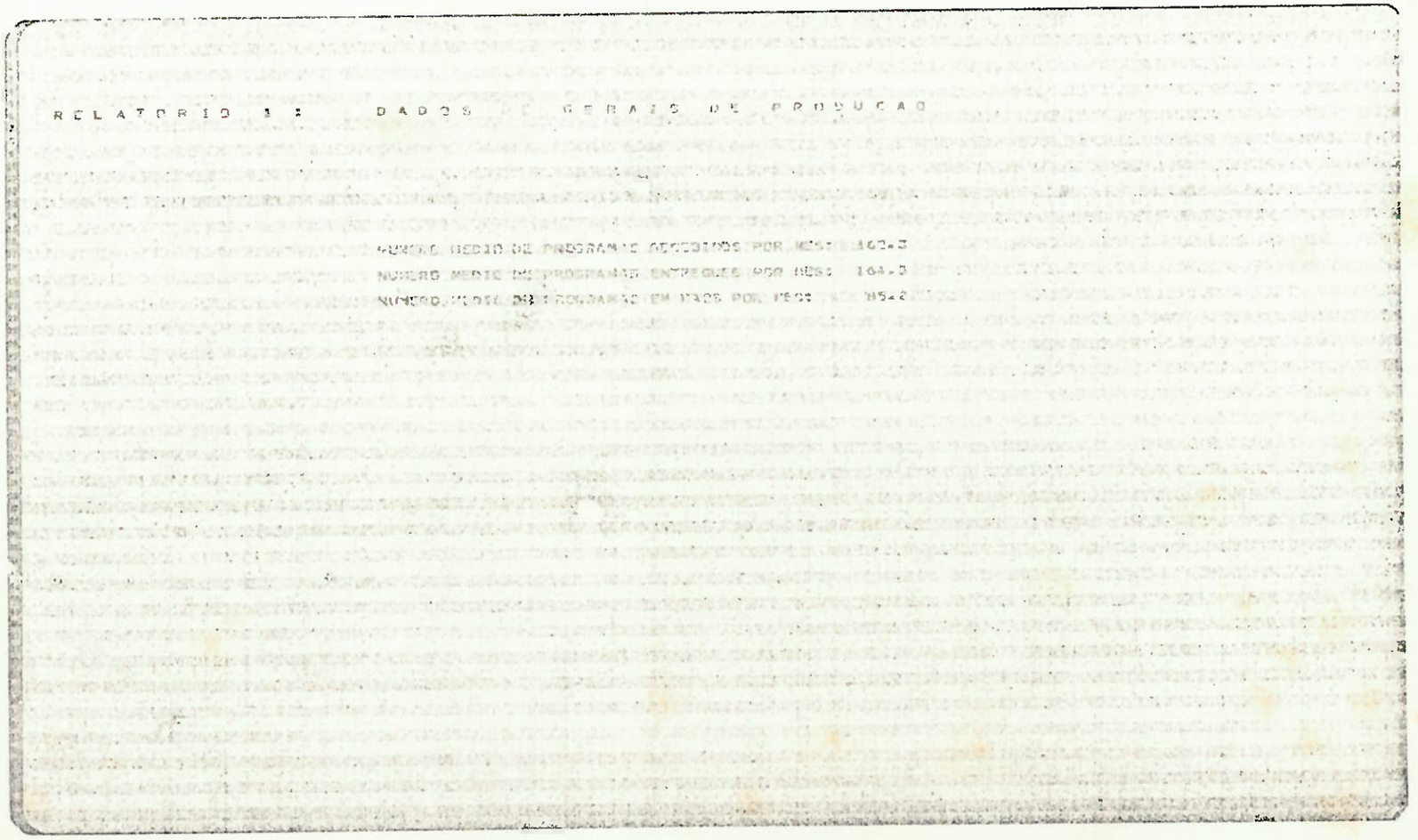
TEMPO CORRIDO PARA ATENDIMENTO:

BATCH/ATUALIZACAO	0.52 DIAS UTEIS
BATCH/RELATORIO	0.43 DIAS UTEIS

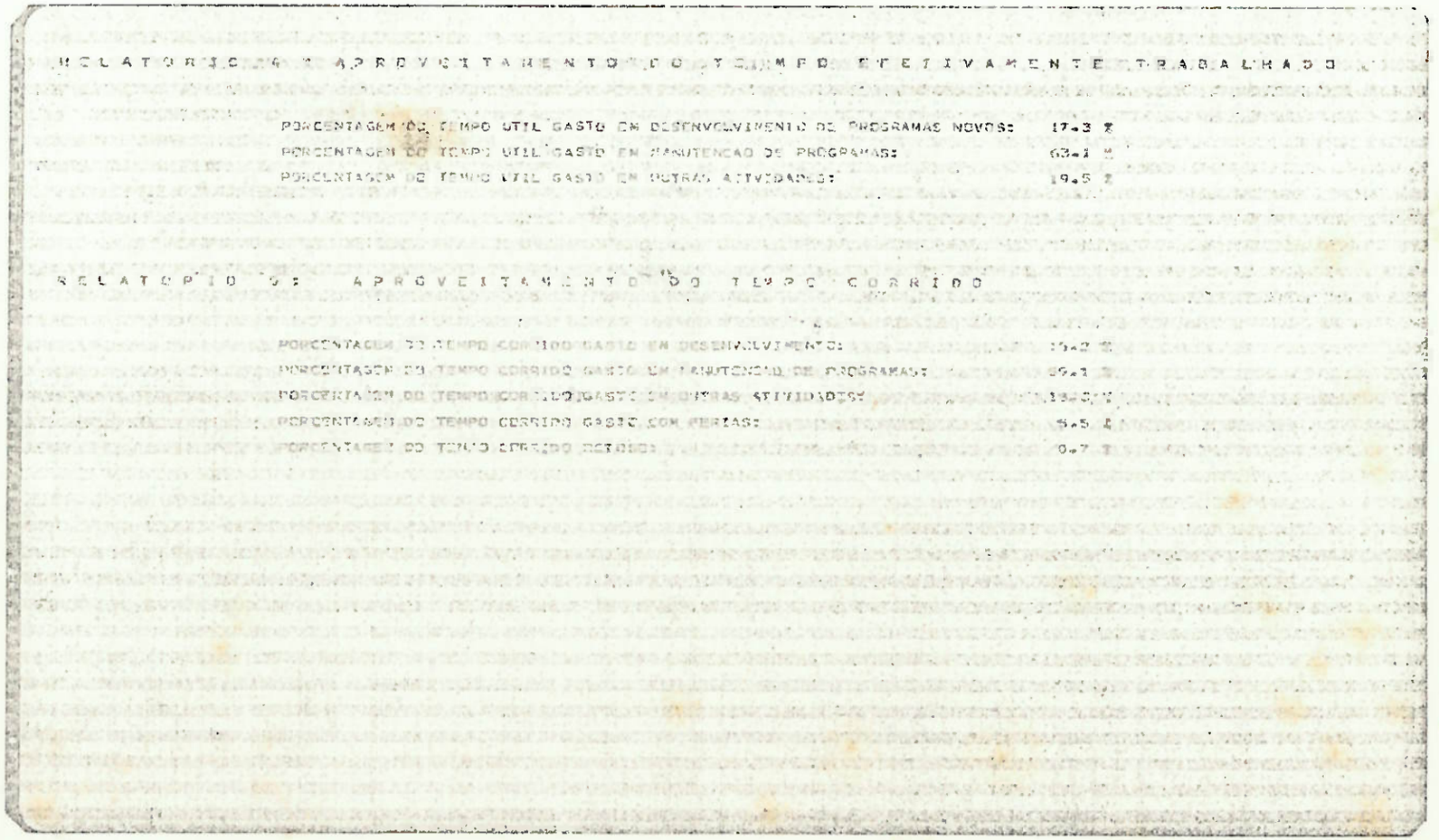
PRAZO PARA ENTREGA:

BATCH/ATUALIZACAO	10.14 DIAS UTEIS
BATCH/RELATORIO	8.82 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VIII.A



ANEXO VIII.B - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DA REDUÇÃO À 30% DOS ATUAIS TEMPOS EFETIVOS PARA O DESENVOLVIMENTO DE PROGRAMAS "BATCH"



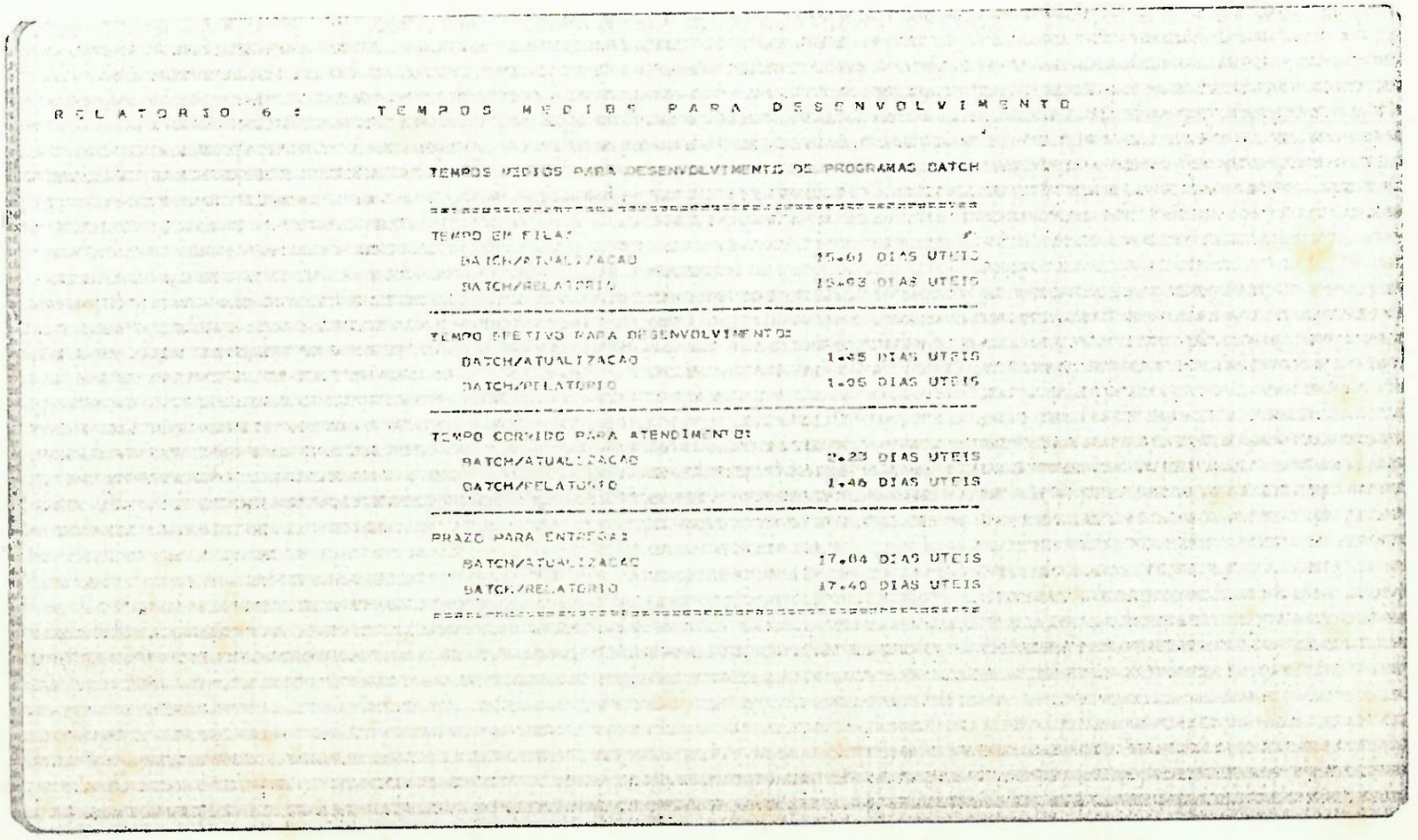
RELATORIO 8: APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS: 17.3 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS: 63.1 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES: 19.5 %

RELATORIO 9: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO: 10.2 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS: 59.1 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES: 18.0 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS: 5.5 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO: 0.7 %

CONTINUAÇÃO DO ANEXO VIII.B



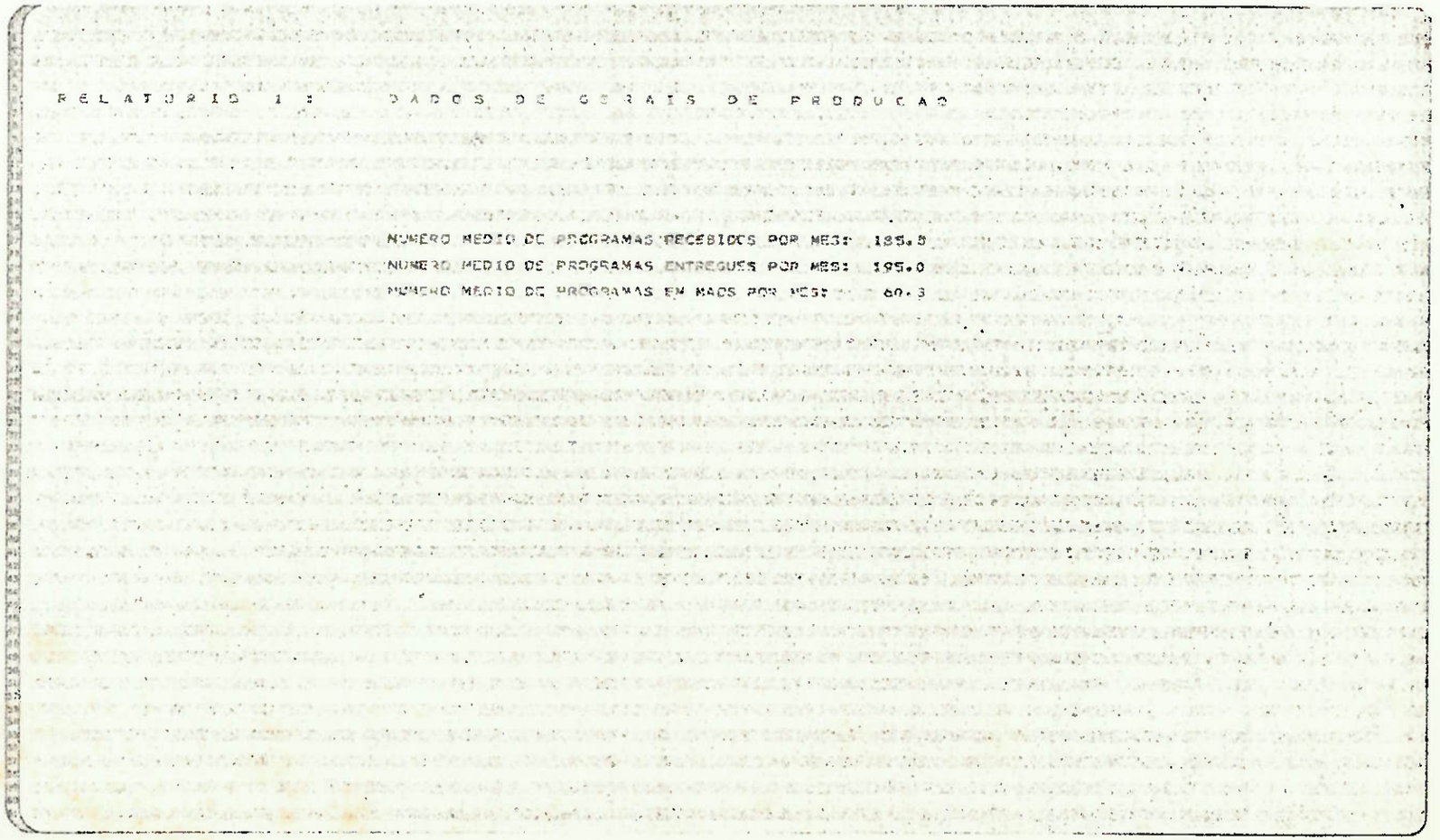
RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	15.61 DIAS UTEIS
BATCH/RELATORIO	15.83 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	1.85 DIAS UTEIS
BATCH/RELATORIO	1.95 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	2.23 DIAS UTEIS
BATCH/RELATORIO	1.46 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	17.04 DIAS UTEIS
BATCH/RELATORIO	17.40 DIAS UTEIS

CONTINUAÇÃO DO ANEXO VIII.B

ANEXO IX - PRINCIPAIS RELATÓRIOS DAS SIMULAÇÕES PROCESSADAS PARA ANÁLISE DE SENSIBILIDADE QUANTO A VARIAÇÕES DO TAMANHO DO GRUPO MÉDIO (ANO DE 1989 COM O META-COBOL)



RELATORIO 1 : DADOS DE GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 185.5
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 195.0
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 60.3

ANEXO IX.A - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1989 COM O META-COBOL (GRUPO MÉDIO DE 15 PROGRAMADORES)

RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	19.1 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	61.4 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	19.3 %

RELATORIO 5: APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	18.1 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	58.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	18.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	3.9 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	1.0 %

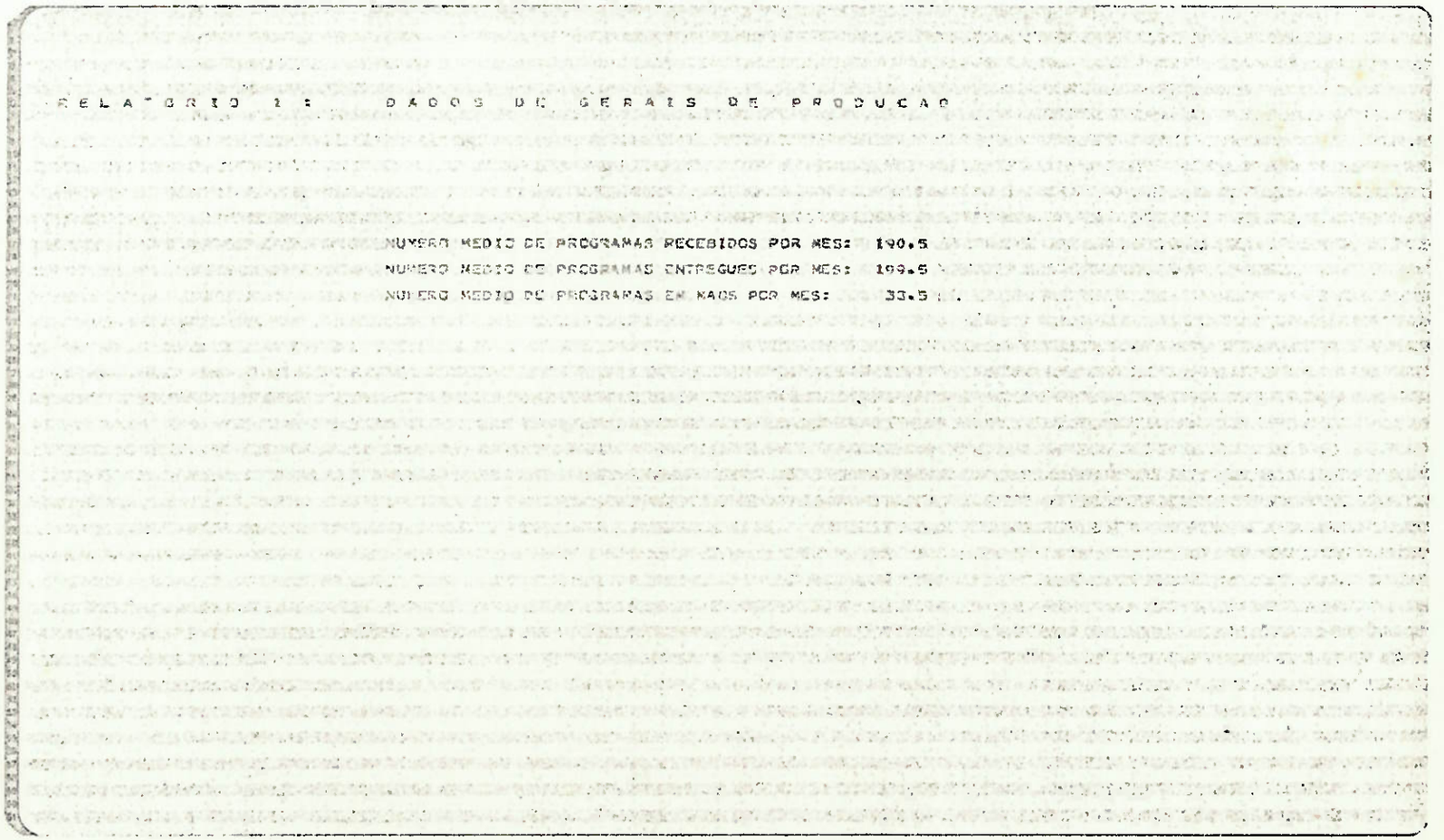
CONTINUAÇÃO DO ANEXO IX.A

RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	10.96 DIAS UTEIS
BATCH/RELATORIO	10.60 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	1.02 DIAS UTEIS
BATCH/RELATORIO	0.32 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	1.50 DIAS UTEIS
BATCH/RELATORIO	1.05 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	12.49 DIAS UTEIS
BATCH/RELATORIO	11.06 DIAS UTEIS

CONTINUAÇÃO DO ANEXO IX.A



RELATORIO 1 : DADOS DE GERAIS DE PRODUCAO

NUMERO MEDIO DE PROGRAMAS RECEBIDOS POR MES: 190.5
NUMERO MEDIO DE PROGRAMAS ENTREGUES POR MES: 199.5
NUMERO MEDIO DE PROGRAMAS EM MAOS POR MES: 33.5

ANEXO IX.B - PRINCIPAIS RELATÓRIOS DA SIMULAÇÃO DO ANO DE 1989 COM O META-COBOL (GRUPO MÉDIO DE 16 PROGRAMADORES)

RELATORIO 4 : APROVEITAMENTO DO TEMPO EFETIVAMENTE TRABALHADO

PORCENTAGEM DO TEMPO UTIL GASTO EM DESENVOLVIMENTO DE PROGRAMAS NOVOS:	15.9 %
PORCENTAGEM DO TEMPO UTIL GASTO EM MANUTENCAO DE PROGRAMAS:	63.2 %
PORCENTAGEM DO TEMPO UTIL GASTO EM OUTRAS ATIVIDADES:	20.9 %

RELATORIO 5 : APROVEITAMENTO DO TEMPO CORRIDO

PORCENTAGEM DO TEMPO CORRIDO GASTO EM DESENVOLVIMENTO:	14.4 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM MANUTENCAO DE PROGRAMAS:	57.2 %
PORCENTAGEM DO TEMPO CORRIDO GASTO EM OUTRAS ATIVIDADES:	18.8 %
PORCENTAGEM DO TEMPO CORRIDO GASTO COM FERIAS:	5.5 %
PORCENTAGEM DO TEMPO CORRIDO OCIOSO:	3.8 %

CONTINUAÇÃO DO ANEXO IX.B

RELATORIO 6 : TEMPOS MEDIOS PARA DESENVOLVIMENTO

TEMPOS MEDIOS PARA DESENVOLVIMENTO DE PROGRAMAS BATCH

TEMPO EM FILA:	
BATCH/ATUALIZACAO	5.97 DIAS UTEIS
BATCH/RELATORIO	4.30 DIAS UTEIS
TEMPO EFETIVO PARA DESENVOLVIMENTO:	
BATCH/ATUALIZACAO	0.97 DIAS UTEIS
BATCH/RELATORIO	0.83 DIAS UTEIS
TEMPO CORRIDO PARA ATENDIMENTO:	
BATCH/ATUALIZACAO	1.29 DIAS UTEIS
BATCH/RELATORIO	1.15 DIAS UTEIS
PRAZO PARA ENTREGA:	
BATCH/ATUALIZACAO	7.26 DIAS UTEIS
BATCH/RELATORIO	5.46 DIAS UTEIS

CONTINUAÇÃO DO ANEXO IX.B

BIBLIOGRAFIA

Referências citadas no texto, conforme ordem de citação:

1. LEME, Leonardo Lellis Pereira. Introdução aos Sistemas de Gerência de Bancos de Dados. São Paulo, Edgard Blücher, 1980.

2. GANE, Chris. Análise Estruturada de Sistemas. Rio de Janeiro, LTC, 1984.

3. TAURION, Cezar. Algumas dicas para avaliar os produtos disponíveis. Data News, Rio de Janeiro, p.28, 31 jul. 1984.

4. EM dois anos, dez produtos lançados. Data News, Rio de Janeiro, p.30, 31 jul. 1984.

5. BARNES, Ralph Mosser. Estudo de Movimentos e de Tempos. São Paulo, Edgard Blücher, 1977.

6. COSTA NETO, Pedro Luís de Oliveira. Estatística. São Paulo, Edgard Blücher, 1977.

7. FRANCISCHINI, Paulino Graciano. Estimativa de prazos e orçamento de programas. São Paulo, 1980.

8. SHANNON, Robert E. Systems Simulation: the art and science. Englewood Cliffs, N.J., Prentice-Hall Inc., 1975.

9. BUENO NETO, Pedro Rodrigues. Simulação. São Paulo, FCAV, 1982.

10. GORDON, Geoffrey. System Simulation. Englewood Cliffs, N.J., Prentice-Hall Inc., 1978.

11. IBM Corp. General Purpose Simulation System V - User's Manual. New York, 1975.

12. BURROUGHS. General Purpose Simulation System (GPSS)-User's Manual. Detroit, 1979.

13. HAHN, Gerald J.; SHAPIRO, Samuel S. Statistical Models in Engineering. New York, John Wiley & Sons Inc., 1967.

14. TEIXEIRA, Sergio R.Pinto. Engenharia de software: experiência e recomendações. São Paulo, Edgard Blücher, 1979.

Outras referências consultadas:

15. ABRIL S.A. CULTURAL. Enciclopédia Prática de Informática. São Paulo, 1984.

16. ASHCAR, Décio. Processamento de dados nas Empresas. São Paulo, FCAV, 1978.

17. BARRON, D.W. Assemblers and Loaders. London, Mac Donald and Jane's, 1972.

18. BASTOS, Alex C. Programação COBOL. Rio de Janeiro, LTC, FENAME/CAPRE, 1977.

19. BAUER, F.L. et alii. Software engineering. An advanced course. Munich, 1972.

20. CHU, Shao Yong. Banco de Dados: organização, sistemas, administração. São Paulo, Atlas, 1983.

21. FERRENTINO, Andrew B. Making software development estimates "good". Datamation, p.179-182, set, 1981.

22. FLEISCHER, Gerald A. Teoria da Aplicação do Capital: um estudo das decisões de investimento. São Paulo, Edgard Blücher, Ed. da Universidade de São Paulo, 1973.

23. GRIES, David. Compiler construction for digital computers. New York, John Wiley & Sons., 1971.

24. HOPGOOD, F.R.A. Compiling Techniques. London, Mac Donald and Jane's, 1971.

25. KLECKA, W.R.; NIE, N.H.; HULL, C.H. SPSS Primer, Mc Graw-Hill, N.Y., 1975.

26. MEYER, Paul L. Probabilidade: aplicações a estatística. Rio de Janeiro, LTC, 1982.

27. OSBORN, Robert W. Theories of productivity analysis. Datamation, p.212-216, set, 1981.

28. THADHANI, A.J. Interactive user productivity. IBM Systems Journal. Armonk, New York, 20(4), p.407-423, 1981.

29. ZAVE, Pamela. The operational versus the conversational approach to software development. Communications of the ACM. New York, 27(2), p.104-117, fev, 1984.