

LEONARDO FELIPE JAPUR DE SÁ

**DESENVOLVIMENTO DE UM SISTEMA DE
PROGRAMAÇÃO DE PRODUÇÃO**

Trabalho de Formatura apresentado à
Escola Politécnica da Universidade de
São Paulo para graduação em Engenharia
de Produção.

Área de concentração:
Programação de Produção

Orientador:
Miguel C. Santoro

São Paulo

2000

TF 2000
Santoro

Aos grandes amigos que conheci nesta
escola e que sempre ajudaram a enfrentar as
batalhas da sabedoria.

AGRADECIMENTOS

Agradeço ao grande mestre e orientador Prof. Miguel C. Santoro pela atenção prestada e pela confiança transmitida.

Aos meus familiares que sempre me apoiaram em minhas decisões e possibilitaram meu estudo nesta faculdade

Aos amigos que sempre permaneceram unidos

À namorada por sua paciência e companheirismo

A todos que direta ou indiretamente colaboraram na execução deste trabalho

SUMARIO

SUMARIO

1.	INTRODUÇÃO	1
1.1	O PROBLEMA DE PROGRAMAÇÃO DE PRODUÇÃO	1
1.2	ESCOLHA DO TEMA	2
1.3	OBJETIVO DO TRABALHO	3
2.	REVISÃO BIBLIOGRÁFICA: Administração da Produção	5
2.1	INTRODUÇÃO À ADMINISTRAÇÃO DA PRODUÇÃO	5
2.1.1	Framework de Anthony	6
2.1.2	Framework da Estrutura do Produto	8
2.2	GERENCIAMENTO DE ESTOQUES	10
2.2.1	Classificações	10
2.2.2	Custos de inventario	12
2.3	PREVISÃO DA DEMANDA	13
2.3.1	Características de uma previsão	13
2.3.2	Passos	14
2.3.3	Abordagens da previsão	15
2.4	PLANEJAMENTO AGREGADO	16
2.5	MRP	18
2.5.1	Definição	19
2.5.2	O que envolve o MRP	20
2.5.3	Formas de estrutura de produto	21
2.5.4	Cálculo MRP	22
2.5.5	MRP de ciclo fechado	23
2.5.6	Outros conceitos avançados	24
2.5.7	Críticas ao MRP	25
2.6	DIMENSIONAMENTO DE LOTES	26
2.6.1	Técnicas de dimensionamento de lotes	27
2.6.2	Exemplos numéricos e comparação	29
2.6.3	Ajustes no dimensionamento de lotes	30
2.7	PROGRAMAÇÃO DE PRODUÇÃO	30
2.7.1	Introdução	31
2.7.2	Sequenciação	32
3.	REVISÃO BIBLIOGRÁFICA: Outros conceitos	39
3.1	MODELOS E SIMULAÇÃO	39
3.1.1	Sistemas e Modelos	39
3.1.2	Classificações de modelos de simulação	41
3.1.3	Simulação de eventos discretos	43
3.2	PROJETO DE BANCO DE DADOS	47
4.	DEFINIÇÃO DO SISTEMA	50
4.1	O QUE DEFINE UM MODELO	51

SUMÁRIO

4.2	ABRANGÊNCIA DO MODELO	52
4.2.1	Premissas	52
4.2.2	Pontos Fortes e Limitações	54
4.3	ESCOLHAS UTILIZADAS	56
4.3.1	Contornando críticas do MRP	56
4.3.2	Técnica de Dimensionamento de lotes	57
4.3.3	Técnica de Sequenciação	59
4.4	ESCALA DE TEMPO	59
4.5	RELATORIOS	61
4.6	AMBIENTE DE DESENVOLVIMENTO	61
5.	DESENVOLVENDO O SISTEMA	64
5.1	ALGUNS PADRÕES NO DESENVOLVIMENTO	64
5.2	TABELAS	65
5.2.1	Tabelas de Registro	65
5.2.2	Tabelas de Transação	67
5.3	SEQUÊNCIA DE OPERAÇÕES	69
6.	UTILIZANDO O SISTEMA	71
6.1	INSERINDO UM MODELO	71
6.2	GERANDO PROGRAMA DE PRODUÇÃO	72
6.3	RELATORIOS	74
7.	APLICAÇÃO DO SISTEMA	75
7.1	FAMÍLIAS E ÍTEMS	75
7.1.1	Famílias	75
7.1.2	Ítems	78
7.2	LISTA DE MATERIAIS	79
7.3	ROTEIROS	79
7.4	OPERAÇÕES	79
7.5	RECURSOS DE TRABALHO	80
8.	RESULTADOS E CONCLUSÕES	81
8.1	CONFIGURAÇÕES DO MODELO	81
8.2	RESULTADOS	82
8.3	CONCLUSÕES	83
9.	BIBLIOGRAFIA	84
9.1	REFERÊNCIAS BIBLIOGRÁFICAS	84
9.2	BIBLIOGRAFIA RECOMENDADA	85

SUMÁRIO

ANEXO A - CÓDIGO DE PROGRAMAÇÃO	87
ANEXO B - FORMULÁRIOS	133
ANEXO C - RELATÓRIOS	138
ANEXO D - DADOS DO EXEMPLO	170
ANEXO E - ESTRUTURA DO BANCO DE DADOS	200

LISTA DE FIGURAS

Figura 2.1 – Desenho Esquemático do planejamento...	20
Figura 2.2 – Diferentes formas de estrutura de produto	21
Figura 2.3 – Cálculo de necessidades líquidas no MRP.....	22
Figura 2.4 – MRP de ciclo fechado	24
Figura 2.5 – Classificação hierarquizada de <i>job shops</i> - exemplo	38
Figura 3.1 – Meios de estudar um problema.....	40
Figura 3.2 – Fluxograma de controle do avanço de tempo.....	45
Figura 4.1 – Gráfico de tempo simulado x tempo real (horas)	61
Figura 6.1 – Formulário principal do sistema.....	71
Figura 6.2 – Exemplo de descrição da informação necessária	72
Figura 6.2 – Formulário de Encomendas.....	73
Figura 7.1 – Estrutura de itens da família CADR (Cadeiras)	76
Figura 7.2 – Estrutura de itens da família MESA (Mesas).....	77
Figura D.1 – Formulário de item.....	133
Figura D.2 – Formulário de operação.....	133
Figura D.3 – Formulário de entrada de operações	134
Figura D.4 – Formulário de recurso de trabalho	134
Figura D.5 – Formulário de disponibilidade	134
Figura D.6 – Formulário de família.....	135
Figura D.7 – Formulário de centro de trabalho	135
Figura D.8 – Formulário de categoria	135
Figura D.9 – Formulário de grandeza	135
Figura D.10 – Formulário de unidade	136
Figura D.11 – Formulário de regra de seqüenciação	136
Figura D.12 – Formulário de verificação	136
Figura D.13 – Formulário de <i>lead-time</i> de matéria-prima	137

LISTA DE TABELAS

Tabela 2.1 – Quadro comparativo entre os diversos	8
Tabela 2.2 – Quadro comparativo de diferentes técnicas	29
Tabela 4.1 – Dados ilustrativos de disponibilidade de trabalho	60
Tabela 6.1 – Relações de precedência de dados	72
Tabela 6.2 – Descrição de relatórios	74

RESUMO

Sob constante pressão para redução de custos, pontualidade de entregas, aumento de produtividade e outros desafios, a indústria (ou qualquer outro ramo empresarial) deve ter em mente que isso precisa ser alcançado pela utilização efetiva de seus recursos, e não pelo excesso dos mesmos. É exatamente esta a função da *programação de produção*: alocar recursos a produtos de modo a retirar maior proveito!

Este é um problema bastante antigo e alvo de uma série de pesquisas. No entanto, ainda hoje organizações têm dificuldade em enfrenta-lo. Isso porque grande parte do conhecimento desenvolvido nesta área ou refere-se a problemas bastante específicos que não condizem com a realidade ou são relativamente complexos para aplicações no dia-a-dia.

Frente a este problema, o desenvolvimento de um sistema simples capaz de propor melhorias rápidas, ainda que não sejam ótimas, para micro e pequenas empresas é uma oportunidade pouco explorada!

O trabalho tem como produto principal o sistema desenvolvido que, quando parametrizado (informações de itens, operações, máquinas, etc.) gera um *programa de produção* a partir de ordens inseridas pelo usuário.

Este relatório escrito contém o embasamento teórico necessário e a documentação de como foi desenvolvido o sistema, premissas, limitações, etc. Ao final (anexos) constam dados e informações utilizadas pelo sistema.

1. INTRODUÇÃO

Cada dia mais a Logística assume um papel de vital importância para a manutenção da competitividade das empresas no mercado, sejam elas do ramo industrial, serviços ou qualquer outro. Sabemos também que os diversos “ramos” da Logística afetam as atividades de toda a organização e, geralmente, também são responsáveis por boa parte da percepção que o cliente recebe dos produtos.

Dentre estes diversos “ramos”, podemos citar o gerenciamento de estoques e prazos de entrega que são fortemente (porém não exclusivamente) derivados da adequação da **programação de produção**, que será o foco deste trabalho.

1.1 O PROBLEMA DE PROGRAMAÇÃO DE PRODUÇÃO

A administração de materiais (ou gerenciamento de estoques) é composta de diversas fases que se diferenciam pelo horizonte de planejamento e pelo grau de detalhamento em que o planejamento é feito.

Em um extremo, temos o *Planejamento Estratégico*, que serve para balizar ou direcionar as ações da empresa no longo prazo em um horizonte que pode chegar a 10 anos. A preocupação com o detalhamento operacional é tão baixa que esta etapa muitas vezes é elaborada em cifras monetárias (e não em quantidade física do produto).

No outro extremo, temos a programação de produção¹, que é a definição de *o quê*, *quando*, *quanto* e *onde* (e às vezes também *como*) cada item deverá ser fabricado. Ou seja, um programa de produção deve ser detalhado o suficiente para ser “realizável”, já

note no JF & JF

considerando as limitações de capacidade, disponibilidade de mão-de-obra, materiais, máquinas e facilidades. Trata-se de uma etapa "operacional" que, devido à precisão que apresenta, deve ter um horizonte de planejamento bastante curto, geralmente entre uma e três semanas. Em algumas indústrias de alto giro (como a de bebidas, por exemplo), a programação de produção chega a ser revisada diariamente para atender às necessidades de estoque e de mercado!

Assim, pode-se verificar que a programação deve apresentar flexibilidade compatível com as oscilações de mercado e, logicamente, com a tecnologia disponível na fábrica. Deve ainda contemplar necessidades de paradas para manutenção, variações da carga de trabalho (por exemplo por conta de um feriado), paradas não programadas (por exemplo por falta de energia), confiabilidade nos prazos de fornecedores, adequação às demais áreas da empresa, etc. Ou seja, por estes e por outros fatores a programação de produção torna-se uma tarefa complexa e que não apresenta "soluções gerais", tornando-se um desafio permanente!

No capítulo 2 deste trabalho (*Administração da Produção*) será apresentada uma descrição mais completa do processo logístico industrial e suas diversas etapas.

1.2 ESCOLHA DO TEMA

Quando da definição do tema do trabalho, o aluno estava em "transição" de estágios, estando portanto impossibilitado de escolher um tema diretamente ligado ao dia-a-dia e conseqüentemente em dificuldade de se realizar um trabalho essencialmente prático. Surgiu então a necessidade de escolha de um tema "genérico" ou teórico que

¹ Pode-se considerar também o *Controle da Produção* como uma etapa de nível "mais baixo" que a programação, porém este não se trata de planejamento

possibilitasse o desenvolvimento de um trabalho com *resultados reais* ou com um *produto final* que pudesse ser apresentado.

O tema *Programação de Produção* foi escolhido por se tratar de um problema bastante antigo encontrado pelas organizações – grandes ou pequenas – mas que ainda gera muito desconforto para qualquer gerente/diretor de Logística. Isso ocorre porque, apesar de muito estudado, ainda não existem soluções “elementares” para o problema.

Ou seja, pelas dificuldades citadas acima e por peculiaridades que cercam cada organização, a programação de produção pode tornar-se uma tarefa extremamente trabalhosa, demandando mão-de-obra qualificada e sendo, portanto, bastante dispendiosa (não se questionando os benefícios de uma programação bem feita!).

Dado este cenário, este tema foi escolhido por uma decisão pessoal do aluno no intuito de facilitar a compreensão do tema e execução do trabalho por parte dos responsáveis pela Logística em micro e pequenas empresas, onde muitas vezes este trabalho é feito por pessoas sem a habilidade e conhecimento necessários, valendo-se “apenas” a experiência do programador.

1.3 OBJETIVO DO TRABALHO

Conforme mencionado acima, é bem comum micro e pequenas empresas sofrerem com o problema de programação. Isto pode ser explicado por diversas razões, que convergem para o custo!

Ou seja, a utilização de mão-de-obra qualificada e com conhecimentos específicos demandaria maiores gastos com a folha de pagamento. Este gasto “extra” pode até render maiores economias, mas nem sempre é aceito por uma administração familiar. O mesmo resultado teria a análise de utilização de *softwares* comerciais.

Neste cenário, o objetivo deste trabalho é **desenvolver um sistema flexível para suportar a programação de produção de empresas de pequeno porte com diferentes características operacionais.**

A *flexibilidade* do sistema refere-se a capacidade de “adaptar-se” a diferentes processos (em linha, sob encomenda, etc.) com diferentes horizontes de programação (escala de tempo), *lead-time* em diferentes ordens de grandeza, etc.

Assim, este trabalho apresenta *dois produtos finais*: este *relatório escrito* e principalmente o *sistema de programação* (elaborado no MS-Access) que o acompanha, onde foi concentrado maior esforço e dedicação.

2. REVISÃO BIBLIOGRÁFICA: Administração da Produção

Sobre Administração da Produção serão apresentados os principais conceitos envolvidos relacionados principalmente ao gerenciamento de estoques e aos métodos utilizados para definição da “produção em função do tempo”. Ou seja, os diversos níveis de planejamento e suas respectivas características (função, horizonte, ...)

Para melhor compreensão dos tópicos, este capítulo foi dividido em:

- Introdução à Administração da Produção
- Gerenciamento de Estoques
- Previsão da demanda
- Planejamento Agregado
- MRP
- Dimensionamento de lotes
- Programação de Produção

Vale ressaltar que neste capítulo será apresentada apenas a abordagem principal de cada tópico para melhor compreensão do problema, não sendo intenção esgotar os temas.

2.1 INTRODUÇÃO À ADMINISTRAÇÃO DA PRODUÇÃO

A Administração da Produção consiste em procurar os melhores resultados de um conjunto de facilidades/instalações com o menor custo. Para tanto, dispõe-se de diversas “atividades foco”. Ou seja, atividades que podem garantir ou não o atingimento dos objetivos. Dentre estas, podemos citar: gerenciamento da capacidade produtiva, alocação de recursos, gerenciamento de estoques, planejamentos de curto e longo prazo, controle e aprimoramento de processos, suprimentos, etc.

Apenas com essa pequena amostra das atribuições da Administração da Produção, já é possível verificar que esta engloba as quatro fases do Ciclo PDCA². Este trabalho, no entanto, ficará restrito ao *planejamento*.

Mesmo com um parco conhecimento sobre processos e custos industriais, pode-se também perceber que a complexidade da Administração da Produção cresce à medida que a organização parte para qualquer tipo de diversificação, seja de produto, mercado, pontos de produção (fábricas), tecnologia, etc.

A fim de facilitar o entendimento dos próximos tópicos deste capítulo, serão introduzidos dois *frameworks*, apresentados por HAX (1984).

2.1.1 FRAMEWORK DE ANTHONY

Apresentado por ANTHONY (1965), classifica as decisões da Administração da Produção em três níveis:

Planejamento estratégico:

“Processo de decisão sobre os objetivos da organização, mudanças nestes objetivos, recursos utilizados para o atingimento destes objetivos e as políticas que governarão as aquisições, utilização e disposição destes recursos” (ANTHONY [1965] apud HAX [1984]).

Em outras palavras, o planejamento estratégico é a base que determinará a alocação de recursos “imobilizados” dentro da organização ao longo do tempo em um horizonte longo, ou seja, *planejamento das instalações*!

² PDCA: *Plan, Do, Check, Act* (Planejar, Fazer, Checar e Agir) – Conceito amplamente utilizado em Qualidade para controle e aprimoramento contínuo de processos

Controle gerencial:

“Processo pelo qual os gerentes asseguram que os recursos são obtidos e utilizados eficaz e eficientemente no cumprimento do objetivo da organização” (ANTHONY [1965] apud HAX [1984]).

Equivalentemente, trata do gerenciamento dos recursos “não-imobilizados” no medio prazo, como definição dos pontos de fabricação e armazenagem, direcionamento de verbas, etc. É neste nível que se elabora o *planejamento agregado*.

Controle operacional:

“Processo de assegurar que tarefas específicas são conduzidas eficaz e eficientemente” (ANTHONY [1965] apud HAX [1984]). Ou seja, este nível trabalha com um curto horizonte e alto nível de detalhamento (itens e operações desagregadas!)

Resumo do *Framework* de Anthony

Fica fácil perceber que a idéia central deste problema é dividir o planejamento das atividades em diferentes níveis de detalhamento/agregação, inversamente proporcionais ao horizonte de planejamento. Os três níveis podem ser resumidos na tabela 2.1 (HAX [1984]):

Finalizando, é imprescindível a percepção da necessidade de interação entre os diversos níveis, uma vez que um nível mais baixo vai “alimentar” o nível superior que balizará o nível mais baixo, formando assim um ciclo!

Fator	Planejamento estratégico	Controle gerencial	Controle operacional
Propósito	Direcionamento e alocação de recursos	Utilização de recursos	Execução, avaliação e controle
Instrumentos de implantação	Políticas, objetivos e investimento	Orçamentos	Procedimentos e relatórios
Horizonte de planejamento	Longo	Médio	Curto
Escopo	Amplo – nível corporativo	Médio – fábrica	Estreito – linha de produção
Nível da gerência / administração envolvida	Alto	Médio	Baixo
Frequência de replanejamento	Baixa	Média	Alta
Fonte de informação	Majoritariamente externa	Externa e interna	Majoritariamente interna
Grau de agregação da informação	Altamente agregado	Moderadamente agregado	Detalhado
Precisão necessária	Baixa	Média	Alta
Grau de incerteza	Alta	Média	Baixa
Grau de risco	Alta	Média	Baixo

Tabela 2.1 – Quadro comparativo entre os diversos níveis do *framework* de Anthony

Fonte: HAX (1984)

2.1.2 FRAMEWORK DA ESTRUTURA DO PRODUTO

Esta “visão” do produto é bastante útil para se entender a “vida” de um produto dentro de uma organização, que pode ser dividida em três fases:

- **Suprimentos:** representa as operações de aquisição de matérias-primas, materiais de embalagem, componentes, semi-acabados, peças de manutenção ou qualquer outro item de origem externa (comprado)
- **Produção:** incluindo fabricação e montagem, representa a transformação dos itens adquiridos por suprimentos, juntamente com outros recursos (mão-de-obra, energia,

maquinas, etc.), em produto acabado. Há uma importante distinção que pode ser feita com relação ao destino da produção:

- Make-to-order: produção sob encomenda, de modo que não há formação de estoques³
 - Make-to-stock: produção baseada em previsão de vendas, sendo necessária a formação de estoques, uma vez que as previsões não são perfeitas
- **Distribuição**: é a entrega do produto acabado ao cliente

As três fases acima representam um caso genérico para uma empresa, sendo que comumente uma (ou mais) pode não estar presente. Como consequência, isso pode gerar uma classificação com relação aos processos da organização: (BUFFA; TAUBERT [1972] apud HAX [1984], BUFFA; MILLER [1979] apud HAX [1984], JOHNSON; MONTGOMERY [1974] apud HAX [1984])

- **Estoque Puro**: Não inclui a fase de produção. Ou seja, a organização tem como “função” apenas colocar os produtos ao alcance dos clientes. Um exemplo típico é o comércio varejista.
- **Produção contínua**⁴: Produção de um ou poucos itens semelhantes, de modo que a linha de produção/montagem sofra nenhuma ou poucas alterações ao longo do tempo. O foco, nestes casos, é para o balanceamento de linhas (evitar “gargalos”). Por exemplo, uma indústria de bebidas que tenha uma linha exclusiva para um único produto.
- **Produção intermitente**: Ocorre quando há mudanças relativamente frequentes nas linhas de produção, sendo esta prática necessária geralmente para a cobertura de um *mix* complexo de produtos. Normalmente a produção intermitente implica a

³ Pode haver formação de estoque de produtos em processo nos centros de trabalho, mas não deveria ocorrer formação de estoque de produto acabado

⁴ Não necessariamente precisa ser uma operação ininterrupta, como as estações de tratamento de água

formação de *lotes*. Por exemplo, uma indústria de bebidas que produza em uma mesma linha diversos produtos diferentes, exigindo *set-ups* e *change-overs*.

- **Projetos:** Caso onde a fabricação/montagem/produção é feita uma única ou poucas vezes, sendo portanto de difícil padronização e com custos geralmente elevados. Como exemplo pode-se citar a construção civil (quando o projeto de um edifício é executado uma única vez).

2.2 GERENCIAMENTO DE ESTOQUES

Administrar estoques é um dos grandes desafios atuais da Logística. Por um lado, o mercado exige variabilidade de produtos e entregas cada vez mais rápidas (e confiáveis), o que incita altos níveis de estoque. Por outro, a pressão dos acionistas/investidores por retornos financeiros demanda estoques baixos (no limite, estoque zero!).

O gerenciamento de estoque, no entanto, não está restrito às duas observações acima! A seguir são apresentados resumidamente os principais conceitos relacionados ao gerenciamento de estoque:

2.2.1 CLASSIFICAÇÕES

Diversas classificações podem ser feitas para o inventário, com base nas mais variadas dimensões. Neste trabalho apresentaremos as duas mais comuns: ABC⁵ e Funcional.

Classificação ABC:

Esta é provavelmente a classificação mais conhecida devido à sua simplicidade e utilidade. Baseia-se na importância de cada item, de forma que os poucos itens que

perfazem grande parte do valor do estoque receberão atenção diferenciada (itens "A").

No outro oposto estão os itens C e em posição intermediária os itens B:

Funcional^o:

Pode-se citar inúmeras razões pelas quais uma empresa é obrigada a manter estoques, seja de matéria-prima, semi-acabados, produto acabado, itens de manutenção ou qualquer outro item. Estas razões, no entanto, podem ser agrupadas de modo que facilite a identificação do "problema" e permita "soluções" e tratamentos mais específicos adequados. Os *grupos funcionais* são:

- Estoque cíclico: Resulta de processos em lotes (ao invés de quantidade unitárias). Sua utilidade é para evitar ou "ratear" custos fixos de produção (como *set-ups* ou custo fixo de uma ordem de compra), de modo que o custo unitário seja reduzido. Logicamente, deve-se sempre procurar que o ganho no custo unitário do produto seja superior ao custo adicional proveniente do estoque
- Estoque de congestionamento: Necessário quando diversos itens "disputam" algum recurso e não podem ser produzidos simultaneamente. Assim, faz-se necessária a formação de estoques para suprir a demanda de todos os itens dado que não podem ser produzidos a qualquer momento (há formação de fila para utilização dos recursos)
- Estoque de segurança: Devido as incertezas das previsões (sempre acompanhadas por uma margem de erro), é preciso manter em estoque uma

^o Os "limites" dos intervalos para a classificação ABC não são fixos. Como ilustração, o aluno transcreveu os valores apresentados por HAX (1984)

^o Esta classificação foi retirada de SILVER et al. (1998). Alguns autores fazem uma classificação ligeiramente diferente mesclando alguns dos grupos apresentados

quantidade superior à demanda prevista para cobrir as oscilações aleatórias e garantir o fornecimento

- Estoque de antecipação: Utilizado quando há uma previsão de forte desbalanceamento entre capacidade de produção e demanda. Pode ser tanto por redução na capacidade (por exemplo, na indústria alimentícia que depende de produtos sazonais) como por aumento da demanda (por exemplo, aumento da venda de televisores em época de Copa do Mundo)
- Pipeline: Representa o estoque de produtos que estão em trânsito, seja na cadeia de distribuição, seja dentro da fábrica como semi-acabado, em quarentena, etc. Recebe este nome pela sua natureza de *fluxo*.

2.2.2 CUSTOS DE INVENTÁRIO

Freqüentemente quando se fala em estoque vem à cabeça a idéia de *custo*. Muito comumente, no entanto, pensa-se no custo de oportunidade como sendo a única fonte de ataques ao estoque. Deve-se lembrar também que estoques incorrem em uma série de outros custos, tais como:

- **Custo de oportunidade**: Representa o quanto a empresa poderia estar ganhando se o capital alocado em estoques estivesse investido em outra oportunidade. Apesar de não incorrer em dispêndio direto⁷ de dinheiro, deve ser considerado sempre que o valor do estoque for significativo.
- **Custo de movimentação e armazenagem**: Este custo engloba tanto o custo de oportunidade da área alocada para a armazenagem e de seus equipamentos como os custos “reais” de mão-de-obra, manutenção da área, utilização de equipamentos de

⁷ Pode acarretar dispêndio real caso a empresa esteja trabalhando com capital de terceiros

movimentação e armazenagem, etc. Este custo será ainda mais significativo quando se referir a itens que necessitam condições especiais de armazenagem (ex.: itens refrigerados)

- **Custos de perdas:** Principalmente sobre produtos perecíveis, item em estoque está susceptível a avarias ou perdas por diversos motivos. Entre eles: expiração do item, deterioração, roubo, dano físico, obsolescência, etc.
- **Outros:** Existem ainda outros custos que podem incorrer sobre os estoques, tais como seguros, impostos, etc.

2.3 PREVISÃO DA DEMANDA

Não é difícil imaginar a influência que a previsão da demanda exerce sobre o planejamento das operações de uma organização. STEVENSON (1996) atribui duas utilidades para a previsão: *planejamento do sistema* (longo prazo) e *planejamento do uso do sistema* (curto e médio prazo).

Os principais conceitos de previsão (que não é exclusivo para demanda) serão apresentados qualitativamente neste tópico: cálculos não serão abordados.

2.3.1 CARACTERÍSTICAS DE UMA PREVISÃO

Apesar da variedade de técnicas e métodos de previsão, todos apresentam algumas características em comum que são inerentes a qualquer afirmação sobre o futuro. SETEVENSON (1996) relaciona quatro:

1. Técnicas de previsão, em geral, assumem que as condições e variáveis do sistema continuarão no futuro com o mesmo comportamento do passado. Isto gera alguns riscos para administradores que confiam em sistemas automáticos e não questionam a alteração de alguma variável não prevista!

2. Previsões não são perfeitas. Ou seja, sempre estarão acompanhadas de um erro ou de uma incerteza, geralmente causadas por variações aleatórias
3. Previsões para grupos de itens tendem a ser mais precisas do que para itens individuais
4. A precisão da previsão decresce à medida que o horizonte aumenta, ou seja, previsões de curto prazo devem apresentar maior acuracidade

2.3.2 PASSOS

A elaboração de um modelo de previsão passa, segundo STEVENSON (1996) por 5 etapas:

- **Propósito da previsão:** Definir o objetivo da previsão permite determinar o grau de detalhe ou precisão necessários. Saber a demanda de um dado produto no próximo mês pode ser muito mais dispendioso do que saber se a demanda deste produto será superior a um dado patamar!
- **Horizonte:** Definir o horizonte da previsão pode ser uma decisão crítica, dado que a acuracidade dos resultados será afetada
- **Técnica:** Cada situação (dependendo dos objetivos, horizonte, disponibilidade de dados, estabilidade do sistema, ...) deve ser prevista com técnicas adequadas à realidade (algumas técnicas serão apresentadas no tópico seguinte)
- **Coleta e análise de dados:** Uma previsão deve ser feita com base em dados, que podem ter algumas premissas ou vies que devem ser considerados na análise
- **Modelagem:** Elaboração de um modelo adequado e com parâmetros “calibrados”

2.3.3 ABORDAGENS DA PREVISÃO

Assim como as variáveis no “mundo real”, uma previsão pode ser *qualitativa* ou *quantitativa*.

No primeiro caso, utiliza-se variáveis subjetivas ou de difícil mensuração numérica. STEVENSON (1996) denomina estas situações como “*Previsões baseados no julgamento e opinião*”⁸.

No segundo caso, as variáveis de *input* são numéricas com precisão satisfatória. A previsão pode ser feita a partir da “extrapolação” do comportamento passado da variável de interesse (baseadas em séries temporais) ou pela combinação de outras variáveis conhecidas (previsões associativas).

Previsões baseadas no julgamento e opinião

Muito utilizada em casos onde as variáveis de *input* não estão disponíveis ou não é possível elaborar um modelo adequado à situação. Nestes casos, deve prevalecer o julgamento e opinião de pessoas ligadas aos processos, mercados, cenários econômicos, etc. Como exemplo, STEVENSON (1996) cita executivos, força de vendas, pesquisa com clientes, consultores externos, gerentes, etc.

Previsões baseadas em séries temporais

Este método, como o nome diz, utiliza dados passados e assume que o comportamento seguirá a mesma tendência no futuro. STEVENSON (1996) enumera 5 “parcelas” que podem ser responsáveis pelas variações ao longo do tempo:

- **Tendência:** movimento gradual de longo prazo

⁸ Tradução pelo aluno. Originalmente: “Forecasts based on judgment and opinion”

- **Sazonalidade:** Variações regulares de curto e médio prazo, geralmente relacionadas com fatores climáticos, férias/feriados, etc.
- **Ciclos:** Variações repetitivas com período maior do que um ano, geralmente relacionadas com variações no ambiente, como a situação econômica ou política do país, por exemplo
- **Variações irregulares:** Picos ou vales fora do “padrão” provocados por singularidades (eventos não repetitivos). Quando identificadas, estas situações devem ser excluídas da série temporal para evitar “contaminação”
- **Variações aleatórias:** Por motivos diversos, praticamente qualquer variável estará sujeita a aleatoriedade, que se propaga para outras variáveis dependentes

Uma série de técnicas já foram desenvolvidas para previsão das três primeiras parcelas (as duas últimas são “imprevisíveis”). Dentre elas, pode-se citar a *média móvel*, *suavizamento exponencial*, etc.

Previsões associativas

Esta técnica procura a correlação da variável de interesse (que é desconhecida e por isso procura-se prevê-la) com outras variáveis conhecidas (ou que podem ser previstas satisfatoriamente). O “produto final” seria uma função $y = f(x_1, x_2, \dots, x_n)$, de modo que dados $x_1, x_2, \dots, x_n$ pode-se “saber” o valor de y .

O método mais comum para estas associações é a regressão linear, gerando uma função do tipo $y = a_0 + a_1x_1 + a_2x_2 + \dots + a_nx_n$. Outros meios mais complexos, como a regressão múltipla, podem ser utilizados quando a regressão linear não for satisfatória.

2.4 PLANEJAMENTO AGREGADO

“No espectro de planejamento de produção, *planejamento agregado* é planejamento de capacidade em médio prazo que tipicamente cobre um horizonte de 2 a 12 meses, apesar

de poder se estender a até 18 meses em algumas empresas. É particularmente útil para organizações sujeitas a sazonalidade ou outras flutuações de demanda ou capacidade. O objetivo do planejamento agregado é atingir um plano de produção que utilizará efetivamente os recursos da empresa para satisfazer a demanda esperada. Planejadores devem tomar decisões com relação às taxas de produção, nível de emprego e variações, nível de estoques e variações e terceirização”⁹

Em outras palavras, o *planejamento agregado* é a elaboração de um plano de produção de médio prazo cuja finalidade é “balizar” a tomada de decisões com relação à necessidade de “pequenas” variações na capacidade (como alteração do quadro de funcionários), alocação de produção em diferentes plantas, orçamento de custos variáveis, etc.

O termo *agregado* é utilizado por não considerar itens isoladamente. Ao contrário, o plano é feito com famílias de itens, uma vez que não há necessidade deste tipo de detalhamento. Para uma fábrica de automóveis, por exemplo, basta saber quantos veículos de determinada plataforma espera-se vender daqui a três meses, independentemente do modelo!

Para os cálculos de necessidade de capacidade produtiva normalmente utiliza-se um consumo padrão de recursos para cada família de produtos. Este consumo pode ser medido, por exemplo, em horas-homem ou horas-máquina. Assim, a partir do plano de produção e do consumo padrão pode-se determinar as necessidades de contratação, realocação de máquinas, elevação de estoques, etc.

⁹ STEVENSON (1996) – tradução pelo aluno

Logicamente, à medida que os períodos do plano de produção se aproximam torna-se necessário realizar a desagregação em itens. A esta desagregação dá-se o nome de *Master schedule*, que, Segundo STEVENSON (1996), é um levantamento das necessidades de fornecimento (e não de produção). A partir do *Master Schedule*, juntamente com outras informações (como inventário atual) elabora-se o *Master Production Schedule (MPS)* – ou *Programa Mestre de Produção* – que representa as reais necessidades de produção e tem um período menor do o planejamento de produção.

2.5 MRP

Já foram vistos, resumidamente, os principais conceitos para gerenciamento de estoque. Não adianta, no entanto, formular uma política de estoque considerando-se os itens independentemente quando o consumo de um é função de outro(s), como é o caso de empresas manufatureiras, onde o consumo de uma matéria-prima, por exemplo, é exclusivamente dependente da produção de um item semi-acabado, que por sua vez é dependente da produção de outro semi-acabado e assim sucessivamente, até um produto acabado, cuja demanda é dada pelo mercado e é desconhecida. Desta forma, podemos classificar os itens com relação à origem da demanda:

- **Demanda independente:** são aqueles cuja demanda é determinada pelo mercado ou por fatores que não estão sob controle da empresa, necessitando, portanto, de uma *previsão*.
- **Demanda dependente:** são aqueles cuja demanda é determinada pelo planejamento de produção de outros itens e portanto estão sob controle da empresa, de modo que as previsões dão lugar a cálculos determinísticos

Estes cálculos foram inicialmente desenvolvidos na década de 1960 e até hoje são de extensa utilização, sendo conhecidos como MRP – *Materials Requirements Planning* – ou seja, dado um planejamento de itens de demanda independente é possível calcular com precisão os materiais que serão necessários (e quando serão necessários) para a produção do item. É este conceito que será apresentado a seguir:

2.5.1 DEFINIÇÃO

Em toda a literatura, não há uma definição única para um conceito tão amplo como o MRP, ficando o aluno livre para dar sua própria definição a partir da apresentação de diversos autores: *MRP é o cálculo de necessidades de materiais (itens de demanda dependente) que se faz a partir do Plano Mestre de Produção, determinando-se uma função quantidade x tempo para cada item (ou família) em uma organização.*

Esta definição ficará mais clara ao final deste tópico com a apresentação de um exemplo numérico.

Nomenclatura

- **Lista de materiais:** É o conjunto de itens (e respectivas quantidades) necessários para a composição de outro item. Para efeitos de ilustração, pode-se pensar como os ingredientes para um determinado produto.
- **Level Coding:** Considerando-se que os “ingredientes” de um produto também podem ser compostos por outros “ingredientes”, verifica-se que a lista de materiais pode ser constituída por diversos níveis. Cada um destes níveis é denominado *level coding*.
- **Low Level Coding:** Como um item pode assumir diversos níveis, dependendo do pai a que está ligado, a utilização do *Low Level Coding* consiste em atribuir aos itens o código mais baixo que assumem em qualquer estrutura da qual participa. Esta prática permite, durante o levantamento da demanda de itens diferentes, assegurar

que um item pode ser verificado uma única vez (começando-se pelos itens de nível mais alto), pois necessariamente toda a necessidade dos itens pais já foram definidas (os pais estão sempre em nível mais alto).

- **Lead time:** É o tempo necessário para a produção de um item (ou de recebimento de uma mercadoria) a partir de sua ordem

2.5.2 O QUE ENVOLVE O MRP

SLACK et al. (1997) apresenta uma esquema bastante didático para ilustrar as entradas e saídas do MRP. A figura 2.1 é uma representação ligeiramente modificada, pois o aluno considerou mais adequado que as *Ordens de Compra* e *Ordens de Trabalho* fossem ligadas ao Plano de Materiais ao invés de estar diretamente ligadas ao *Planejamento de Necessidades de Materiais* (como no original)

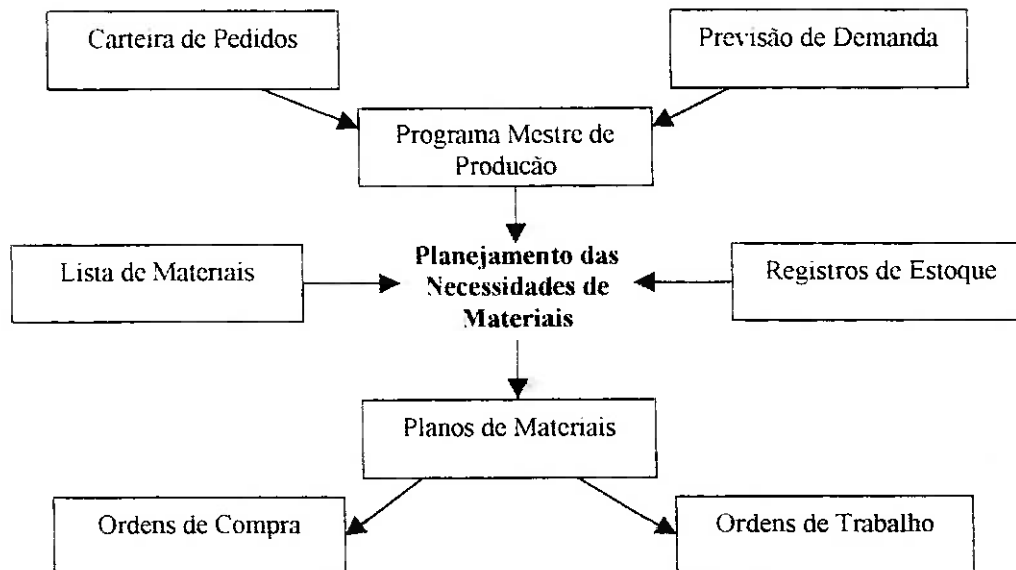


Figura 2.1 – Desenho Esquemático do planejamento de necessidades de materiais (MRP)

Fonte: SLACK et al. (1997), modificado pelo autor

- **Plano Mestre de Produção:** da maneira com foi apresentado no tópico anterior, o MPS é composto pelas necessidades “finais” da organização, ou seja, compõe-se da

carteira de pedidos somada à previsão de vendas. Logicamente, para a manufatura destes itens “finais” é necessário passar por outras etapas intermediárias, o que geraria um MPS com itens de semi-acabados ou mesmo de matérias-primas. É exatamente esta a função do MRP: gerar o MPS “completo”!

- **Lista de Materiais:** para se calcular as necessidades dos itens de demanda dependente, logicamente é necessário conhecer “de quem cada item depende”. Ou seja, é preciso conhecer os *ingredientes* e respectivas *quantidades* para a composição dos itens “independentes”. Por exemplo, para se montar uma determinada mesa são necessários 1 tampo, 4 pés e 4 travessas.
- **Registro de Estoque:** materiais que já estão em estoque também devem ser levados em consideração ao se levantar as necessidades dos itens de demanda dependente, uma vez que a necessidade de produção logicamente será menor se parte dos materiais já estiver disponível.
- **Plano de Materiais:** é o produto “chave” do MRP, do qual pode-se tirar outros subprodutos de grande importância para administração da produção: *ordens de compra e ordens de trabalho*.

2.5.3 FORMAS DE ESTRUTURA DE PRODUTO

Esta não é uma classificação comumente encontrada na literatura, mas o aluno achou bastante interessante as “formas” de estrutura de produto apresentadas por SLACK et al. (1997):

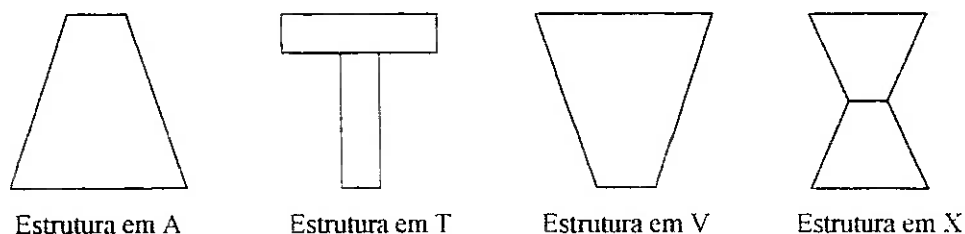


Figura 2.2 – Diferentes formas de estrutura de produto

Fonte: SLACK et al. (1997)

- **Estrutura em A:** poucos produtos finais produzidos/montados a partir de uma grande variedade de itens de nível inferior
- **Estrutura em T:** pequena variedade de matérias-primas utilizadas em processos padronizados com alto grau de diferenciação em etapas finais (ex.: gráfica que produz embalagens de apresentação – necessariamente “personalizadas”)
- **Estrutura em V:** semelhante a estrutura em T, porém com processos menos padronizados, de modo que a diferenciação começa a ser feita em níveis inferiores
- **Estrutura em X:** a empresa para de uma grande gama de matérias-primas produzindo um número relativamente baixo de componentes os quais podem ser montados de diversas maneiras, gerando uma novamente uma grande gama de produtos acabados.

2.5.4 CÁLCULO MRP

O cálculo MRP foi muito bem ilustrado por SLACK et al. (1997):

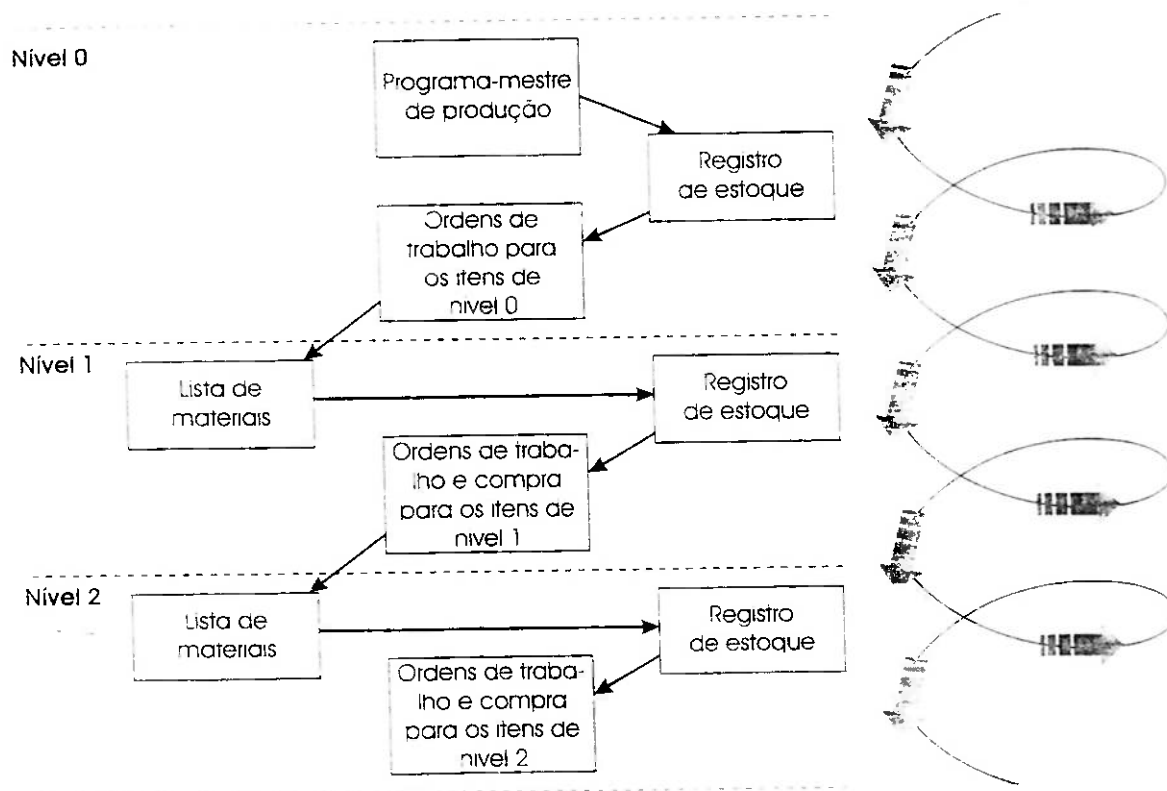


Figura 2.3 – Cálculo de necessidades líquidas no MRP

Fonte: SLACK et al. (1997)

Da figura 2.3 pode-se perceber que a programação inicia-se no nível 0 e vai “regredindo” no tempo, uma vez que os itens nível 1 têm que ser produzidos antes dos itens de nível 0. Por este motivo (retornar no tempo) o MRP compõe uma técnica denominada *Programação para Trás!*

2.5.5 MRP DE CICLO FECHADO

O leitor deve ter percebido que o cálculo MRP, como mostrado, assume capacidade “infinita de produção”, ou seja, se determinada demanda gerou a necessidade de 1.000 unidades de um dado item em um dado período, nenhuma verificação foi feita a respeito da viabilidade de produção destes 1.000 itens!

Com base nesta observação, foi criado o *MRP de ciclo fechado*, que se baseia também nos seguintes conceitos:

- **Planejamento de recursos:** plano realizado a longo prazo para comparação da capacidade atual com o planejamento de produção, podendo-se definir a necessidade de novos investimentos (ou mesmo desinvestimentos!)
- **Planos de capacidade agregada:** *RC'CP (Rough-Cut Capacity Plans)* – equivale a um corte “grosseiro” no MPS com base na capacidade atual instalada
- **Planos de Necessidade de Capacidade:** *CRP (Capacity Requirements Plans)* – com um nível de detalhamento maior sobre a produção, é feito um planejamento sobre as necessidades futuras de capacidade

A figura 2.4, retirada de VOLLMAN et al. (1992) apud SLACK (1997) representa a interações entre o MRP “simples” e as limitações de capacidade

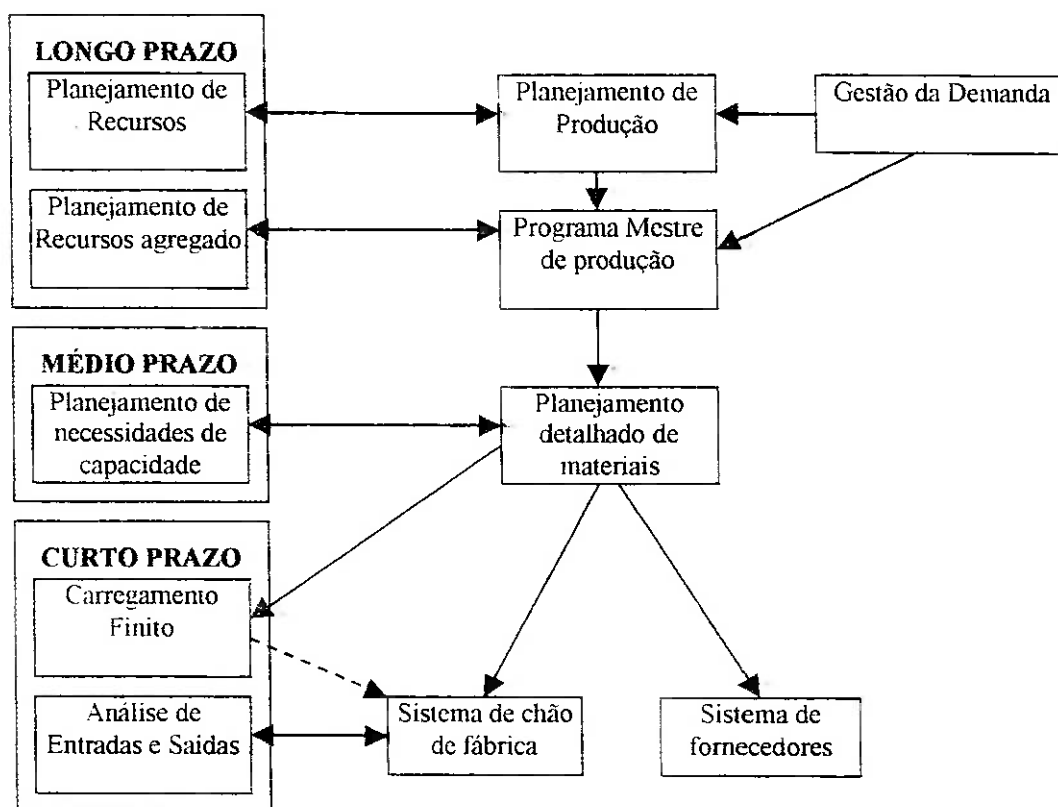


Figura 2.4 – MRP de ciclo fechado

Fonte: VOLLMAN et al. (1992) apud SLACK et al. (1997)

2.5.6 OUTROS CONCEITOS AVANÇADOS

- Manufacturing Resource Planning (MRP II):** Mais desenvolvido do que o MRP é o MRP II. Este outro conceito engloba o MRP (também chamado de MRP 1 por alguns autores) além de outras funções e áreas da empresa. Segundo WIGHT (1984) apud SLACK (1997), MRPII é “um plano global para o planejamento e monitoramento de todos os recursos de uma empresa de manufatura: manufatura, marketing, finanças e engenharia. Tecnicamente, ele envolve a utilização do MRP de ciclo fechado para gerar números financeiros”
- Optimized Production Technology (OPT):** O OPT é uma “técnica” utilizada principalmente no projeto da capacidade e balanceamento de linhas através do foco nos “gargalos” da produção, com base na Teoria das Restrições. Para melhor ilustrar o “fundamento” do OPT, abaixo estão transcritos alguns dos princípios retirados de SLACK (1997):

- Uma hora perdida num recurso gargalo é uma hora perdida para sempre em todo o sistema
- Uma hora poupada num recurso não-gargalo é uma miragem
- **Just-in-Time (JIT):** “Filosofia” de administração e práticas gerenciais cujo principal conceito é a produção para entrega no momento da necessidade/consumo (alguns autores traduzem *Just-in-Time* como “*Bem na hora*”). Esta abordagem da produção tem como objetivo a redução dos custos (de inventário e outros) e melhoria contínua dos processos e utiliza como “ferramenta” a redução de estoques que freqüentemente camufla os problemas da organização (CORRÊA [1995]). Alguns aspectos de considerável importância na filosofia JIT são (CORRÊA [1995]):
 - Produção sem estoques
 - Eliminação de desperdícios
 - Manufatura de fluxo contínuo
 - Esforço contínuo na resolução de problemas
 - Melhoria contínua dos processos

2.5.7 CRÍTICAS AO MRP

Apesar de ser uma ferramenta bastante poderosa e vastamente utilizada no mercado, o MRP é comumente alvo de críticas por assumir algumas premissas que poucas vezes são inteiramente verdadeiras. A listagem abaixo foi retirada de SILVER et al. (1998):

- **Lead times:** O MRP assume que o lead time é único para o item. No entanto, não é difícil supor que o *lead-time* pode variar, para um mesmo item, de acordo com as condições momentâneas do centro de trabalho (tempo de espera em fila, paradas não programadas para manutenção, etc.) ou com o tamanho do lote. A solução mais encontrada na prática é superestimar este tempo para evitar atrasos, o que pode, por

outro lado, aumentar o nível de estoques intermediários (ou mesmo “prejudicar” outros itens!)

- **Tamanho dos lotes:** O MRP não oferece nenhuma ferramenta para o dimensionamento de lotes
- **Estoques de segurança**¹⁰: O nível de estoque de segurança deve ser inserido para cada item e, assim como o *lead-time*, pode ser superestimado
- **Incentivos para aprimoramento:** Pelas próprias dificuldades de parametrização de um sistema MRP (por exemplo o *lead-time* e o estoque de segurança), após a estabilização das operações da empresa dificilmente alguém se “arrisca” a mudar em busca de aprimoramentos na gestão.

2.6 DIMENSIONAMENTO DE LOTES

Ao se desenvolver um programa de produção a partir do MPS, um dos principais problemas é o dimensionamento de lotes. Por um lado, lotes grandes podem elevar excessivamente o estoque (o que implica em custos) e sobrecarregar desnecessariamente um centro de trabalho, prejudicando o agendamento de outros itens. Por outro lado, lotes pequenos podem incorrer um grande número de *set-ups* que, em geral, é extremamente prejudicial para a empresa.

Neste tópico serão apresentadas as técnicas mais usuais para dimensionamento de lotes (retiradas de ORLICKY [1975]) e algumas considerações necessárias sobre ajustes nos tamanhos dos lotes.

¹⁰ Não há um consenso entre os autores sobre a necessidade de estoques de segurança para itens de demanda dependente. Podemos citar, por exemplo, ORLICKY (1975) apud HAX (1984), que defende esta prática apenas para itens de demanda independente, enquanto MEAL (1975) apud HAX (1984) a estende também para itens de demanda dependente

2.6.1 TÉCNICAS DE DIMENSIONAMENTO DE LOTES

Por não se tratar do foco deste trabalho, assim como todo o conteúdo teórico apresentado, o aluno fará apenas uma introdução qualitativa das técnicas de dimensionamento de lotes, sem maiores preocupações com fórmulas, cálculos ou demonstrações. As técnicas serão melhor compreendidas no próximo item, onde são apresentados exemplos numéricos para efeito de comparação.

É importante lembrar que estas técnicas podem ser utilizadas tanto para dimensionamento de lotes de produção como de lotes de compra. No segundo caso, deve-se entender como *set-up* os custos fixos de uma ordem de compra (ex.: frete, custo do comprador, etc.)

- **Fixed Order Quantity:** *Ordem de quantidade fixa* – como o próprio nome indica, os lotes têm tamanho fixo pré-determinado, geralmente por critérios empíricos, sendo variável o intervalo entre ordens. Esta prática, apesar da simplicidade, não é muito recomendada pois pode gerar uma série de incoerências, como carregamento de *estoque remanescente* (insuficiente para cobrir a próxima necessidade) ou mesmo excesso de *set-ups*.
- **Economic Order Quantity (EOQ):** *Ordem de quantidade econômica* – também trabalha com quantidades fixas (gerando também estoque remanescente), porém o tamanho do lote é determinado com base na previsão da demanda futura (ou MPS) na tentativa de minimizar o total de custos de *set-up* e de inventário (utilizando a idéia de que neste ponto o custo de *set-up* é igual ao custo de inventário)
- **Lof-for-lot ordering:** *Lote por lote* – aconselhada somente quando o custo de *set-up* for muito baixo em relação ao custo de inventário, consiste sempre em ordenar apenas a quantidade necessária para o momento.
- **Fixed Period Requirements:** *Necessidades por período fixo* – determina-se um período fixo para o qual cada lote deverá atender. Novamente, a escolha deste período é feita de maneira empírica, sendo variável a quantidade da ordem.

- **Period Order Quantity (POQ):** *Ordem de quantidade por período* – baseado na técnica EOQ, porém com uma adaptação para demanda em períodos discretos. Ou seja, faz os mesmos cálculos da EOQ, porém ajusta-se os resultados para evitar estoque remanescente. Desta forma, o número de *set-ups* será o mesmo, porém o custo de inventário será necessariamente menor.
- **Least Unit Cost (LUC):** *Menor custo unitário* – esta técnica também tem como objetivo minimizar o custo total (*set-up* e inventário). Para tanto, parte de uma quantidade inicial (equivalente à necessidade do primeiro período) e vai agregando as necessidades dos períodos seguintes, sempre verificando o custo total por unidade. A partir do momento que este custo começa a subir, um lote é definido e o processo se reinicia para os períodos seguintes
- **Least Total Cost (LTC):** *Menor custo total* – assim como a EOQ, baseia-se na idéia de que o menor custo se dará quando o custo de *set-up* se aproximar ao máximo do custo de inventário, porém sempre evitando estoque remanescente. Esta técnica utiliza como ferramenta uma medida adimensional chamada EPP (*Economic part-period*), ou “Parte-Período”¹¹ Econômico”, que indica quantos partes-períodos de estoque equivalem um *set-up*. Assim, semelhante ao LUC, vai-se agregando necessidades, porém a verificação é feita comparando-se o número de partes-períodos acumulados a EPP.
- **Part-Period Balancing (PPB):** *Balanceamento de Partes-Período* – utiliza a mesma lógica do LUC, porém realiza dois testes para verificação do resultado inicialmente obtido:
 - *Look-ahead test:* Agrega, quando possível, a necessidade do próximo período e verifica o custo total. Em algumas situações perceber-se-á que é preferível aumentar o estoque em determinado momento e adiar a próxima ordem!

¹¹ Como parte-período entende-se carregar inventário de uma unidade por um período (ex.: kg x semana, litro x mês, ...)

- *Look-back test*: Inverso do teste anterior, retira do lote a última necessidade agregada. Em algumas situações perceber-se-á que é preferível antecipar a próxima ordem por redução de estoque.
- **Wagner-Whitin Algorithm:** *Algoritmo de Wagner-Whitin* – Esta técnica baseia-se em um problema de programação dinâmica e é a única, dentre as apresentadas, que gera uma solução ótima¹². No entanto, a dificuldade de se trabalhar com esta técnica a torna bastante limitada, uma vez que usuários de sistemas MRP muitas vezes não têm conhecimento dos conceitos de programação dinâmica.

2.6.2 EXEMPLOS NUMÉRICOS E COMPARAÇÃO

O exemplo apresentado abaixo foi retirado de HAX (1984) e considera os seguintes parâmetros:

- Custo de *set-up* = \$100,00
- Custo de inventário de uma unidade por um período = \$1,00

Período	1	2	3	4	5	6	7	8	9	Total	Custo
Método	35	10		40		20	5	10	30	150	Total
Ordem de quantidade fixa ¹³	60			60					60	180	\$ 480
Ordem de quantidade econômica	58			58				58		174	\$ 506
Lote por lote	35	10		40		20	5	10	30	150	\$ 700
Necessidades por período fixo ¹⁴	45			40		25		40		150	\$ 445
Ordem de quantidade por período	85					35			30	150	\$ 455
Menor custo unitário	45			60			45			150	\$ 420
Menor custo total	85					65				150	\$ 445
Balanceamento de partes-período	85					65				150	\$ 445
Algoritmo de Wagner-Within	45			65				40		150	\$ 395

Tabela 2.2 – Quadro comparativo de diferentes técnicas de dimensionamento de lotes

Fonte: HAX (1984)

¹² Considerando-se apenas o horizonte planejado, do qual se dispõe de previsão de demanda. Nada garante que a solução será ótima após a inserção de novos dados de períodos futuros.

¹³ O valor de 60 unidades foi escolhido aleatoriamente para ilustração

¹⁴ O prazo de 2 períodos foi escolhido aleatoriamente para ilustração

É importante observar que estes valores são apenas ilustrativos e não são suficientes para qualquer tipo de conclusão sobre qual a melhor técnica. Caso algum parâmetro de custo ou a demanda de algum período fosse alterada, a relação entre os custos totais das diversas técnicas também poderiam se alterar.

2.6.3 AJUSTES NO DIMENSIONAMENTO DE LOTES

- **Quantidade mínima e máxima:** Não é difícil encontrar processos que apresentem limitações técnicas (ou mesmo econômicas) quanto a quantidade que pode ser trabalhada. Em uma panificadora, por exemplo, há um limite de pães por fornada que não pode ser excedido.
- **Quantidade múltipla:** Existem também processos que somente podem ser realizados em quantidades múltiplas. É o caso de uma linha de embalagem que prepara pacotes múltiplos com 12 unidades, por exemplo.
- **Indivisibilidade da matéria-prima:** Da mesma forma que existe a limitação de quantidades múltiplas por características técnicas dos processos, um caso particular seria a dificuldade de separação de quantidade fracionadas de matéria-prima. Por exemplo, se um determinado produto em comprado em tambores, pode não ser muito adequado determinar um lote que consuma 3,8 tambores da matéria-prima.
- **Perdas de processo:** Dado que é comum ocorrerem perdas nos processo industriais, pode-se desejar que uma quantidade maior de matéria-prima seja alocada para determinado lote.

2.7 PROGRAMAÇÃO DE PRODUÇÃO

Até agora foram apresentados conceitos e técnicas para realização de um planejamento agregado e até mesmo para a determinação da dimensão dos lotes de produção, muitas vezes sem preocupação com a capacidade ou viabilidade dos planos.

Na fase de programação, no entanto, deve-se fazer o maior detalhamento possível de modo que o programa possa ser executável. Ou seja, um *programa de produção* deve conter informação suficiente para que um operário do chão de fábrica seja capaz de saber exatamente o que deve fazer. Desta forma, a programação não faz referência a *períodos* de tempo como o planejamento agregado. Ao contrário, deve determinar precisamente em que dia e horário determinado item deve ser produzido, em que quantidade e em que máquina/recurso. De acordo com *framework* de Anthony, esta é a etapa operacional (decisões detalhadas e de curto prazo)!

2.7.1 INTRODUÇÃO

Retomando-se o *framework da estrutura do produto* apresentado no tópico 2.1.2 deste trabalho, ressalta-se novamente os quatro tipos básicos de processos de produção: estoque puro, produção contínua, produção intermitente e projetos. O problema de programação resolve-se por diferentes modelos em cada um destes processos:

- **Estoque puro:** não há atividade específica de programação (apenas planejamento e controle de estoques)
- **Produção contínua:** balanceamento de linha
- **Produção intermitente:** sequenciação
- **Projetos:** programação em redes

Sistemas de *estoque puro* não têm necessidades específicas de programação porque assume-se que a empresa tem capacidade “infinita”. Ou seja, apesar de um comprador não ser capaz de realizar vários pedidos simultaneamente, o tempo gasto por compra é relativamente pequeno se comparado, por exemplo, ao tempo de entrega (o “verdadeiro” *lead-time*)

Já o *balanceamento de linha*, segundo HAX (1984), é um problema “pontual”. Ou seja, uma vez projetada e montada a linha, o agendamento de atividades passa a ser relativamente simples. Isto porque toda a linha deverá funcionar “harmonicamente” como se fosse uma única máquina.

A *sequenciação*, dependendo da configuração do sistema produtivo e das características dos processos, pode gerar desde problemas elementares a problemas com soluções bastante complexas. Dado um conjunto de máquinas (*job shop*) e um conjunto de operações (com suas respectivas restrições) que precisam ser realizadas, o problema consiste em priorizar os elementos que estarão “disputando” a capacidade das máquinas (*gerando filas*).

Já os *projetos* são tratados como um caso particular de produção intermitente, sem haver, no entanto, a padronização das atividades. Nestes casos, o problema consiste na alocação dos recursos ao longo do tempo para os diversos “consumidores”.

2.7.2 SEQUENCIAÇÃO

Talvez a maior dificuldade encontrada na resolução de problemas genéricos de sequenciação seja sua natureza combinatória. Ou seja, a diversidade de roteiros que os materiais podem percorrer chega a gerar um problema com $(n!)^m$ soluções, onde n representa o número de trabalhos para serem realizados e m é o número de máquinas. Logicamente, restrições aos problemas reais (como precedência de operações) limita as soluções viáveis, mas, em geral, ainda assim este número permanece bastante grande para se encontrar uma solução por procura exaustiva.

Percebe-se que esta dificuldade pode tornar-se bem mais tênue quando a configuração do centro de trabalho e as restrições das ordens de produção assumem algumas

condições específicas. Neste trabalho serão apresentadas algumas soluções para os tipos mais comuns de configuração: uma única máquina, máquinas em paralelo, máquinas em fluxo, centro de trabalho genérico

Nomenclatura e Definições

Na literatura especializada há relativa diversidade de definições e nomenclaturas. Neste trabalho serão utilizadas as definições¹⁵ de CONWAY et al. (1967) apud HAX (1984), com algumas alterações propostas pelo aluno.

- **Operação:** Tarefa “homogênea” que deve ser realizada
- **Tempo de operação**¹⁶: Tempo requerido para realização de uma operação. Nos casos em que o tempo de *set-up* for independente da operação anterior na máquina (ou seja, o tempo de *set-up* é constante), este pode ser incluso no tempo de operação
 - **Ordem:** Conjunto de operações inter-relacionadas por restrições de precedência
 - **Roteiro:** Conjunto de restrições de precedência
 - **Máquina:** Equipamento ou recurso capaz de realizar uma operação
 - **Data de chegada:** Momento em que uma ordem chega ao centro. Representa o momento em que a primeira operação da ordem pode ser realizada
 - **Data de entrega:** É a data em que o produto deve estar pronto para ser despachado/entregue ao cliente
 - **Data de término:** Data a partir da qual o produto está disponível para ser despachado/entregue ao cliente
 - **Programa de Produção:** Programação detalhada que deve ser cumprida para atingimento dos níveis desejados de produção

¹⁵ Traduções pelo aluno

¹⁶ Em CONWAY (1967) apud HAX (1984) a nomenclatura utilizada é *processing time*. O aluno julga a expressão *Tempo de Operação* mais adequada para manter coerência com o termo *Operação*

Open shop x Close shop

Uma primeira classificação que pode ser feita aos *job shops* é com relação à origem do planejamento:

- Open shop: Produção sob encomenda (*make-to-order*). Como nestes casos não há previsão de vendas, a previsão realizada é com relação à demanda do centro de trabalho (geralmente em horas de trabalho). Desta forma tenta-se prevenir sobrecarga ou ociosidade excessiva.
- Close shop: Produção para estocagem (*make-to-stock*). Neste caso, a previsão é um pouco mais complexa e tem como foco os produtos e quantidades (de onde pode-se também obter a previsão da carga de trabalho).

Medidas de desempenho

Como qualquer outra atividade gerencial em uma empresa deve-se estabelecer indicadores e critérios para avaliação dos planos. Com a programação não é diferente e os indicadores podem ter diferentes objetivos, avaliando o desempenho tanto das ordens como dos centros produtivos:

- **Desempenho das ordens**: Normalmente o que se deseja controlar é o cumprimento do prazo. Assim, os indicadores mais comum são com relação a atraso, podendo estar baseados em média, máximo, número de atrasos, etc.
- **Desempenho dos centros de trabalho**: Simultaneamente deve-se controlar o estado de *saturação* ou *ociosidade* dos centros de trabalho. Bons indicadores podem ser obtidos através do nível dos estoques *pipeline*, número de ordens em fila, tempo médio em fila, etc.

Os indicadores acima foram apresentados de forma genérica, sem que isto impeça que determinadas organizações necessitem desenvolver critérios específicos para suas operações.

Premissas

Assim como as definições e nomenclaturas, as premissas utilizadas nos modelos apresentados na literatura são bastante diversificados. Novamente será utilizado como base as premissas de CONWAY et al. (1967) apud HAX (1984):

1. As máquinas estão “inteiramente” disponíveis para programação, não havendo previsão de paradas para manutenção, quebras, feriados, etc.
2. As ordens representam um conjunto de operações ordenadas, ou seja, os roteiros das devem ser respeitados
3. Há uma e apenas uma máquina que pode realizar determinada operação
4. Cada máquina pode realizar no máximo uma operação de cada vez
5. Uma operação não pode ser interrompida antes de sua conclusão para realização de outra operação
6. Uma operação somente pode ser iniciada após a conclusão da operação que a precede no roteiro
7. Há apenas um recurso limitante em cada centro de trabalho: a máquina. Os demais recursos são considerados abundantes

Métodos de seqüenciação

Pela importância do problema de seqüenciação, uma série de métodos já foram desenvolvidos, de variada complexidade, sendo alguns aplicados a casos específicos. Seguem alguns dos métodos enumerados por PACHECO; SANTORO (1999):

- **Enumeração explícita:** geração exaustiva de possibilidades
- **Modelos de programação linear:** através de modelos que utilizam variáveis inteiras e contínuas, buscam soluções ótimas
- **Branch & Bound:** utiliza estrutura de busca em árvore, de modo que os nós são “qualificados” ou “descartados” à medida que o problema vai se resolvendo

- **Programação dinâmica:** utiliza a programação dinâmica para a resolução de problema multi-estágio, procurando otimização
- **Algoritmos eficientes:** são métodos que podem gerar soluções ótimas a partir de um “procedimento”, que geralmente se aplica a casos específicos
- **Seqüências pré-defnidas:** como o nome diz, a programação das máquinas obedece uma seqüência obrigatória, definida no instante inicial da programação
- **Regras de liberação:** Simulação de seqüenciação das operações em cada máquina com base em regras de priorização dos elementos nas *filas*. Este é um método atualmente bastante utilizado por possibilitar grande flexibilidade e gerar resultados viáveis e “bons” (nem sempre ótimos)
- **Redes neurais:** Baseados em sistemas de inteligência artificial, geram regras de decisão a partir de entradas e saídas conhecidas, simulando o aprendizado da mente humana
- **Métodos de Gargalo:** métodos heurísticos concentrados nos recursos gargalo.

Centros com uma máquina

Caso específico em que uma única máquina deve realizar diversas operações que estão em fila, onde o trabalho de seqüenciação consiste na priorização das ordens

Em algumas situações, pode ser possível, interessante ou mesmo necessário interromper o processamento de uma ordem antes de sua conclusão para o início de outra. A esta prática dá-se o nome *preemption*. Segundo HAX (1984), uma operação é chamada *preempt-resume* quando o tempo total de processamento independente do número de interrupções feitas. *Preempt-repeat* são aquelas que precisam ser reiniciadas após cada interrupção.

Uma outra técnica que pode ser utilizada nestes modelos são os programas com tempo ocioso inserido¹⁷. Nestes casos a máquina fica ociosa mesmo havendo ordens na fila, pois uma outra ordem de alta prioridade deve estar “chegando”.

HAX (1984) apresenta algumas regras de seqüenciação adequadas para determinados objetivos. As mais comumente utilizadas são *menor tempo de processo* – que minimiza o tempo médio do fluxo – e *data de entrega mais próxima* – que minimiza o maior atraso.

Centros com máquinas em paralelo

Neste caso, há mais do que uma máquina capaz de realizar as operações. No caso de uma única máquina, comumente o *makespan* não é relevante, pois é equivalente à soma dos tempos de operação (não há como otimizar). Havendo mais que uma máquina, no entanto, a situação se inverte, uma vez a sobrecarga de uma máquina (e ociosidade das outras) pode gerar um gargalo para o conjunto!

Pela importância do equilíbrio entre as diversas máquinas, alguns algoritmos heurísticos já foram desenvolvidos para que se realize uma alocação adequada das ordens. Como exemplo, pode-se citar a seqüenciação pelo *maior tempo de processo*, alocando as ordens às máquinas em ordem decrescente do tempo de processo, sempre na máquina que tiver menor tempo já programado

Centros com máquinas em fluxo

Nestes casos, tem-se *m* máquinas por onde as ordens de produção devem passar, respeitando uma dada seqüência, formando uma linha. Não é obrigatório, no entanto,

¹⁷ HAX (1996). Do inglês, *schedules with inserted idle time* (tradução pelo aluno)

que todas as ordens passem por todas as máquinas, ou equivalentemente, passar por alguma(s) com tempo zero.

Esta configuração aparentemente simples pode tornar-se um problema extremamente complexo conforme m aumenta. Nos casos em que $m = 2$ ou $m = 3$, o algoritmo de Johnson gera bons resultados. A partir deste limite, métodos mais complexos devem ser utilizados.

Centros de configuração genérica

Centros de configuração genérica são “caracterizados por permitir diferentes fluxos das ordens entre as máquinas e diferentes números de operações por ordem, que são processadas apenas uma vez em cada máquina” (PACHECO; SANTORO [1999]). Por conta desta “generalização”, não há um método geral de solução.

Na literatura são abundantes as classificações de *job shops* (como são chamados os centros de configuração genérica), muitas vezes classificações “independentes”. PACHECO; SANTORO (1991) propõem uma abordagem hierarquizada (em níveis), de forma que cada um método de sequenciação (como os apresentados no item 2.7.2 deste texto). A título de ilustração, a figura 2.5 reproduz o “início” da árvore de classificações:

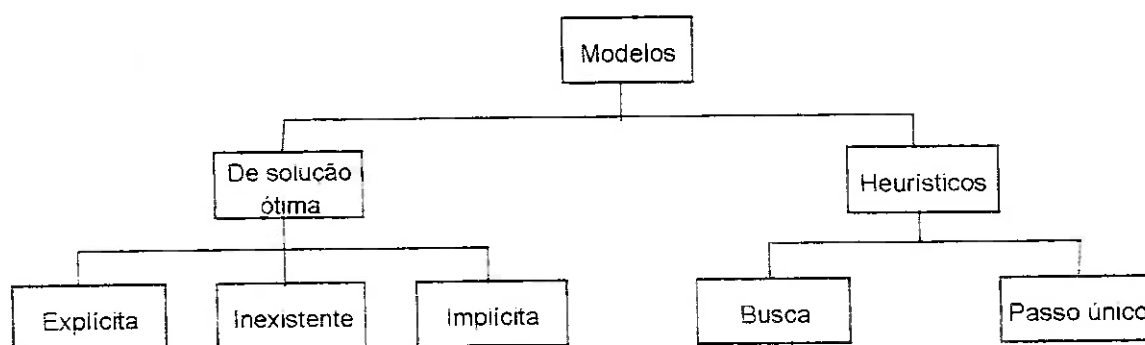


Figura 2.5 – Classificação hierarquizada de *job shops* - exemplo

3. REVISÃO BIBLIOGRÁFICA: Outros conceitos

Neste capítulo serão abordados dois tópicos de relevante participação no desenvolvimento do trabalho, apesar de representarem ferramentas e não o foco (como os conceitos de Administração da Produção): *simulação e projeto de banco de dados*¹⁸.

3.1 MODELOS E SIMULAÇÃO

Este tópico tem como finalidade apresentar ao leitor alguns dos principais conceitos sobre modelos e simulação, principalmente no que tange à classificação e estruturação dos modelos.

3.1.1 SISTEMAS E MODELOS

Para melhor compreensão do que são modelos, é interessante entender como LAW: KELTON (1991) classifica as técnicas para a compreensão e estudo de um problema real, iniciando-se por duas definições:

- **Sistema:** “Coleção de entidades, isto é, pessoas ou máquinas, que atuam e interagem juntos buscando o atingimento de um fim lógico” (SCHMIDT; TAYLOR [1970]¹⁹ apud LAW; KELTON [1991])
- **Modelo:** Conjunto de premissas e relações elaboradas para permitir a estruturação do comportamento de um sistema

Alternativamente, pode-se dizer que um *modelo é uma simplificação da realidade, sem perda das características principais do sistema*. Desta forma, não é difícil imaginar que à medida que os sistemas tornam-se mais complexos, os modelos vão se distanciando da

¹⁸ Neste tópico também serão apresentadas algumas considerações sobre projeto de *softwares*

realidade (caso contrário perderiam sua finalidade de *simplificação*). A partir deste raciocínio, pode-se classificar os métodos de modelagem de sistemas conforme LAW; KELTON (1991):

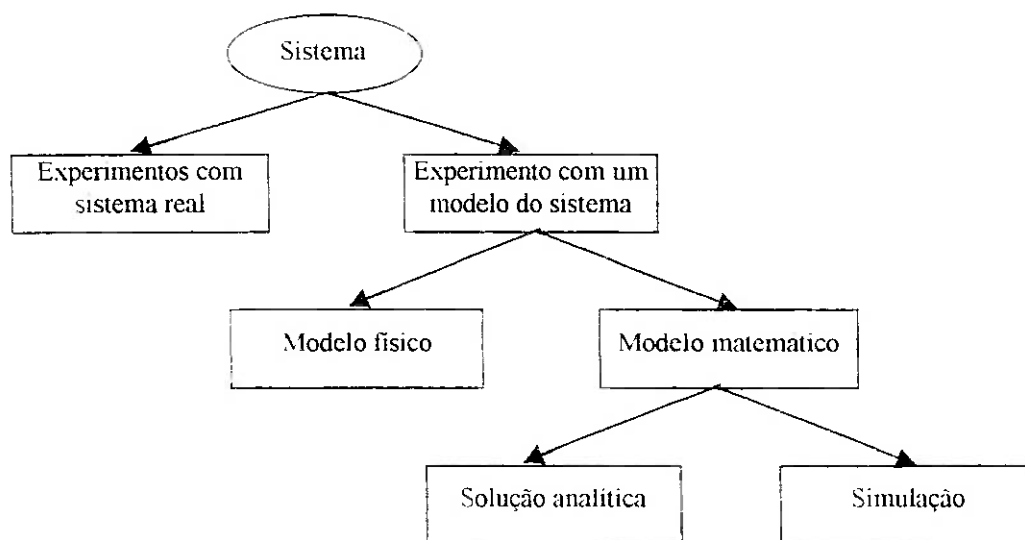


Figura 3.1 – Meios de estudar um problema

Fonte: LAW; KELTON (1991)

Da figura, podemos retirar:

- **Experimento com sistema real x Experimento com modelo do sistema:** Em algumas situações pode ser viável²⁰ estudar o problema através de experimentos (alterações nas condições) no sistema real. A grande vantagem deste método é a confiabilidade do resultado. No entanto, as condições de viabilidade raramente são atingidas, tornando necessários experimentos com um modelo do sistema.
- **Modelo físico x Modelo matemático:** Uma representação física do modelo (geralmente em escala reduzida), também chamado de *protótipo* ou *piloto*, pode também apresentar soluções bastante confiáveis, mas são de aplicação a sistemas

¹⁹ Tradução pelo aluno

²⁰ O experimento deve ser técnica e economicamente e principalmente apresentar consequências "aceitáveis"

bastante restritos (seria difícil elaborar um protótipo de estoque, por exemplo!). Por outro lado, o desenvolvimento de relações matemáticas entre diversas variáveis pode ser uma boa alternativa para “previsão” do comportamento de um sistema a partir de determinadas condições. Estas relações são os chamados *modelos matemáticos*.

- **Solução analítica x Simulação:** As relações matemáticas desenvolvidas para compreensão de um problema podem, de acordo com o sistema, assumir formas variadas, desde uma única equação linear a um conjunto de funções cuja solução é “praticamente impossível” de se obter. No primeiro caso, as soluções analíticas são mais adequadas, podendo gerar soluções muito próximas da realidade. No segundo caso, a “previsão” do comportamento dos sistemas complexos podem demandar uma simulação com o auxílio de computadores.

A simulação, muitas vezes considerada como “última opção”, tem tido sua importância e aplicação bastante ampliadas com o desenvolvimento dos computadores que realizam cada vez mais cálculos em tempos menores. Uma das críticas que recebe, no entanto, são ao fato de que muitas vezes um modelo de simulação acaba tornando-se mais um exercício de programação de *software* do que resolução do problema em si!

3.1.2 CLASSIFICAÇÕES DE MODELOS DE SIMULAÇÃO

Dinâmico x Estático:

Dependendo do objetivo da simulação, pode ser interessante verificar o comportamento de um sistema ao longo do tempo, analisando-se um “comportamento médio” ou uma “somatória” – é o caso da *simulação dinâmica*, onde as variáveis são determinadas em função do tempo.

Em outras situações, uma “fotografia” é suficiente, ou seja, deseja-se saber como o sistema se comportaria em uma dada situação – é o caso da *simulação estática*, onde as

variáveis não dependem do tempo ou são de interesse apenas em um momento específico.

Determinístico x Estocástico:

Um modelo determinístico é elaborado de modo que não haja componentes aleatórias ou probabilísticas em sua formação. Desta forma, o resultado deverá ser sempre igual para um mesmo modelo com os mesmos parâmetros. Em resumo, *os resultados são totalmente determinados pelo modelo*. É o caso, por exemplo, da simulação da trajetória de um corpo sujeito a diversas forças: dado que a reação do corpo a cada uma destas forças é conhecida "com precisão", não há motivos para se acrescentar aleatoriedade ao modelo!

Já no caso estocástico, algumas entradas do sistema (ou algumas relação entre variáveis) não têm comportamento pré-determinado. Nestes casos, estimativas inexatas devem ser feitas para o comportamento das mesmas, geralmente com a utilização de cálculos probabilísticos. Como estes métodos normalmente são susceptíveis de aleatoriedade, há "espaço" para diferentes resultados em diferentes simulações com os mesmos parâmetros sem que seja possível, a princípio, determinar qual resultado é mais adequado. Estas diferenças são inerentes à própria aleatoriedade do sistema real! Como exemplo, pode-se citar a demanda de um produto no mercado.

Discreto x Contínuo

Define-se um *sistema discreto* como aquele onde as variáveis mudam de valor instantaneamente em momentos específicos. Ao contrário, os *sistemas contínuos* são aqueles cujas variáveis podem mudar de valor continuamente ao longo de períodos de tempo. O que percebe-se na prática, no entanto, é que poucos sistemas são

exclusivamente discretos ou contínuos, surgindo uma “terceira” categoria de sistemas mistos.

Analogamente, definem-se os modelos *discretos* e *contínuos*. É importante ressaltar, no entanto, que não necessariamente um sistema discreto deve ser modelado discretamente (o mesmo valendo para sistemas contínuos). Isso porque deve haver uma conciliação entre o sistema, o modelo e os objetivos da modelagem!

3.1.3 SIMULAÇÃO DE EVENTOS DISCRETOS

Nesta seção será abordada a exclusivamente a *simulação de eventos discretos* por ser o modelo mais adequado à programação de produção, conforme descrito:

A simulação de eventos discretos caracteriza-se pelos eventos que podem alterar a condição do sistema. Como, por definição, o sistema somente pode ser alterado na ocorrência destes eventos, obtém-se uma oportunidade para “avançar” o tempo nas simulações dinâmicas. Ou seja, o *relógio de simulação* (do inglês, *simulation clock*) pode andar a intervalos irregulares, guiados pelo próximo evento de modo a reduzir o processamento de dados! Em contraposição, tem-se a opção “menos econômica” de avanço no tempo em intervalos fixos.

Uma observação que não pode ser esquecida é quanto a unidade de tempo. Normalmente não há necessidade de se explicitar a unidade utilizada. Isto porque se todas as medidas forem coerentes (mesma unidade), o resultado também será!

Seqüência de operações

A seqüência das operações de uma simulação foi bem ilustrada por LAW; KELTON (1991), conforme reproduzido na figura 3.1 e complementado pelas definições abaixo²¹:

- **Estado do sistema:** Conjunto de variáveis de estado necessárias para descrever o sistema em um dado momento
- **Relógio de simulação:** Variável que indica o valor do tempo simulado
- **Lista de eventos:** Listagem contendo o próximo momento que cada evento deverá ocorrer
- **Contadores estatísticos:** Variáveis utilizadas para armazenagem de informações estatísticas sobre o desempenho do sistema
- **Rotina de inicialização:** Sub-programa que inicializa o modelo no momento zero
- **Rotina de temporização:** Sub-programa que determina o próximo evento da lista de eventos e avança o relógio de simulação para o momento em que o evento ocorrerá
- **Rotina de evento:** Sub-programa que atualiza o sistema quando um evento ocorre
- **Biblioteca de rotinas:** Conjunto de sub-programas usado para gerar observações aleatórias de distribuições probabilísticas que foram determinadas no modelo
- **Gerador de relatórios:** Sub-programa que processa os contadores estatísticos e gera relatórios ao final da simulação
- **Programa principal:** Programa que “coordena” a simulação

Em poucas palavras, LAW; KELTON (1991) resume o processo de simulação de eventos discretos:

²¹ Retirado de LAW; KELTON (1991). tradução pelo aluno

“Um relógio de simulação e uma lista de eventos interagem para determinar qual o próximo evento a ser processado. o relógio é avançado para o momento do evento e o computador executa a lógica do evento, que pode incluir atualização de variáveis de estado, manipulação de listas de filas e eventos, geração de números e variáveis aleatórias e coleta de dados estatísticos. Esta lógica é executada na ordem do tempo simulado do evento. ou seja, a simulação é sequencial.”²²

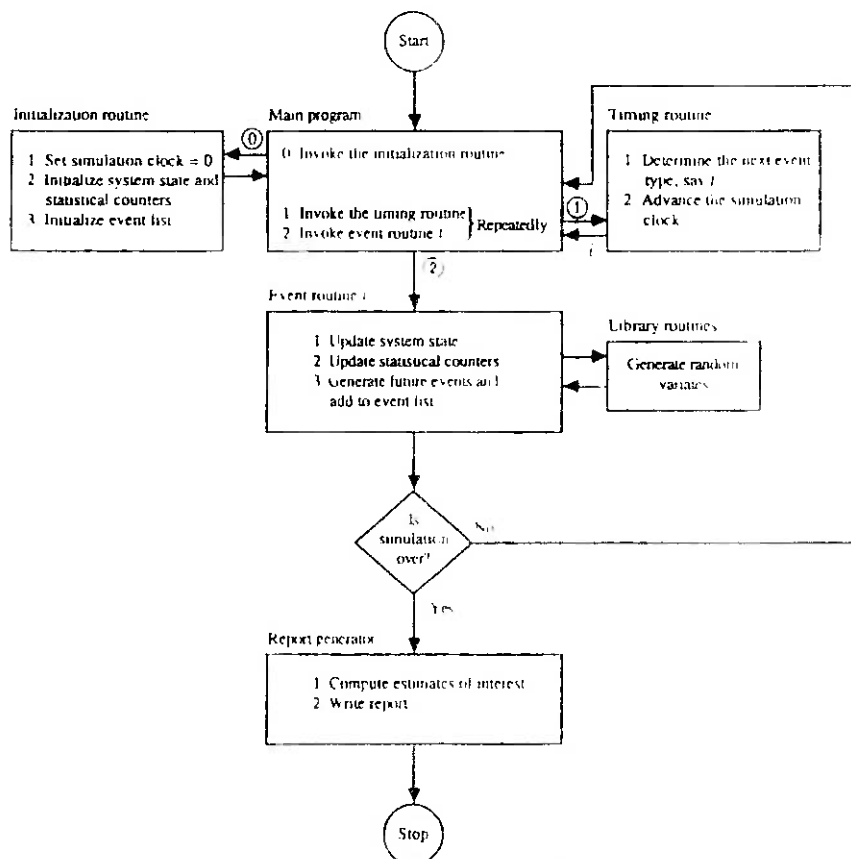


Figura 3.2 – Fluxograma de controle do avanço de tempo

Fonte: LAW; KELTON (1991)

Etapas no desenvolvimento

Para melhor gerenciar o desenvolvimento de um modelo de simulação, LAW; KELTON (1991) estrutura o processo em etapas de planejamento e execução:

²² LAW; KELTON (1991) – tradução pelo aluno

- **Formular problema e planejar estudo:** De início, deve-se ter claramente definido qual o problema a ser estudado e o que se espera como resultado. Ou seja, os objetivos do estudo. Com isso, pode-se planejar e elaborar um modelo com foco nas necessidades reais
- **Coletar dados e definir o modelo:** Um “pré-modelo” ou alguns parâmetros do modelo somente podem ser definidos a partir de dados coletados no sistema real. É o caso, por exemplo, da distribuição aleatória das chegadas dos clientes em um determinado posto de serviço. Pela grande dificuldade que algumas modelagens podem apresentar nesta etapa, LAW; KELTON (1991) enfatiza que “A construção de um modelo matemático e lógico de um sistema real para um dado objetivo ainda é uma arte tanto quanto uma ciência”²³. Por conta desta dificuldade, uma das poucas recomendações “unânicas” encontradas na literatura é de sempre partir de modelos simples e progredir para modelos mais complexos e completos.
- **Validar o modelo:** A validação do modelo deve, preferencialmente, ser feita a medida em que é desenvolvido, apesar de muitas vezes, na prática, não ser. Em ambas as situações, no entanto, uma validação “rigorosa” deve ser feita após a conclusão de cada *versão* (final ou preliminar). Ou seja, deve-se verificar todas as premissas e cálculos propostos e certificar-se de que não estão distantes da realidade, procurando sempre pessoas ligadas aos processos envolvidos.
- **Elaborar programa de computador e verificar:** Considerando-se que as simulações serão feitas por computador, o primeiro passo é escolher uma linguagem para o desenvolvimento (que pode ser genérica ou específica para simulações). O desenvolvimento do código deve ser sempre acompanhado de testes e verificações para garantir que o que está escrito é o que pretendia-se escrever.
- **Teste piloto:** Testes pilotos são simulações realizadas com *dados piloto*, que podem ser em menor escala que o real, com o propósito de validação do sistema
- **Verificar validação:** Além da validação do modelo, deve-se fazer a validação do simulador. Ou seja, deve-se comparar as entradas do teste piloto com as saídas do

²³ Tradução pelo aluno

simulador. Caso sejam coerentes, pode-se prosseguir com a validação do sistema; caso contrário, deve-se repensar verificar o desenvolvimento (programa do simulador) ou mesmo repensar o modelo

- **Projetar experimentos:** Em alguns casos, os simuladores podem permitir várias configurações do sistema a ser simulado. Nestes casos, é parte do desenvolvimento do modelo projetar (selecionar) as configurações e possibilidades de interesse
- **Rodar simulador:** Com o modelo em mãos, chega o momento de utilizá-lo com sua real finalidade.
- **Analisar resultados:** O trabalho praticamente se encerra com os resultados obtidos, que podem ser analisados de diversas maneiras (comumente utiliza-se ferramentas estatísticas) conforme o objetivo.
- **Documentar, apresentar e implementar resultados:** Encerrando o processo, deve-se garantir que o modelo esteja devidamente documentado, pois provavelmente outras pessoas necessitarão estudá-lo, compreendê-lo e atualizá-lo. Finalmente, considerando-se que os resultados serão válidos e viáveis, chega a fase de implementação!

3.2 PROJETO DE BANCO DE DADOS

Neste trabalho, um banco de dados deverá ser utilizado para armazenar informações e parâmetros do modelo. No entanto, por não se tratar do foco do trabalho, apenas os principais conceitos serão abordados, pois foram necessários para se garantir o melhor aproveitamento do sistema.

Normalização de dados

Segundo SPENIK; SLEDGE (1996), “*Normalização* é conjunto de regras padrão que testa a adequação de um projeto de banco de dados”. O conjunto de três regras gera as três formas normais (após a terceira, o banco de dados é dito *normalizado*)

- **1ª Forma Normal:** *nenhum grupo repetido* – Se uma informação pode aparecer repetidamente, por exemplo para três meses, deve-se criar uma coluna indicando o mês e outra para a informação. Não se deve criar na tabela três colunas para cada mês pois isto retira toda a flexibilidade do sistema
- **2ª Forma Normal:** *nenhum atributo não chave depende de parte das chaves primárias* – Isto quer dizer nas tabelas onde mais do que um campo é chave primária (por exemplo, em um arquivo de notas fiscais pode-se utilizar a série e o número da nota), nenhum atributo não chave deve ser colocado se for dependente apenas de parte das chaves primárias (este atributo deve estar na tabela específica desta parte!)
- **3ª Forma Normal:** *nenhum atributo depende de outros atributos não chave* – Não se deve repetir informações desnecessariamente, sendo isto feito a partir da adequação das tabelas. Por exemplo, se deseja-se registrar os treinamentos dos funcionários em uma empresa, deve-se ter uma tabela com os dados do funcionário e outra com as informações de treinamento onde aparece apenas o código do funcionário (nesta tabela de treinamento não há necessidade de repetir o nome ou cargo, pois pelo código já é possível saber estes dados!)

Integridade Referencial

Integridade referencial é o relacionamento que se cria entre tabelas para garantir a consistência da normalização. Mantendo o exemplo anterior de treinamento de funcionários, deve-se garantir que todos os funcionários que estão na tabela de treinamento devem aparecer antes na tabela de funcionários (não se pode treinar um funcionário inexistente!!!)

A integridade referencial deve ser feita através do relacionamento entre campos chave refletir alterações de dados, exclusão e acréscimo de registros.

Índices

Índices são campos que não necessariamente são chave, mas que contém informações “importantes” e freqüentemente procuradas.

Quando o banco de dados recebe um comando solicitando alguma informação, uma das situações abaixo deve ocorrer:

- **Informação solicitada não permite repetições:** deve-se procurar até encontrar o primeiro registro. Como não há repetições, não há necessidade de olhar para o resto da tabela, pois certamente não haverá outro registro que satisfaça as condições de procura
- **Informação permite repetição:** Neste caso, encontrar um primeiro registro não é condição suficiente para parada de busca. Se não houver ordenação no campo procurado, será necessário olhar todos os registros para certificar-se de que a resposta é completa. Por outro lado, se o campo estiver ordenado, o intervalo onde deve-se procurar é bem menor.

O que um *índice* faz é criar uma “ordenação virtual”, ou um vetor de índices que indica onde podem estar os registros procurados, mesmo que na tabela os dados estejam espalhados. A utilização de índices normalmente gera maior eficiência nas buscas do banco de dados.

4. DEFINIÇÃO DO SISTEMA

A partir dos conceitos apresentados nos capítulos anteriores, deve-se agora definir o que será necessário para o desenvolvimento do sistema, bem como suas limitações, pontos fortes e abrangência. Os principais pontos de definição estão mencionados abaixo e detalhadamente descritos em seguida.

O primeiro passo para o desenvolvimento do simulador é o estabelecimento da maneira como um modelo será definido ou caracterizado. Ou seja, o que é preciso para se caracterizar um sistema produtivo (como definir roteiros, caracterizar operações, ...) ?

Outro ponto que deve ser definido é a abrangência (ou nível de *flexibilidade*) que o simulador terá. Apesar do objetivo ser desenvolver um sistema flexível, estamos trabalhando com um modelo que é, por definição, uma simplificação da realidade. A medida que acrescentamos flexibilidade ao sistema estaremos também acrescentando complexidade ao desenvolvimento do mesmo. Assim, faz-se necessário definir um “ponto de parada”, ou seja, até que ponto o esforço para se agregar funções ao sistema é válido (equivalentemente, onde está o ponto de equilíbrio da relação custo/benefício?). Nesta etapa devem sair os pontos fortes, as limitações e as premissas do sistema.

Em seguida, deve-se decidir as técnicas que serão utilizadas em cada etapa. Por exemplo, qual será o critério para agrupamento de necessidades e formação das ordens de produção?

Um outro ponto que não foi muito comentado anteriormente é como transformar a sequenciação em um programa de produção. Apesar de ser uma questão *conceitualmente* simples, pode haver algumas complicações para sua operacionalização.

Os relatórios gerados pelo sistema também devem ser considerados, pois é uma fonte de informação para os dados que poderão ser necessários a qualquer momento. Ou seja, alguns dados “provisórios” (gerados durante a simulação) podem precisar ser armazenados para geração de relatórios.

Finalizando a definição do sistema é necessário definir o ambiente no qual será desenvolvido. Temos atualmente uma infinidade de opções de banco de dados e linguagens de programação, apresentando diferentes custos, porte e características técnicas. Esta decisão pode ser de vital importância para a viabilidade do produto final, devendo portanto ser feita com muita atenção.

4.1 O QUE DEFINE UM MODELO

Esta etapa tem como função definir o mapeamento da operação da empresa. Em outras palavras, deve-se fazer um levantamento de todas as informações que o sistema precisará para rodar adequadamente:

- **Item:** Cadastro completo de todos os itens, incluindo matérias-primas, semi-acabados e produto acabado
- **Lista de materiais:** Necessária para formação das necessidades dos itens de nível mais baixo
- **Roteiro:** Relação dos recursos de trabalho pelos quais uma ordem deve passar
- **Operação:** Parâmetros que caracterizam uma operação e permitem os cálculos necessários
- **Recurso de Trabalho:** Cadastro dos recursos de trabalho, que podem ser máquinas (caso mais comum), equipamentos ou mesmo mão-de-obra. Ou seja, dado que uma operação pode utilizar mais do que um recurso simultaneamente, o cadastro deve contemplar sempre o mais escasso ou limitante de capacidade

- **Disponibilidade:** Carga permitida de trabalho para cada recurso. Necessário para a transformação da seqüenciação realizada em um programa de produção

4.2 ABRANGÊNCIA DO MODELO

Qualquer operação real apresenta restrições e peculiaridades que podem, se consideradas “integralmente”, dificultar ou mesmo impedir a resolução do problema. No entanto, é comum que um pequeno “relaxamento” nestas restrições gere uma grande facilidade para a resolução do mesmo problema. É exatamente esta a função de um *modelo*: ser uma simplificação da realidade, sem perda das características principais do sistema real!

Seguindo este raciocínio, o simulador proposto deve partir de algumas premissas, as quais determinarão a abrangência, pontos fortes e limitações com as quais o programa poderá lidar. Alternativamente, pode-se determinar inicialmente a abrangência desejada e em seguida definir as premissas.

4.2.1 PREMISSAS²⁴

Como já foi visto na revisão bibliográfica, o problema de programação é extremamente amplo e, de maneira mais generalista, pode tomar dimensões muito além do domínio de um departamento de Programação de Produção. Assim, de imediato parte-se das premissas:

1. *A capacidade produtiva é dada e não se questiona a possibilidade de ampliação, redução ou realocação*

²⁴ Algumas das premissas apresentadas são retiradas ou adaptadas de CONWAY (1967) apud HAX (1984)

2. *As necessidades (item, quantidade e data de entrega) de produção serão entradas do usuário, de modo que o planejamento mestre e as etapas de nível superior não serão abrangidas*

Seguindo lógica de foco em um “Departamento de Programação”, deseja-se excluir as responsabilidades de um “Departamento de Suprimentos” do sistema:

3. *Matérias-primas, materiais de embalagem ou quaisquer outros itens de origem externa estarão disponíveis em estoque quando necessário*

Foi visto também que em diferentes processos produtivos (estoque puro, produção contínua, produção intermitente e projetos) a programação assume diferentes papéis. O balanceamento de linhas (produção contínua), por se tratar mais de um problema de *projeto* do que de *rotina*, não será foco deste trabalho, assim como a *produção por projetos*. Processos de estoque puro não são abordados pela programação, restando assim os **processos intermitentes**. Disso retira-se a terceira premissa do sistema:

4. *O simulador terá como foco os processos intermitentes, que têm como problema de programação a **seqüenciação** das ordens de produção nas máquinas*

Os recursos utilizados na produção (ou os processos produtivos) podem, na prática, apresentar particularidades que ajudem a melhorar a produtividade e aumentar a eficiência logística do sistema. Como tratam-se de situações muito específicas, o sistema limitará estas possibilidades:

5. *A cada operação deve ser alocado um, e apenas um, “tipo” (especificação) de recurso de trabalho. Este recurso pode ser uma máquina, equipamento ou mesmo mão-de-obra, desde que seja o fator limitante*
6. *Pode haver repetição de recursos do mesmo tipo, desde que tenham mesma capacidade*

7. *Cada recurso pode realizar no máximo uma operação de cada vez*
8. *Uma operação não pode ser interrompida antes de sua conclusão para realização de outra operação*
9. *O resultado de uma operação somente pode ser utilizado depois da conclusão total da mesma*
10. *Depois de concluída uma operação, o produto resultante tem espaço disponível para armazenagem de modo que o recurso será imediatamente liberado*

Apesar de uma fábrica trabalhar, em seus diversos departamentos, com diferentes jornadas de trabalho e turnos, esta consideração geraria grandes dificuldades de sincronização entre os centros de trabalho e cálculos com *lead-time*. Desta forma:

11. *A disponibilidade dos recursos de trabalho é a mesma para qualquer recurso e pode ser definida independentemente para cada dia da semana (sendo sempre uma semana igual à outra), com um horário de início e um de encerramento (ou 24 horas no dia)*
12. *Os recursos estão “inteiramente” disponíveis para programação, não havendo previsão de paradas para manutenção, quebras, feriados, etc.*
13. *As operações podem ser interrompidas ao término do dia para conclusão no dia seguinte ou, alternativamente, há utilização de banco de horas ilimitado (sem limite de horas que possam ser acumuladas)²⁵*

4.2.2 PONTOS FORTES E LIMITAÇÕES

Definidas as premissas que governarão o desenvolvimento do sistema, pode-se determinar os pontos fortes e as limitações do programa. Como *pontos fortes* entende-se a *robustez* e *flexibilidade* que o simulador deve apresentar. Já as limitações são os casos

em que o simulador não seria adequado ou onde algum cuidado especial seria necessário para utilizá-lo.

Algumas limitações que são consequência direta das premissas iniciais não serão novamente listadas:

Pontos Fortes

1. *Não há restrições quanto à seqüência das operações, de modo que o sistema poderá ser utilizado em um centro de configuração genérica (não necessariamente em um centro com máquinas em fluxo)*
2. *Há possibilidade de recursos de trabalho iguais para realizarem uma mesma operação, desde que tenham mesma capacidade*
3. *Há possibilidade de diferenciação²⁶ nos tempos de set-up de acordo com o item que foi previamente produzido no recurso de trabalho*
4. *As jornadas de trabalho podem variar por dia da semana*
5. *Tempo de transporte entre recursos pode ser considerado. Para tanto, deve-se registrar uma operação de transporte*

Limitações

1. *Operações que não possam ser interrompidas podem ser agendadas inadequadamente caso não haja utilização de banco de horas*
2. *Uma operação não pode estar susceptível a limitações por dois ou mais recursos de trabalho*
3. *Não há considerações de perdas de processo. Quando significativas, as perdas de processo devem ser incluídas na lista de materiais corrigindo-se as quantidades*

²⁶ O motivo desta premissa será elucidado no tópico 4.4 – Escala de Tempo

4. *Não há previsão de alocação de tempo para manutenção ou outras paradas programadas, assim como as semanas são consideradas iguais (inexistência de feriados, tempo para treinamento de funcionários, etc.)*
5. *A jornada diária é ininterrupta*

4.3 ESCOLHAS UTILIZADAS

Este tópico tem como finalidade a tomada de algumas decisões quanto à operacionalização do sistema. Primeiro, serão elaboradas algumas técnicas e ferramentas para contornar algumas das críticas ao MRP apresentadas no capítulo 2. Em seguida será escolhido algum dos métodos apresentados para dimensionamento de lotes. Finalizando, será determinada a técnica de seqüenciação dentre as opções apresentadas no item 2.7.2.

4.3.1 CONTORNANDO CRÍTICAS DO MRP

Uma das principais críticas ao MRP, conforme citado no item 2.5.8, é a consideração de *lead time* constante para o item. Visto que o *lead time* é uma soma de tempos de espera processamento e do tempo de processamento propriamente dito, o sistema desenvolvido utilizará esta diferenciação. Assim, o usuário deverá cadastrar não o *lead time* do item, mas o tempo esperado em fila e o tempo de operação. Esta adaptação reduz as distorções de tempo para centros onde o tempo em fila sofre grandes oscilações.

O tempo de operação, no entanto, também pode variar consideravelmente se o tamanho dos lotes produzidos também variar. Assim, o tempo de operação deverá ser dividido

²⁵ A diferenciação será feita em três níveis: quando o item anterior for igual ao que será produzido, for da mesma família ou quando for de família diferente

em tempo de *set-up*, tempo de operação unitário²⁷ e tempo de *clean-up*. Pode-se ainda pensar o tempo de *set-up* pode variar de acordo com o item produzido anteriormente na mesma máquina. Para comportar este efeito, o sistema permitirá três tempos de *set-up* diferentes, caso o item anterior seja igual, da mesma família ou diverso.

Pelas considerações acima, surge uma dificuldade na modelagem relativa à ordem de grandeza de tempo. Ou seja, enquanto algumas empresas teriam os tempos medidos em segundos ou minutos, outras teriam o tempo medido em dias ou mesmo semanas. Para oferecer maior flexibilidade ao sistema, a unidade de tempo será também um parâmetro variável.

Deve-se lembrar também que o *lead time*, dependendo da escala de tempo utilizada, pode sofrer grandes distorções por conta da descontinuidade das operações, ou seja, parada noturna e fins-de-semana. Por exemplo, falar que determinado lote necessita quatro horas para ser processado, pode, na prática, representar quatro mais doze horas, pois a empresa pára a produção durante a noite. Para evitar este problema, todos os cálculos serão realizados com base em horas úteis (ou “trabalháveis”) e somente o resultado final será “desconvertido” para data e hora reais, conforme será detalhado no tópico 4.4

4.3.2 TÉCNICA DE DIMENSIONAMENTO DE LOTES

Primeiramente, é interessante entender por que é necessário dimensionar os lotes, uma vez que conceitualmente o MRP propõem uma programação *lot-for-lot* à medida que diz “precisamente” quando um item será requisitado e quando deve ser produzido

²⁷ O sistema deve considerar também operações cujo tempo independe do tamanho do lote, como uma

(teoricamente dispensando estoques). No entanto, esta técnica pode (e provavelmente vai) gerar um alto número de *set-ups*, acarretando alto custo e baixa produtividade. Por este motivo, o aluno considera adequada a possibilidade de “juntar” lotes seguindo critérios de alguma das técnicas apresentadas.

Dentre os nove modelos apresentados, o única que apresenta resultados ótimos é o algoritmo de Wagner-Whitin. No entanto, devido à sua complexidade ficaria fora do escopo deste trabalho.

Os modelos de *ordem de quantidade fixa e necessidades por período fixo* são muito “pobres” e somente produzirão soluções boas e econômicas em situações bastante específicas. O modelo de *ordem de quantidade econômica* pode frequentemente ser inadequado por gerar estoques remanescentes sistematicamente.

O *menor custo unitário, menor custo total e ordem de quantidade por período* “tentam” minimizar custos, porém não se preocupam em verificar a qualidade da solução.

Já o método de ***Part-period balancing (PPB)*** tem a grande vantagem dos testes *look-ahead e look-back* que, após definir um lote “preliminar” verifica se é preferível abandonar o último lote agregado ou agregar um próximo lote. Assim, será a técnica utilizada.

Com parente
pobre

4.3.3 TÉCNICA DE SEQUÊNCIAÇÃO

A técnica de sequenciação utilizada será por simulação de regras de priorização. Este método é o mais largamente utilizado por gerar bons resultados (não necessariamente ótimos) e pela relativa facilidade de aplicação.

Métodos heurísticos podem também ser simples, mas são restritos a casos muito específicos.

4.4 ESCALA DE TEMPO

Pelas “adaptações” propostas para contornar as críticas ao MRP, surgiu a necessidade de elaboração de uma escala de tempo que considera apenas o “tempo útil”. Se o objetivo do sistema fosse fazer apenas a sequenciação, qualquer escala arbitrária poderia ser utilizada. No entanto, a geração de um *programa de produção* exige alguns critérios um pouco mais elaborados.

As datas de entrega determinadas pelo usuário serão convertidas em uma escala cuja unidade foi escolhida arbitrariamente como milésimos de hora²⁸. A origem desta escala será o início da semana (zero hora de domingo) do dia definido como início de simulação (entrada do usuário). Esta origem foi escolhida para que se mantenha um padrão para a conversão, uma vez que as semanas são consideradas iguais, porém os dias diferentes. Qualquer outro momento poderia ter sido escolhido, o importante é que se saiba onde é o início para a “desconversão” da escala criada para datas e horários reais.

²⁸ Utiliza-se milésimos de hora ao invés de hora pois isto permite trabalhar apenas com números inteiros e sem perda de precisão. Números inteiros são preferíveis para a armazenagem e processamento de dados

A figura 4.1 ilustra como seria o comportamento do *tempo simulado* (escala criada) em função do *tempo real* para os dados de disponibilidade apresentados na tabela 4.1.

Após a conversão das datas inseridas pelo usuário, todas as demais operações são realizadas nesta escala, considerando-a contínua e ininterrupta. É por este motivo que foi levantada a limitação (1) acima mencionada (as operações podem ser agendadas com interrupção ou forçando utilização de banco de horas), visto que durante o processo não é feita nenhuma verificação sobre qual o valor real das “datas” aparentes.

Dia da semana	Início das operações	Término das operações
Domingo	00:00	00:00
Segunda-feira	08:00	18:00
Terça-feira	08:00	18:00
Quarta-feira	08:00	18:00
Quinta-feira	08:00	18:00
Sexta-feira	08:00	18:00
Sábado	08:00	12:00

Tabela 4.1 – Dados ilustrativos de disponibilidade de trabalho

No gráfico abaixo, perceber que a função vale zero no início da semana (zero hora de domingo) e apresenta crescimento zero nos períodos em que não se trabalha e crescimento linear com inclinação de 45° (1 h real equivale a 1 h simulado) nos períodos onde há trabalho. Ao final da semana ($t = 168$ h), tem-se uma carga de trabalho acumulada de 54 h.

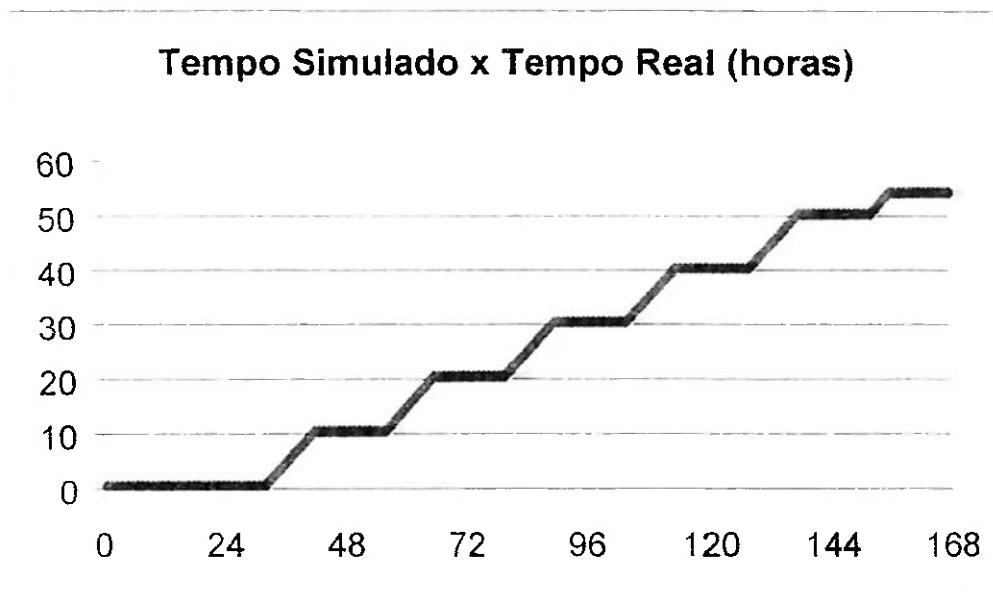


Figura 4.1 – Gráfico de tempo simulado x tempo real (horas)

4.5 RELATÓRIOS

Os relatórios gerados pelo sistema devem abranger as seguintes informações:

- **Programa de Produção:** Agregado de diferentes modos, por exemplo: por item, por recurso de trabalho, pela data de início, etc.
- **Ocupação de recursos de trabalho:** Indicador da parcela de tempo em que o recurso de trabalho esteve (teoricamente) ocupado e ocioso, bem como outros indicadores relacionados (como número médio de ordens na fila, etc.)
- **Cumprimento de datas de entrega:** Um dos indicadores mais importantes em um programa de produção, o relatório deve conter relação de todos os pedidos do usuário e uma comparação entre data de entrega e data programada para entrega. Como resumo, deve-se ter número de pedidos atrasados, atraso médio (dias) e atraso máximo (dias)

4.6 AMBIENTE DE DESENVOLVIMENTO

Nos últimos anos, uma grande diversidade de *softwares* e aplicativos de banco de dados foram desenvolvidos e estão comercialmente disponíveis. Estas soluções têm grande

variedade tanto em capacidade e ferramentas como em preço. A seleção do ambiente para desenvolvimento do sistema deve contemplar duas partes independentes: banco de dados e linguagem de programação.

O MS-Access “sozinho” abrange os dois aspectos e apresenta uma série de pontos fortes que seguem:

- **Difusão no mercado:** É um produto altamente difundido no mercado, sendo reconhecido pela “simplicidade” de uso. Esta facilidade é de grande importância quando se considera que universo de pessoas que podem utilizar o sistema.
- **Preço:** Quando comparado a outras linguagens de programação ou banco de dados, o preço é muito atraente e coerente com o orçamento de empresas de pequeno porte
- **“Portatibilidade”:** O MS-Access gera arquivos relativamente pequenos²⁹ que podem ser gravados em disquete e facilmente transferidos para outro computador sem necessidade de instalação. O mesmo não ocorreria, por exemplo, se o desenvolvimento fosse feito em Visual Basic com suporte de um banco de dados como SQL-Server (um dos mais difundidos na categoria).
- **“Economia de trabalho”:** O MS-Access, por ser composto tanto do banco de dados como pela linguagem de programação (baseada em Visual Basic), contém ferramentas que facilmente faz a conexão entre o código de programação, formulários, dados, relatórios e etc. Por se tratar de um trabalho acadêmico, esta facilidade possibilita maior foco no tema principal (em detrimento de um *desing* mais elaborado)

Algumas limitações do MS-Access, no entanto, devem ser levantadas:

²⁹ Os arquivos gerados pelo MS-Access podem “crescer” rapidamente, o alterações não são descartadas. No entanto, é possível reduzir o arquivo através de uma ferramenta de compactação do banco de dados (menu *Ferramentas – Utilidades – Compactar banco de dados*)

- **Porte do banco de dados:** Em aplicações de pequeno porte, o MS-Access é bastante eficiente e prático. No entanto, aplicações de grande porte³⁰ podem ficar bastante prejudicadas pelas limitações de capacidade
- **Controles e Design:** por não se tratar de um programa específico de programação (a função principal é de banco de dados), pode-se encontrar maior dificuldade na aplicação de alguns controles e no *design* de formulários.

³⁰ O MS-Access começa a apresentar "problemas" em tabelas que tenham a partir de 10.000 registros

5. DESENVOLVENDO O SISTEMA

Neste capítulo serão apresentadas as etapas e passos no desenvolvimento do sistema. Primeiramente, será abordado o processo de elaboração das tabelas, uma vez que estas definem o sistema, ou seja, os dados necessários para um modelo estarão sempre armazenados em tabelas.

Em seguida, há a explicação da lógica da programação. Neste tópico é descrito como chegar das encomendas do usuário ao programa de produção.

5.1 ALGUNS PADRÕES NO DESENVOLVIMENTO

Não é fácil entender um código de programação quando este foi elaborado por outra pessoa. Assim, para melhorar a compreensão das operações realizadas e para melhor manutenção do sistema, alguns “padrões” foram adotados (que também auxiliam o programador do sistema a *debugá-lo*).

- **Variáveis:** Os nomes sempre são escolhidos de modo a dar idéia da função da variável. Além disso, uma letra sempre é adicionada no início indicando o tipo da variável (por exemplo, “i” para variáveis inteiras, “s” para *strings*, ...), ajudando a evitar erros de incompatibilidade de tipos
- **Comentários:** Todo o código foi exaustivamente comentado de modo que, ao final, os módulos são “auto-documentados”. Ou seja, o próprio código do programa contém informação suficiente para compreensão e documentação de cada comando
- **Funções:** Deve-se evitar funções muito extensas pois prejudica a localização de erros. Elaborar funções pequenas e com tarefas específicas gera grande flexibilidade e facilidade de manutenção

- **Conexão ao banco de dados:** Para evitar que uma conexão fosse feita a cada função (o que geraria enorme ineficiência), a conexão é feita uma vez e passada como parâmetro as funções (variáveis globais também foram evitadas)
- **Nomes de campos de tabelas:** Em todas as tabelas, os nomes dos campos têm um “prefixo” de três ou quatro letras que permite, em uma consulta, distinguir facilmente a origem do dado
- **Uso de constantes:** Nomes específicos são evitados no meio do código. Sempre que necessário fazer referência a um nome específico (como uma consulta, por exemplo) uma constante com referido valor é criada

5.2 TABELAS

Há dois “tipos” de tabela que precisam ser criadas: *tabelas de registro* e *tabelas de transação*. As tabelas de registro são aquelas utilizadas para cadastros e cujo conteúdo é pouco variável ao longo do tempo (por exemplo, cadastro de itens). As tabelas de transação são aquelas cujo conteúdo é volátil, seja com o acréscimo frequente de novos registros, seja com a alteração dos mesmos (por exemplo, um arquivo de notas fiscais emitidas).

5.2.1 TABELAS DE REGISTRO

As tabelas de registro são aquelas necessárias para que o sistema “entenda” os processos da organização. Portanto, devem conter as informações que definem um sistema, conforme capítulo 4.1. As tabelas criadas foram (os dados completos estão no anexo D e a descrição das tabelas, bem como consultas, no anexo E):

- **Item:** Cadastro de itens
- **Operacao:** Cadastro de operações, incluindo tempos de processo, recursos necessário, etc. Como toda operação gera um e apenas um item e todo item é gerado

por uma e apenas uma operação, tem-se uma relação um para um. Desta forma, um dos parâmetros de uma operação deve ser o produto de saída.

- **EntradaOperacao:** Cadastro dos itens de entrada de cada operação. Visto que a relação item-operacao é um para um, a partir dos itens de entrada obtém-se a listas de materiais.
- **RecursoDeTrabalho:** Cadastro de todos os recursos de trabalho disponíveis na organização. Um dos parâmetros que devem ser inseridos nesta tabela é a regra de seqüenciação
- **Disponibilidade:** Cadastro da jornada de trabalho diária
- **RegraSeq:** Cadastro de regras de seqüenciação. A partir de campos da tabela onde ficam os dados de fila, pode-se “criar” regras que ordenam (ascendente ou descendente) com base em duas propriedades da fila.
- **CamposFila:** Relação de campos da tabela Fila (será apresentada com as demais tabelas de transação) a partir dos quais pode-se elaborar regras de seqüenciação. Ou seja, os campos onde são armazenadas as propriedades dos elementos das filas (por exemplo, data de entrega, tempo de processo, etc.)
- **Categoria:** Uma categoria define a origem de um produto. Ou seja, a categoria *Matéria-Prima*, por exemplo, contém itens que são comprados externamente (e portanto não precisam ser processados). Já a categoria *Semi-Acabados* define itens cuja produção é interna e portanto precisam ser processados.
- **Centro de Trabalho:** Um conjunto de recursos de trabalho. Não é obrigatório alocar um centro de trabalho a cada recurso de trabalho. Esta propriedade somente é utilizada para organização de relatórios
- **Família:** Cadastro de famílias, que representam um conjunto de itens. A maior importância às famílias é para o cálculo de *set-up*
- **Grandeza:** Cadastro de grandezas de medida. Também somente é utilizado para geração de relatórios

- **Unidade:** Cadastro de unidades de medida. A principal função é indicar a unidade na medida dos tempos de operação. Unidades de outras grandezas são utilizadas apenas para melhor organização dos dados, não sendo utilizada na programação
- **LeadTime MP:** Relação de prazos de entrega para os itens de origem externa. Como estes itens não precisam ser processados, a utilização desta tabela restringe-se a elaboração de relatórios de necessidades de matéria-prima

Percebe-se que não há uma tabela específica para *roteiros*. Isto porque não há necessidade, uma vez que as tabelas *Operacao* e *EntradaOperacao* conjuntamente contém as informações de roteiro, pois cada etapa de um roteiro pode ser definida com uma operação que gera um item a partir de determinadas entradas.

5.2.2 TABELAS DE TRANSAÇÃO

As tabelas de transação são aquelas onde informações geradas são armazenadas e portanto têm conteúdo bastante dinâmico. Grande parte destas informações geradas são apenas provisórias e podem ser descartadas ao final da simulação. Para o cumprimento da lógica da programação (que será apresentada no próximo tópico) foi necessária a criação das seguintes tabelas (a relação completa de tabelas e respectivos campos está no anexo E):

- **Encomenda:** São as necessidades inseridas pelo usuário, que devem conter um item, quantidade e data e hora de entrega
- **EncomendaInterna:** São necessidades de produção geradas pelo sistema, podendo ser a própria encomenda do usuário ou provenientes de demanda dependente para produção das ordens solicitadas pelo usuário

- **Fila³¹:** Relação de todas as ordens que devem ser processadas, relacionando-as a algum recurso de trabalho e atribuindo algumas propriedades que podem ser utilizadas como critério de priorização
- **Inventario Dinamico:** Tabela que apresenta o *status* de inventário ao longo do tempo simulado. É utilizada durante a simulação para determinar se uma ordem pode ou não ser processada (disponibilidade de materiais)
- **Inventario Inicial:** Apresenta o *status* inicial do inventário, de forma que itens disponíveis inicialmente podem ter sua necessidade produção reduzida
- **LowLevelCoding:** Antes de toda simulação, uma tabela que relaciona itens e *level coding* é atualizada, pois uma falha nesta relação pode inviabilizar a programação e mesmo gerar um *loop* infinito
- **OrdemDeProducao:** São os lotes de produção gerados pelo sistema, provenientes das necessidades liquidas (*encomendas internas*) e do agrupamento de lotes
- **Programa:** É o programa de produção gerado, na escala do tempo simulado
- **Programa DataHora:** É um “resumo” do programa de produção com indicações de tempo real (data e hora) para início e conclusão de cada operação. Apesar da mesma informação estar disponível na tabela *Programa*, esta duplicação é necessária pois a “desconversão” da escala de tempo depende da data definida para início da simulação, que pode não estar disponível em momento posterior para emissão de relatórios, por exemplo.
- **RelacionamentoDeOrdens:** Armazena as informações de relacionamento entre ordens, ou seja, para onde será destinada a produção de cada ordem concluída
- **Verificacao:** Armazena mensagens de erro geradas nas verificações feitas no modelo antes da execução da programação

³¹ Esta tabela “desobedece” o conceito de normalização de dados apresentado no capítulo 3 (a partir da ordem de produção pode-se tirar o recurso de trabalho correspondente). No entanto, esta “falha” na armazenagem dos dados é bastante compensada pela maior facilidade no processamento da informação

5.3 SEQUÊNCIA DE OPERAÇÕES

As operações realizadas pelo sistema para geração de um programa de produção foram divididas em três fases:

- **Fase 0 – Inicializações:** São feitas verificações da consistência de dados e algumas inicializações
- **Fase 1 – Geração das Ordens de Produção:** Nesta etapa é rodado o MRP e são gerados os lotes (ou ordens) de produção, os quais são alocados em uma fila para posterior sequenciação
- **Fase 2 – Simulação e Sequenciação:** Com as filas formadas em todos os recursos de trabalho, tem início a simulação propriamente dita que gera a sequenciação e o programa de produção

Seguem as operações realizadas em cada uma destas fases (o código de programação está no anexo A:

Fase 0 – Inicializações

Função	Descrição
<i>InicializarModelo</i>	- Função principal da fase 0 que "gerencia" as demais funções
<i>ApagarTabelaVerificacao</i>	- Apaga erros gerados em verificações anteriores
<i>VerificarEntradaSaidaOperacao</i>	- Verificar se alguma operação tem o mesmo item como entrada e saída
<i>AtualizarLLC</i>	- Atualizar relação LLC
<i>VerificarModelo</i>	- Função que agrupa outras funções de verificação
<i>VerificarEncomendas</i>	- Verificar se há encomendas do usuário
<i>VerificarDisponibilidadeTrabalho</i>	- Verificar coerência do cadastro de disponibilidade de trabalho
<i>VerificarRestricaoLote</i>	- Verificar se é possível formar lote seguindo restrições de máximo, mínimo e múltiplo
<i>VerificarItemSemOperacao</i>	- Verificar se algum item não tem operação correspondente
<i>VerificarOperacaoSemEntrada</i>	- Verificar se alguma operação não tem entrada correspondente
<i>VerificarOrdemEntradaOperacao</i>	- Verificar se alguma operação tem repetição na ordem de entrada
<i>LimparTabelasDeSimulacao</i>	- Apagar todas as tabelas de uso provisório

ImportarInventarioInicial

Importar inventário inicial de um banco de dados externo. Como não é possível criar uma função "geral" para esta tarefa (depende do sistema de informações da organização), esta função não foi desenvolvida

Fase 1 – Geração das ordens de produção

Função	Descrição
<i>PreparaElementosDeSimulacao</i>	- Função principal da fase 1 que "gerencia" as demais funções
<i>TransferirEncomendas</i>	- Transferir as encomendas do usuário para a tabela de necessidades brutas (EncomendaInterna)
<i>NormalizarEncomendas</i>	- Converter encomendas internas para escala de tempo de simulação
<i>GerarOrdemDeProducao</i>	- Agrupar necessidades e gerar ordens de produção para itens nível 0
<i>GerarOrdemDeProducaoIndireta</i>	- Agrupar necessidades e gerar ordens de produção para itens de outro nível, descontando inventário inicial
<i>GerarFila</i>	- Gera registros na tabela de filas

Fase 2 – Simulação e seqüenciação

Função	Descrição
<i>SequenciarOrdensDeProducao</i>	- Função principal da fase 2 que "gerencia" as demais funções
<i>AtualizarPropriedadesDinamicas</i>	- Atualizar as propriedades da fila que variam ao longo do tempo (por exemplo, tempo restante para entrega)
<i>AtualizarTempoSetUp</i>	- Atualizar o tempo de set-up para cada ordem, de acordo com o que tiver sido produzido anteriormente
<i>AtualizarDisponibilidadeMateriais</i>	- A cada conclusão de operação, novos itens entram em estoque e outras operações podem ser iniciadas
<i>AtualizarTemposDinamicos</i>	- Atualizar algumas propriedades da fila relacionadas ao tempo de processo e tempo restante
<i>DefinirRecursoDeTrabalho</i>	- Selecionar recurso(s) de trabalho disponível(is) para alocação de nova ordem
<i>AbrirRecordsetFila</i>	- Selecionar ordenadamente as ordens do(s) recurso(s) de trabalho que estiver livre
<i>RealizarOperacao</i>	- Definir uma operação como realizada, agendando-a na seqüenciação
<i>GerarProgramaDataHora</i>	- Gerar um programa de produção a partir da seqüenciação

6. UTILIZANDO O SISTEMA

Este capítulo tem como finalidade servir como *manual do usuário* para o sistema desenvolvido. Será explicado como inserir um modelo (como “ensinar” os processos ao sistema), como gerar a programação e que informação contém cada relatório.

Neste capítulo, é interessante que o leitor tenha em mente o formulário principal do sistema (figura 6.1):

Figura 6.1 – Formulário principal do sistema

6.1 INSERINDO UM MODELO

Inserir um modelo significa preencher todas as *tabelas de registro* que foram apresentadas no capítulo 5.2. Devido a restrições de integridade referencial, deve-se respeitar uma sequência ao *inputar* os dados. Por exemplo, não se pode criar uma operação sem que antes seja criado o item associado a ela. Os botões que dão acesso aos formulários de cadastro estão identificados como *Gerenciamento do Modelo*.

Cada um destes botões dará acesso a um formulário (a imagem de todos os formulários está no anexo B). Movendo-se o cursor entre os campos deste formulário, no canto esquerdo inferior da janela do MS-Access aparecerá uma descrição da informação necessária, conforme exemplo na figura 6.2

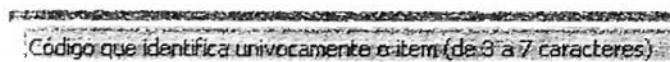


Figura 6.2 – Exemplo de descrição da informação necessária (tabela Item, campo código de item)

Ao *inputar* os dados, deve-se sempre ter em mente que há algumas relações de precedência entre eles (integridade referencial), ou seja, não é possível inserir, por exemplo, uma operação sem antes criar o item que será associado a ela! A tabela 6.1 resume as relações de precedência

Dado	Precedência
Item	Familia, Categoria
Operação	Item, Recurso de Trabalho
Entrada de Operações	Operação
Recurso de Trabalho	Centro de Trabalho
Disponibilidade	<i>Independente</i>
Regra de Sequenciação	CamposFila
CamposFila	Esta tabela não deve ser manipulada pelo usuário, pois é constante para o sistema e não o modelo
Categoria	<i>Independente</i>
Centro de Trabalho	<i>Independente</i>
Familia	<i>Independente</i>
Grandeza	<i>Independente</i>
Unidade	Grandeza
LcadTime MP	Item

Tabela 6.1 – Relações de precedência de dados

6.2 GERANDO PROGRAMA DE PRODUÇÃO

Definido o modelo, para se gerar o programa de produção é, logicamente, necessário entrar as ordens que devem ser entregues. Clicando no botão “*Encomenda*”, o

formulário da figura 6.2 será exibido, solicitando item, quantidade e data de entrega (o código é de auto-numeração, não sendo necessário preenchê-lo)

Figura 6.2 – Formulário de Encomendas

Dadas as encomendas, deve-se definir uma data e horário para início da simulação e em seguida pressionar o botão *Simular*. Durante o processamento dos dados, é exibido um *status* para que o usuário possa acompanhar o andamento da simulação.

A opção “*Importar inventário*” não pôde ser desenvolvida. Sua função seria buscar dados em outro banco de dados (o sistema de informações da empresa onde está registrado o estoque) para reduzir as necessidades de produção. No entanto, como esta função deve ser diferente para cada “cliente”, não é possível generalizá-la (dados podem ser inseridos manualmente, direto na tabela).

Encerrada a simulação, basta selecionar os relatórios desejados e em seguida pressionar o botão “*Visualizar relatório*” ou “*Imprimir relatório*”.

Exceto a data e hora de início da simulação, nenhuma outra informação é perdida quando o sistema é fechado. No entanto, quando nova programação for feita, os dados da programação anterior serão apagados (as tabelas de registro não são alteradas).

6.3 RELATÓRIOS

Apesar de apenas 8 relatórios estarem inicialmente disponíveis, outros podem ser desenvolvidos de acordo com necessidades específicas (os dados gerados durante a simulação não são perdidos até a próxima simulação). A tabela 6.2 exibe um resumo dos 8 relatórios:

Relatório	Descrição
Cumprimento de datas de entrega	Relação de ordens e respectivas datas de entrega e entrega programada
Programação de produção por centro de trabalho	Programa de produção agrupado por centro de trabalho
Programação de produção por recurso de trabalho	Programa de produção agrupado por recurso de trabalho
Programação de produção item	Programa de produção agrupado por item
Programação de produção data	Programa de produção agrupado por data de início
Ocupação de recursos de trabalho	Taxa de ocupação dos recursos de trabalho
Performance de recurso de trabalho	Tempos de fila em cada recurso de trabalho
Necessidades de matéria-prima	Sugestões de compra de matéria-prima

Tabela 6.2 – Descrição de relatórios

7. APLICAÇÃO DO SISTEMA

parte do sistema?

Por motivos já apresentados, o aluno esteve impossibilitado de realizar um trabalho aplicado a um caso prático e real. Assim, para a aplicação do modelo desenvolvido, foi elaborado um exemplo hipotético de uma indústria moveleira (a qual será referida como a *indústria*).

O modelo será descrito a seguir, sempre levando-se em conta que os valores utilizados são apenas ilustrativos! Os resultados serão apresentados e discutidos no capítulo 8. A relação completa dos dados utilizados aparecem no anexo D.

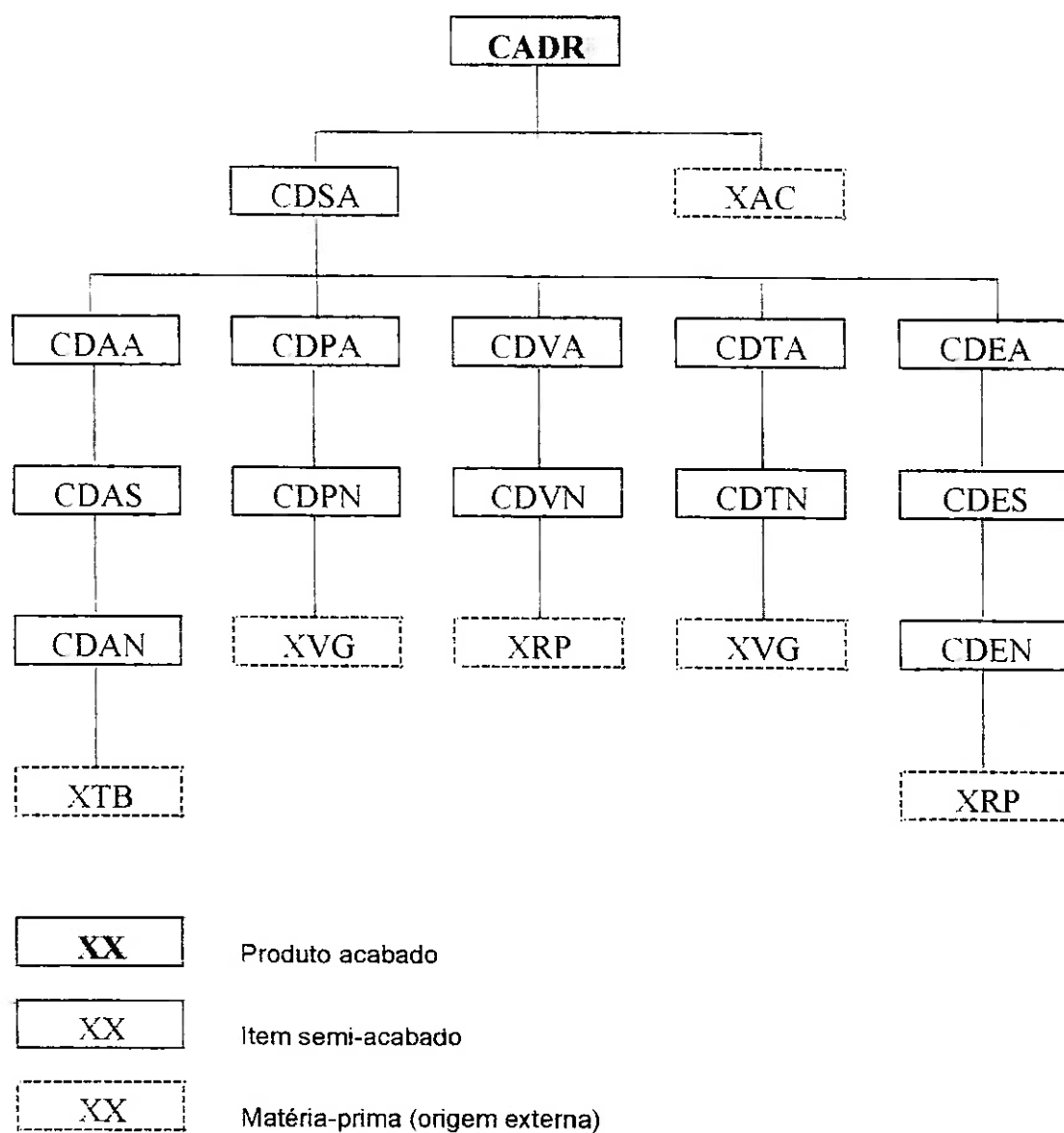
7.1 FAMÍLIAS E ÍTEMS

7.1.1 FAMÍLIAS

A *indústria* tem em sua linha de produtos duas famílias principais (de produto acabado): **mesas** e **cadeiras**. Todos os móveis fabricados pela *indústria* são em madeira e são compostos pela combinação de diversos itens de outras famílias de semi-acabados. Desta maneira, a estrutura dos produtos assemelha-se à forma em "X" (figura 2.2). Para tanto, as famílias de semi-acabado foram elaboradas de maneira que itens de uma mesma família podem ser trocados. Ou seja, há, por exemplo, uma família exclusiva para itens do tipo "*pé de cadeira*" que são todos intercambiáveis. Como trata-se de um caso ilustrativo, no entanto, não foram geradas todas as combinações para simplificação da apresentação.

Pelo fato apresentado da "especialização" das famílias, pode-se apresentar a estrutura de uma cadeira ou de uma mesa a partir das famílias, conforme as figuras 7.1 e 7.2 a seguir:

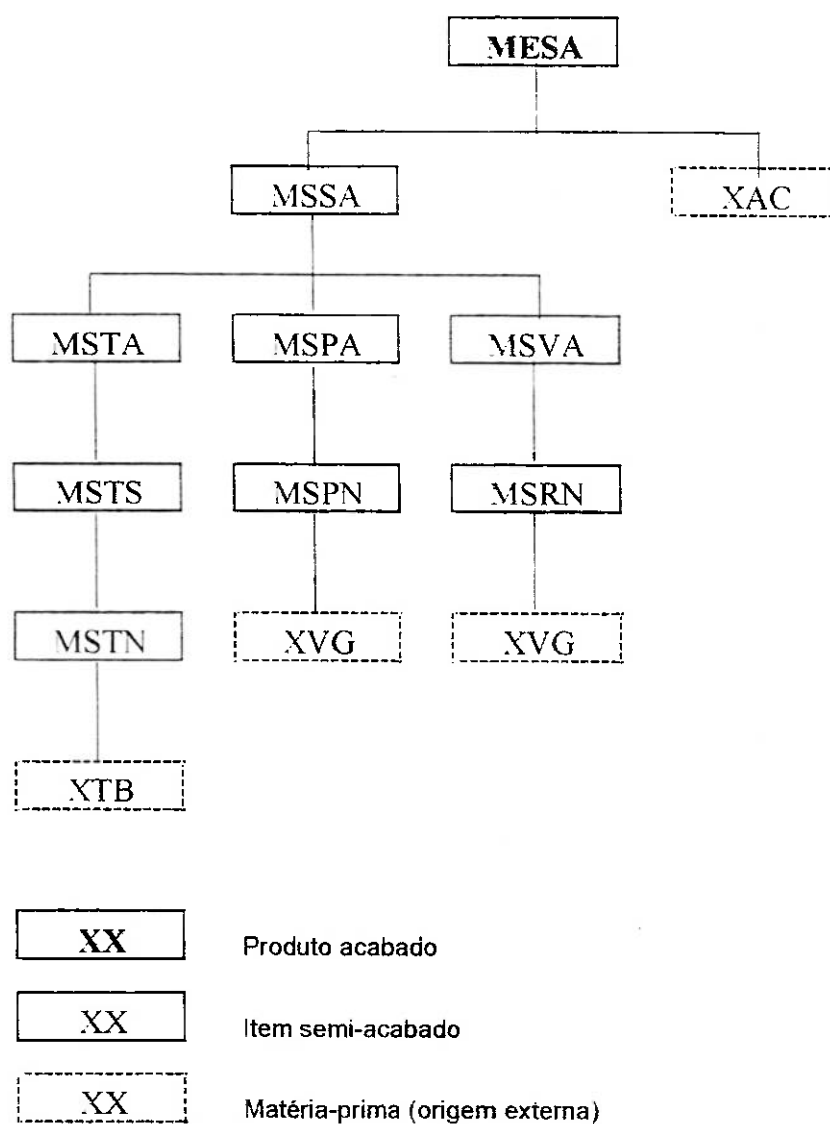
Estrutura genérica de um item da família "CADR" *



* Para melhor efeito ilustrativo, itens de fixação (cola, pregos e parafusos) foram suprimidos da estrutura

Figura 7.1 – Estrutura de itens da família CADR (Cadeiras)

Estrutura genérica de um item da família "MESA" *



* Para melhor efeito ilustrativo, itens de fixação (cola, pregos e parafusos) foram suprimidos da estrutura

Figura 7.2 – Estrutura de itens da família MESA (Mesas)

A relação completa com descrição de famílias está no anexo D.

7.1.2 ITENS

De acordo com as estruturas de item apresentadas nas figuras 7.1 e 7.2, pode-se comentar a composição dos itens das referidas famílias (a relação completa de itens da *industria* esta no anexo D).

Para melhor compreensão da composição de cada item, a codificação foi feita seguindo-se alguns critérios pré-definidos. Ou seja, dado um item, é possível, pelo código, determinar quais são os “sub-itens” (semi-acabados) que o compõem e retornar na estrutura até as matérias-primas.

Cadeiras

Uma cadeira é composta por uma cadeira “semi-acabada” (madeira crua, sem pintura ou verniz) e o “acabamento” (pintura ou verniz). A cadeira “semi-acabada” (que é um item que pode ser estocado! – como qualquer outro), por sua vez, é composta por um *assento, pés, travessas para os pés, estrutura de encosto e encosto*.

Cada uma partes pode ser feita em dois materiais diferentes: peroba e pinho. Os pés, assim como a estrutura de encosto, podem ser simples (sem adornos) ou trabalhados (torneados) ou ainda podem ter seção retangular.

Mesas

As produção de uma mesa segue a mesma lógica da cadeira, sendo a mesa “semi-acabada” formada por um *tampo, pés e travessas para os pés*.

Novamente podem ser utilizados dois materiais (peroba e pinho). Os pés, assim como as travessas, podem ter seção circular ou retangular. O tampo pode assumir diversas configurações: além do material pode-se variar a espessura e o formato (retangular ou redondo).

7.2 LISTA DE MATERIAIS

Conforme explicado nos capítulos 4 e 5, a lista de materiais de um item é configurada a partir da operação que o gera (correspondência um para um) e dos itens que servem de entrada para esta operação. Assim, não há uma tabela de lista de materiais. Há, no entanto, uma tabela de entradas de operações.

7.3 ROTEIROS

Da maneira como a modelagem foi feita (vide capítulos 4 e 5), os roteiros são *consequência* das operações e suas saídas (produto resultante) e entradas. Em outras palavras, como deve-se criar um item para cada etapa de um processo completo (ou seja, entre operações), não há formação de um roteiro! Ao contrário, cada item é feito a partir de uma única operação!!!

Assim, o roteiro de uma cadeira, por exemplo, é substituído por uma série de operações independentes (apesar da relação de precedência)...

7.4 OPERAÇÕES

Novamente, dada a “especialização” das famílias de item, as operações que geram itens de uma mesa família apresentam características muito semelhantes, conforme pode ser verificado no anexo D.

A formação dos códigos das operações segue uma regra pré-definida: dado que cada operação gera um e somente um item, o código desta operação será o prefixo “OP” e o código do item.

Os tempos de *set-up* foram definidos como zero caso o item anterior seja igual ou da mesma família que o da ordem que será processada.

A *economic part-period* foi definida com base no custo estimado do item e no custo estimado de *set-up* (tempo de *set-up* para item diverso multiplicado pela taxa horária do recurso de trabalho correspondente

7.5 RECURSOS DE TRABALHO

Retomando a definição do capítulo 4.1 sobre *recurso de trabalho*, no caso exemplo serão apenas máquinas. Mão-de-obra e outros recursos são considerados abundantes. As máquinas escolhidas – normalmente encontradas em “empresas reais” – foram ainda agrupadas em 5 *centros de trabalho*, conforme apresentado no anexo D

8. RESULTADOS E CONCLUSÕES

Sem que fossem alteradas as características principais do modelo (famílias e itens, lista de materiais, roteiros, operações e recursos de trabalho), foram criadas três configurações diferentes, variando-se a regra para seqüenciação das filas nos recursos de trabalho e a quantidade disponível dos mesmos. O sistema foi rodado para as três situações para posterior comparação dos resultados.

8.1 CONFIGURAÇÕES DO MODELO

As três configurações utilizadas foram as seguintes³²:

- **Primeira configuração**

- *Regra a seqüenciação*: menor data de entrega
- *Disponibilidade de recursos*:
 - Mesa de operações: 3
 - Lixadeira plana: 1
 - Torno manual: 1

- **Segunda configuração**

- *Regra a seqüenciação*: menor data de entrega
- *Disponibilidade de recursos*:
 - Mesa de operações: 4
 - Lixadeira plana: 2
 - Torno manual: 1

- **Terceira configuração**

- *Regra a sequenciação*: menor folga dinâmica
- *Disponibilidade de recursos*:
 - Mesa de operações: 4
 - Lixadeira plana: 2
 - Torno manual: 1

8.2 RESULTADOS

Os relatórios gerados pelas diferentes configurações estão apresentados no anexo B. Apenas os relatórios *Cumprimento de datas de entrega*, *Performance de recursos de trabalho* e *Ocupação de recursos de trabalho* foram acrescentados para os três casos (na ordem 1^a, 2^a e 3^a configuração). Os demais foram colocados apenas da primeira configuração.

Resumindo os resultados, vale ressaltar os alguns comentários:

1. A comparação entre as duas regras utilizadas para a mesma disponibilidade indicam que ambas apresentam mesmo atraso máximo e atraso médio bem próximos. No entanto, a *menor data de entrega* gerou menor número de atrasos.
2. O aumento na disponibilidade de recursos foi feita com base no relatório *Ocupação de recursos*, para aqueles que apresentavam altos índices. A redução dos atrasos foi significativa da primeira para a segunda configuração.
3. Os tempos em fila foram drasticamente reduzidos com aumento da disponibilidade nos recursos de alta ocupação

⁸² A disponibilidade dos demais recursos não está indicada por serem iguais nas três configurações

8.3 CONCLUSÕES

Um trabalho como este cujo foco é o desenvolvimento de um *sistema* tende a gerar mais um *produto* do que conclusões. No entanto, este sistema pode ser utilizado como “pesquisa” de como operar a manufatura de uma empresa, localização de recursos críticos, etc.

Muito já se pesquisou sobre seqüenciação a partir de regras de liberação que geraram resultados que apresentavam uma correlação entre regras e resultados. Por exemplo:

- Utilizando-se o *menor tempo de processo* obtém-se o menor tempo médio de fluxo
- Regras que consideram a data de chegada (FIFO) devem reduzir o desvio padrão do tempo em fluxo, o que não ocorre com o *menor tempo de processo*
- Utilizando-se a *menor data de entrega* deve-se melhorar o atraso máximo

Estes são alguns exemplos, que, apesar de “tenderem” a estes resultados, não necessariamente será sempre assim!

9. BIBLIOGRAFIA

9.1 REFERÊNCIAS BIBLIOGRÁFICAS

CORRÊA, H.L.; GIANESI, I.; CAON, M. – **Planejamento, Programação e Controle da Produção**. s.ed., editora Atlas. São Paulo, 1997

LAW, Averill M.; KELTON, W. D. – **Simulation Modeling & Analysis**. 2^a ed., editora McGraw Hill. s.l., 1991

HAX, Arnold C.; CANDEA, Dan – **Production And Inventory Management**. s.ed. editora Prentice Hall/Englewood Cliffs, New Jersey, 1984

ORLICKY, Joseph – **Materials Requirement Planning**. s.ed. editora McGraw Hill. New York, 1975

PACHECO, Ricardo F.; SANTORO, Miguel C. – **Proposta de classificação hierarquizada dos modelos de solução para o problema de *job shop scheduling***. Gestão & Produção, 1999

SILVER, Edward A.; PYKE, David F.; PETERSON, Rein – **Inventory Management and Production Planning and Scheduling**. 3^a ed., editora John Wiley & Sons, 1998

SLACK, Nigel; CHAMBERS, Stuart; HARLAND, Christine; HARRISON, Alan; JOHNSTON, Robert – **Administração da Produção**. editora Atlas/São Paulo, 1^a ed. 1997

SPENIK, Mark; SLEDGE, Orryn – **Microsoft SQL Server 6.5 DBA: Survival Guide**. 2^a ed. editora Sams, 1996

STEVENSON, William J. – **Production/Operations Management** – editora Irwin, 5^a edição, 1996

9.2 BIBLIOGRAFIA RECOMENDADA

- ANTHONY, R. N. – **Planning and Control Systems: A Framework for Analysis.** Harvard University Press, Cambridge, Mass., 1965
- BUFFA, E. S.; MILLER, J. G. – **Production-Inventory Systems: Planning and Control.** Irwin, Homewood, Ill., 1979
- BUFFA, E. S.; TAUBERT, W. H. – **Production-Inventory Systems: Planning and Control.** Irwin, Homewood, Ill., 1972
- CONWAY, R. W.; MAXWELL, W. L.; MILLER, L. W. – **Theory of Scheduling –** Addison-Wesley, Reading, Mass., 1967
- HAX, A. C.; MEAL, H. C. – **Hierarchical Integration of Production Planning and Scheduling.** Amsterdam, 1975
- JOHNSON, A. J.; MONTGOMERY, D. C. – **Operations Research in Production Planning, Scheduling and Inventory Control.** Wiley, New York, 1974
- MEAL, H. C. – **Manufacturing Control and Material Requirements Planning.** presented at the *Seventh Annual Meeting of the American Institute for Decision Sciences*, Cincinnati, 1975
- SCHIMIDT, J. W.; TAYLOR, R. E. – **Simulation and Analysis of Industrial Systems.** Richard D. Irwin, Homewood, Ill. (1970)
- VOLLMAN, T. E.; BERRY, W. L.; WHYBARK, D. C. – **Manufacturing planning and control systems.** 3. ed Irwin, 1992
- WIGHT, Oliver – **Manufacturing Resources Planning: MRP II** – Oliver Wight Ltd., 1984

ANEXOS

A - Código de Programação

B – Formulários

C – Relatórios

D – Dados do exemplo

E – Estrutura do banco de dados

ANEXO A - CÓDIGO DE PROGRAMAÇÃO

Fase 0:

Option Explicit

Const mcModulo As String = "Fase 0"

'Esta função faz as inicializações e verificações necessárias para
'garantir a consistência da simulação e evitar erros

Sub InicializarModelo(pForm As Form, pDB As Database)

'Constantes

Const cRotuloStatus As String = "lblStatus"

Const cRotuloSubStatus As String = "lblSubStatus"

Const cMensLLC As String = "Atualizando classificação LLC"

Const cMensErros As String = "Verificando consistência dos dados"

Const cMensLimpar As String = "Apagando dados das simulações antigas"

Const cMensInv As String = "Importando inventario inicial"

Const cMensEmBranco As String = ""

Const cNomeFunc As String = "InicializarModelo"

'Variáveis

Dim lAcaoPosVerificacao As Byte

'----

On Error GoTo ErroDesconhecido

'Apagar o conteúdo prévio da tabela de verificação

Call ApagarTabelaVerificacao(pDB)

'Atualizar a classificação LLC

pForm.Controls(cRotuloSubStatus).Caption = cMensLLC

pForm.Repaint

'Verificar operações sem entrada

Call VerificarEntradaSaidaOperacao(pForm, pDB)

'Fazer atualização

Call AtualizarLLC(pDB)

'Verificar se há erros cadastrais no modelo

pForm.Controls(cRotuloSubStatus).Caption = cMensErros

pForm.Repaint

Call VerificarModelo(pForm, pDB)

'Apagar o conteúdo das tabelas transitórias (utilizadas na simulação)

pForm.Controls(cRotuloSubStatus).Caption = cMensLimpar

pForm.Repaint

Call LimparTabelasDeSimulacao(pDB)

'Importar o inventario do banco de dados da organização

pForm.Controls(cRotuloSubStatus).Caption = cMensInv

pForm.Repaint

Call ImportarInventarioInicial(pDB)

'Atualizar status

pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco

pForm.Repaint

'Encerrar função

Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err. Err.Description, cNomeFunc, mcModulo)

End Sub

'Funções locais

'Esta função apaga o conteúdo prévio da tabela de verificação
Private Sub ApagarTabelaVerificacao(pDB As Database)

'Constantes

Const cConsultaLimparVerificacao = "qry_LimparVerificacao"

Const cNomeFunc As String = "ApagarTabelaVerificacao"

'Variáveis

Dim myQRY As QueryDef

'----

On Error GoTo ErroDesconhecido

'Executar a consulta

Set myQRY = pDB.QueryDef(cConsultaLimparVerificacao)

myQRY.Execute

'Fechar os objetos

myQRY.Close

'Encerrar função

Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err. Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica a existência de alguma operação cuja entrada é também a saída
Private Sub VerificarEntradaSaidaOperacao(pfForm As Form, pDB As Database)

'Constantes

Const cConsultaListarEntradaIgualSaida As String = "qry_InicVerif_ListarEntradaIgualSaida"

Const cCampoCodOperacao As String = "OpEn_OpCod"

Const cCampoNomeOperacao As String = "Opr_Nome"

Const cTabOperacaoEntrada As String = "OperacaoEntrada"

Const cMens As String = "O produto de saída da operação é também entrada!!!"

Const cNomeFunc As String = "VerificarEntradaSaidaOperacao"

'Variáveis

Dim myQRY As QueryDef

Dim myRS As Recordset

'----

On Error GoTo ErroDesconhecido

'Abrir recordset de verificação

Set myQRY = pDB.QueryDef(cConsultaListarEntradaIgualSaida)

Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Analisar resultado da consulta

'Se houver registros indicando erro

If myRS.EOF Then

GoTo FecharObjetos

```

Else
    'Gerar mensagens de erro
    Do Until myRS.EOF
        Call InserirMensagemErro(pDB:=pDB, psTabela:=cTabOperacaoEntrada, _
            psErro:=cMens, _
            psCampoChave:=cCampoCodOperacao, _
            psRegistroChave:=myRS(cCampoCodOperacao), _
            psRegistroDesc:=myRS(cCampoNomeOperacao))
        myRS.MoveNext
    Loop
    'Encerrar simulação
    Call EncerrarPorErro(pForm, pDB)
End If

FecharObjetos:

'Fechar os objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função atualiza a tabela de Low Level Coding (LLC)
Private Sub AtualizarLLC(pDB As Database)

'Constantes da tabela e de tabelas relacionadas
Const cConsultaLimparLLC As String = "qry_LimparLLC"
Const cConsultaIniciarLLC_Nivel_0 As String = "qry_IniciarLLC_Nivel_0"
Const cConsultaAtualizarLLC As String = "qry_AtualizarLLC"
Const cConsultaParametroNivel As String = "Nivel"
Const cNomeFunc As String = "AtualizarLLC"

'Variáveis
Dim iNivel As Integer
Dim iRegistrosAfetados As Integer
Dim myQRY As QueryDef
'----

On Error GoTo ErroDesconhecido

'Apagar os dados atuais da tabela LLC
Set myQRY = pDB.QueryDefs(cConsultaLimparLLC)
myQRY.Execute

'Inserir todos os itens na tabela LLC, com default de nivel = 0
Set myQRY = pDB.QueryDefs(cConsultaIniciarLLC_Nivel_0)
myQRY.Execute

'Em loop, baixa o nivel de cada item
'Definir a consulta
Set myQRY = pDB.QueryDefs(cConsultaAtualizarLLC)

'Definir valores iniciais
iRegistrosAfetados = 1          'Forçar a entrada no loop
iNivel = 0

'Entrar em loop, baixando o nivel dos itens
Do Until iRegistrosAfetados = 0

```

```

        myQRY.Parameters(cConsultaParametroNivel) = iNivel
        myQRY.Execute
        iRegistrosAfetados = myQRY.RecordsAffected
        iNivel = iNivel + 1
    Loop

'Fechar os objetos
    myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função faz uma série de verificações quanto a consistência dos dados
Private Sub VerificarModelo(pForm As Form, pDB As Database)

'Constantes
Const cNomeFunc As String = "VerificarModelo"

'Variáveis
Dim bErroLocalizado As Boolean
'---

On Error GoTo ErroDesconhecido

'Inicializar variáveis
    bErroLocalizado = False

'Verificar existência de encomendas
    Call VerificarEncomendas(pDB)

'Verificar dias com disponibilidade de trabalho definida coerentemente
    Call VerificarDisponibilidadeTrabalho(bErroLocalizado, pDB)

'Verificar operações com incoerência nas restrições de lotes
    Call VerificarRestricaoLote(bErroLocalizado, pDB)

'Verificar itens sem operação correspondente
    Call VerificarItemSemOperacao(bErroLocalizado, pDB)

'Verificar repetições das ordens de entrada das operações
    Call VerificarOperacaoSemEntrada(bErroLocalizado, pDB)

'Verificar repetições das ordens de entrada das operações
    Call VerificarOrdemEntradaOperacao(bErroLocalizado, pDB)

'Comunicar e perguntar se deseja ver erros localizados
    If bErroLocalizado = True Then
        Call EncerrarPorErro(pForm, pDB)
    End If

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função apaga o conteúdo de algumas tabelas utilizadas para a nova simulação

```

```

Private Sub LimparTabelasDeSimulacao(pDB As Database)

'Constantes
Const cConsultaLimparEncomendaInterna As String = "qry_LimparEncomendaInterna"
Const cConsultaLimparOrdemDeProducao As String = "qry_LimparOrdemDeProducao"
Const cConsultaLimparRelacionamentoDeOrdens As String = "qry_LimparRelacionamentoDeOrdens"
Const cConsultaLimparFila As String = "qry_LimparFila"
Const cConsultaLimparInventarioDinamico As String = "qry_LimparInventarioDinamico"
Const cConsultaLimparPrograma As String = "qry_LimparPrograma"
Const cConsultaLimparProgramaDataHora As String = "qry_LimparProgramaDataHora"
Const cNomeFunc As String = "LimparTabelasDeSimulacao"

'Variáveis
Dim myQRY As QueryDef
'-----

On Error GoTo ErroDesconhecido

'Apagar o conteúdo da tabela 'EncomendaInterna'
Set myQRY = pDB.QueryDefs(cConsultaLimparEncomendaInterna)
myQRY.Execute

'Apagar o conteúdo da tabela 'OrdemDeProducao'
Set myQRY = pDB.QueryDefs(cConsultaLimparOrdemDeProducao)
myQRY.Execute

'Apagar o conteúdo da tabela 'RelacionamentoDeOrdens'
Set myQRY = pDB.QueryDefs(cConsultaLimparRelacionamentoDeOrdens)
myQRY.Execute

'Apagar o conteúdo da tabela 'Fila'
Set myQRY = pDB.QueryDefs(cConsultaLimparFila)
myQRY.Execute

'Apagar o conteúdo da tabela 'InventarioDinamico'
Set myQRY = pDB.QueryDefs(cConsultaLimparInventarioDinamico)
myQRY.Execute

'Apagar o conteúdo da tabela 'Programa'
Set myQRY = pDB.QueryDefs(cConsultaLimparPrograma)
myQRY.Execute

'Apagar o conteúdo da tabela 'Programa_DataHora'
Set myQRY = pDB.QueryDefs(cConsultaLimparProgramaDataHora)
myQRY.Execute

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função importa o inventário inicial de outro banco de dados
Private Sub ImportarInventarioInicial(pDB As Database)

End Sub

'Sub-funções locais
'=====

```

'Esta função verifica a existência de encomendas
Private Sub VerificarEncomendas(pDB As Database)

'Constantes

Const cConsultaContarEncomendas As String = "qry_InicVerif_ContarEncomendas"

Const cMens As String = "Não é possível simular: não há encomendas!!!"

Const cTitulo As String = "Auditoria"

Const cNomeFunc As String = "VerificarEncomendas"

'Variáveis

Dim myQRY As QueryDef

Dim myRS As Recordset

'----

On Error GoTo ErroDesconhecido

'Abrir consulta de verificação

Set myQRY = pDB.QueryDef(cConsultaContarEncomendas)

Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Analisar contagem

If myRS(0) = 0 Then

'Comunicar erro

MsgBox cMens, vbExclamation, cTitulo

'Fechar objetos

myRS.Close

myQRY.Close

pDB.Close

'Encerrar simulação

End

Else

'Fechar objetos

myRS.Close

myQRY.Close

End If

'Encerrar função

Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica o cadastro de disponibilidade de trabalho

Private Sub VerificarDisponibilidadeTrabalho(ByRef pbErroLocalizado As Boolean, _
pDB As Database)

'Constantes

Const cConsultaContarDisponibilidade As String = "qry_InicVerif_ContarDisponibilidade"

Const cTabDisponibilidade As String = "Disponibilidade"

Const cTextoEmBranco As String = ""

Const cMens As String = "Há um ou mais erros na disponibilidade de trabalho. " &

"Verifique se todos os dias estão cadastrados ou se " &

"há algum dia cujo início e posterior ao fim!"

Const cNomeFunc As String = "VerificarDisponibilidadeTrabalho"

'Variáveis

Dim myQRY As QueryDef

Dim myRS As Recordset

```

'-----

On Error GoTo ErroDesconhecido

'Abrir consulta de verificação
Set myQRY = pDB.QueryDef(cConsultaContarDisponibilidade)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Verificar se contagem é igual a 7 (7 dias na semana)
If myRS(0) <> 7 Then
    pbErroLocalizado = True
    Call InserirMensagemErro(pDB:=pDB, psTabela:=cTabDisponibilidade, _
        psErro:=cMens)
End If

'Fechar os objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica a existência de operações com incoerência nas restrições de lote
Private Sub VerificarRestricaoLote(ByRef pbErroLocalizado, pDB As Database)

'Constantes
Const cConsultaListarIncoerenciaLote As String = "qry_InicVerif_ListarIncoerenciaLote"
Const cCampoCodOperacao As String = "Opr_Codigo"
Const cCampoNomeOperacao As String = "Opr_Nome"
Const cTabOperacao As String = "Operacao"
Const cMens As String = "Há incoerência nas restrições de tamanho de lote. " & _
    "Verifique os valores de lote minimo, lote maximo " & _
    "e quantidade multipla!!!"
Const cNomeFunc As String = "VerificarRestricaoLote"

'Variáveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'-----

On Error GoTo ErroDesconhecido

'Abrir recordset de verificação
Set myQRY = pDB.QueryDef(cConsultaListarIncoerenciaLote)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Analisar resultado da consulta
'Se houver registros indicando erro
If myRS.EOF Then
    GoTo FecharObjetos
Else
    'Marcar flag de indicação de erro
    pbErroLocalizado = True
    'Gerar mensagens de erro
    Do Until myRS.EOF
        Call InserirMensagemErro(pDB:=pDB, psTabela:=cTabOperacao, _
            psErro:=cMens, _
            psCampoChave:=cCampoCodOperacao, _
            psRegistroChave:=myRS(cCampoCodOperacao), _

```

```

        psRegistroDesc:=myRS(cCampoNomeOperacao))
    myRS.MoveNext
Loop
End If

FecharObjetos:

'Fechar os objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica a existência de algum item sem operação correspondente
Private Sub VerificarItemSemOperacao(ByRef pbErroLocalizado As Boolean, _
    pDB As Database)

'Constantes
Const cConsultaListarItemSemOperacao As String = "qry_InicVerif_ListarItemSemOperacao"
Const cCampoCodItem As String = "Item_Codigo"
Const cCampoNomeItem As String = "Item_Nome"
Const cTabItem As String = "Item"
Const cMens As String = "Não há operação correspondente para o item!!!"
Const cNomeFunc As String = "VerificarItemSemOperacao"

'Variáveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'----

On Error GoTo ErroDesconhecido

'Abrir recordset de verificação
Set myQRY = pDB.QueryDef(cConsultaListarItemSemOperacao)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Analisar resultado da consulta
'Se houver registros indicando erro
If myRS.EOF Then
    GoTo FecharObjetos
Else
    'Marcar flag de indicação de erro
    pbErroLocalizado = True
    'Gerar mensagens de erro
    Do Until myRS.EOF
        Call InserirMensagemErro(pDB:=pDB, psTabela:=cTabItem, _
            psErro:=cMens, _
            psCampoChave:=cCampoCodItem, _
            psRegistroChave:=myRS(cCampoCodItem), _
            psRegistroDesc:=myRS(cCampoNomeItem))
        myRS.MoveNext
    Loop
End If

FecharObjetos:

'Fechar os objetos
myQRY.Close

```

```

myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica a existência de alguma operação sem entrada correspondente
Private Sub VerificarOperacaoSemEntrada(ByRef pbErroLocalizado As Boolean, _
    pDB As Database)

'Constantes
Const cConsultaListarOperacaoSemEntrada As String = "qry_InicVerif_ListarOperacaoSemEntrada"
Const cCampoCodOperacao As String = "Opr_Codigo"
Const cCampoNomeOperacao As String = "Opr_Nome"
Const cTabOperacaoEntrada As String = "OperacaoEntrada"
Const cMens As String = "Não há entrada correspondente para a operação!!!"
Const cNomeFunc As String = "VerificarOperacaoSemEntrada"

'Variáveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'-----

On Error GoTo ErroDesconhecido

'Abrir recordset de verificação
Set myQRY = pDB.QueryDef(cConsultaListarOperacaoSemEntrada)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Analisar resultado da consulta
'Se houver registros indicando erro
If myRS.EOF Then
    GoTo FecharObjetos
Else
    'Marcar flag de indicação de erro
    pbErroLocalizado = True
    'Gerar mensagens de erro
    Do Until myRS.EOF
        Call InserirMensagemErro(pDB:=pDB, psTabela:=cTabOperacaoEntrada, _
            psErro:=cMens, _
            psCampoChave:=cCampoCodOperacao, _
            psRegistroChave:=myRS(cCampoCodOperacao), _
            psRegistroDesc:=myRS(cCampoNomeOperacao))
        myRS.MoveNext
    Loop
End If

FecharObjetos:

Fechar os objetos
myQRY.Close
myRS.Close

Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

```

```

'Esta função verifica a existência de alguma operação sem entrada correspondente
Private Sub VerificarOrdemEntradaOperacao(ByRef pbErroLocalizado As Boolean, _
    pDB As Database)

'Constantes
Const cConsultaListarEntradaRepetida As String = "qry_InicVerif_ListarRepeticaoOrdemEntrada"
Const cCampoCodOperacao As String = "OpEn_OpCod"
Const cCampoNomeOperacao As String = "Opr_Nome"
Const cTabOperacaoEntrada As String = "OperacaoEntrada"
Const cMens As String = "Há número de ordem repetida para a operação!!!"
Const cNomeFunc As String = "VerificarOrdemEntradaOperacao"

'Variáveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'----

On Error GoTo ErroDesconhecido

'Abrir recordset de verificação
Set myQRY = pDB.QueryDef(cConsultaListarEntradaRepetida)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Analisar resultado da consulta
'Se houver registros indicando erro
If myRS.EOF Then
    GoTo FecharObjetos
Else
    'Marcar flag de indicação de erro
    pbErroLocalizado = True
    'Gerar mensagens de erro
    Do Until myRS.EOF
        Call InserirMensagemErro(pDB:=pDB, psTabela:=cTabOperacaoEntrada, _
            psErro:=cMens, _
            psCampoChave:=cCampoCodOperacao, _
            psRegistroChave:=myRS(cCampoCodOperacao), _
            psRegistroDesc:=myRS(cCampoNomeOperacao))
        myRS.MoveNext
    Loop
End If

FecharObjetos:

'Fechar os objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função insere um registro na tabela de verificação
Private Sub InserirMensagemErro(pDB As Database, _
    ByVal psTabela As String, _
    ByVal psErro As String, _
    Optional ByVal psCampoChave As String, _
    Optional ByVal psRegistroChave As String, _
    Optional ByVal psRegistroDesc As String)

```

'Constantes

```
Const cConsultaInserirMensagemErro As String = "qry_InserirMensagemDeErro"
Const cConsultaParametroTabela As String = "Tabela"
Const cConsultaParametroCampoChave As String = "CampoChave"
Const cConsultaParametroRegistroChave As String = "RegistroChave"
Const cConsultaParametroDescricaoDoRegistro As String = "DescricaoDoRegistro"
Const cConsultaParametroMensagemDeErro As String = "MensagemDeErro"
Const cNomeFunc As String = "InserirMensagemErro"
```

'Variáveis

```
Dim myQRY As QueryDef
```

'----

On Error GoTo ErroDesconhecido

'Executar a consulta

```
Set myQRY = pDB.QueryDefs(cConsultaInserirMensagemErro)
myQRY.Parameters(cConsultaParametroTabela) = psTabela
myQRY.Parameters(cConsultaParametroCampoChave) = If(IsMissing(psCampoChave), Null, psCampoChave)
myQRY.Parameters(cConsultaParametroRegistroChave) = If(IsMissing(psRegistroChave), Null,
psRegistroChave)
myQRY.Parameters(cConsultaParametroDescricaoDoRegistro) = If(IsMissing(psRegistroDesc), Null,
psRegistroDesc)
myQRY.Parameters(cConsultaParametroMensagemDeErro) = psErro
myQRY.Execute
```

'Fechar os objetos

```
myQRY.Close
```

'Encerrar função

```
Exit Sub
```

ErroDesconhecido:

```
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

```
End Sub
```

'Esta função encerra a simulação por erro

```
Private Sub EncerrarPorErro(pForm As Form, pDB As Database)
```

'Constantes

```
Const cTitulo As String = "Auditoria"
Const cMens As String = "Não sera possivel realizar a simulação. " & _
    "Foram encontrados erros de cadastro no modelo." & _
    "Deseja ver quais foram os erros?"
Const cRotuloStatus As String = "lblStatus"
Const cRotuloSubStatus As String = "lblSubStatus"
Const cMensEmBranco As String = ""
Const cTabVerificacao As String = "Verificacao"
Const cNomeFunc As String = "EncerrarPorErro"
```

'Variáveis

```
Dim iResp As Byte
```

'----

On Error GoTo ErroDesconhecido

'Perguntar se deseja-se exibir os erros

```
iResp = MsgBox(cMens, vbYesNo, cTitulo)
```

'Apagar status de simulação

```
pForm.Controls(cRotuloStatus).Caption = cMensEmBranco
pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco
pForm.Repaint
```

```

'Exibir erros se solicitado
  If iResp = vbYes Then
    DoCmd.OpenForm cTabVerificacao
  End If
'Encerrar simulação
  pIDB.Close
  End

'Encerrar função
Exit Sub

ErroDesconhecido:
  Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

```

Fase 1:

Option Explicit

'Vetor que armazena as disponibilidades de funcionamento por dia da semana

Private Type tDisponibilidadeDiana

iDia As Integer	'Dia da semana
dHoraInicio As Date	'Hora de inicio
dHoraFim As Date	'Hora de encerramento
nHorasUteis As Double	'Horas uteis no dia
nHorasUteisAcumuladas As Double	'Horas uteis acumuladas até o dia

End Type

'Vetor que armazena as características de cada necessidade (encomenda interna)

Private Type tEncomendaInterna

iChave As Long	'Chave da encomenda interna
iOrdemRelacionada As Long	'Ordem de onde veio a necessidade
sItem As String	'Item necessario
nQuantidade As Single	'Quantidade necessaria
nDataMax As Single	'Data da necessidade
nEPP As Single	'Economic part-period
nCustoUnitInv As Single	'Custo unitário de inventário
nQtdeMin As Single	'Lote minimo do item
nQtdeMax As Single	'Lote máximo do item
nQtdeMultiplo As Single	'Quantidades multiplas para lote do item

End Type

'Vetor que armazena os relacionamento entre ordens (onde será alocada a produção)

Private Type tRelacionamentoDeOrdens

iReferencia As Long	'Para qual ordem será alocada
nQuantidade As Double	'Quantidade que será alocada

End Type

Const mcModulo As String = "Fase 1"

'Esta função gera as ordens de produção e forma as filas

```

Sub PreparaElementosDeSimulacao(ByVal pDataHoraInicioSimulacao As Date, _
    ByRef pIniNormalizado As Single, _
    pIDB As Database)

```

'Constantes

Const cNomeFunc As String = "PreparaElementosDeSimulacao"

'----

On Error GoTo ErroDesconhecido

'Gerar as encomendas internas diretas (entradas do usuário) _

 A transferência é feita com os valores da entrada do usuário _

 A normalização das datas é feita em etapa posterior

 Call TransferirEncomendas(pDB)

'Normalizar as datas das encomendas diretas transferidas

 Call NormalizarEncomendas(pdDataHoraInicioSimulacao, pnInicioNormalizado, pDB)

'Agrupar as encomendas internas diretas e gerar ordens de produção

 Call GerarOrdemDeProducao(0, pDB)

'Gerar as encomendas internas indiretas _

 (necessidades para as encomendas internas diretas)

 Call GerarOrdemDeProducaoIndireta(pnInicioNormalizado, pDB)

'Montar a base das filas

 Call GerarFilas(pnInicioNormalizado, pDB)

'Encerrar função

Exit Sub

ErroDesconhecido:

 Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Funções locais

=====

'Transferir as encomendas (entradas do usuário) para a tabela Encomendas Internas

Private Sub TransferirEncomendas(pDB As Database)

'Constantes

Const cConsultaGerarEncomendaInterna As String = "qry_GerarEncomendaInternaDireta"

Const cNomeFunc As String = "TransferirEncomendas"

'Variáveis

Dim mvQRY As QueryDef

'----

On Error GoTo ErroDesconhecido

'Rodar a consulta de transferência

 Set mvQRY = pDB.QueryDefs(cConsultaGerarEncomendaInterna)

 mvQRY.Execute

'Fechar os objetos

 mvQRY.Close

'Encerrar função

Exit Sub

ErroDesconhecido:

 Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função gera as encomendas internas iniciais, normalizando as datas de entrega

Private Sub NormalizarEncomendas(ByVal pdDataHoraInicioSimulacao As Date, _

 ByRef nInicioNormalizado As Single, _

 pDB As Database)

```

'Constantes
Const cConsultaDataEncomendaInterna As String = "qry_DataEncomendaInterna"
Const cCampoDataMax As String = "EncIn_DataMax"
Const cFatorDeConversaoDeTempo As Double = 24 * 1000
Const cNomeFunc As String = "NormalizarEncomendas"

'Variaves
Dim tInfoDisponibilidade(vbSunday - 1 To vbSaturday) As tDisponibilidadeDiaria

Dim nValorAntigo As Double
Dim nValorNormalizado As Double
Dim dInicioPrimeiraSemana As Date
Dim nDisponibilidadeSemanal As Double

Dim myRS As Recordset
Dim myQRY As QueryDef
'----

On Error GoTo ErroDesconhecido

'Definir o momento inicial da primeira semana (referência de tempo)
dInicioPrimeiraSemana = DateValue(pdDataHoraInicioSimulacao - _
    WeekDay(pdDataHoraInicioSimulacao, vbSunday) + 1)

'Montar um vetor com a disponibilidade diária
nDisponibilidadeSemanal = VerificarDisponibilidade(pDB, tInfoDisponibilidade, _
    cFatorDeConversaoDeTempo)

'Abrir a tabela de encomendas internas em um recordset para atualização
Set myQRY = pDB.QueryDef(cConsultaDataEncomendaInterna)
Set myRS = myQRY.OpenRecordset(dbOpenDynaset)
'Fechar a consulta
myQRY.Close

'Normalizar a data/hora de inicio da simulação
nInicioNormalizado = NormalizarData(dInicioPrimeiraSemana, pdDataHoraInicioSimulacao, _
    nDisponibilidadeSemanal, _
    tInfoDisponibilidade(WeekDay(pdDataHoraInicioSimulacao, vbSunday)), nHorasUteisAcumuladas, _
    tInfoDisponibilidade(WeekDay(pdDataHoraInicioSimulacao, vbSunday)), dHoraInicio, _
    tInfoDisponibilidade(WeekDay(pdDataHoraInicioSimulacao, vbSunday)), dHoraFim, _
    cFatorDeConversaoDeTempo)

'Normalizar a data/hora de cada registro
Do Until myRS.EOF
    'Valor não normalizado
    nValorAntigo = myRS(cCampoDataMax)
    'Valor normalizado
    nValorNormalizado = NormalizarData(dInicioPrimeiraSemana, CDate(nValorAntigo), _
        nDisponibilidadeSemanal, _
        tInfoDisponibilidade(WeekDay(nValorAntigo, vbSunday)), nHorasUteisAcumuladas, _
        tInfoDisponibilidade(WeekDay(nValorAntigo, vbSunday)), dHoraInicio, _
        tInfoDisponibilidade(WeekDay(nValorAntigo, vbSunday)), dHoraFim, _
        cFatorDeConversaoDeTempo)
    'Preparar o recordset para edição
    myRS.Edit
    'Valor normalizado, com relação ao inicio da simulação
    myRS(cCampoDataMax) = nValorNormalizado
    'Atualiza o recordset
    myRS.Update
    'Próximo registro
    myRS.MoveNext
Loop

```

'Fechar os objetos
myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função gera as ordens de produção para os itens de um dado nível I.L.C

Private Sub GerarOrdemDeProducao(ByVal piNivel As Integer, pIDB As IDatabase)

'Constantes

Const cConsultaMarcarEncomendaInterna As String = "qry_MarcarEncomendaOrdenada"

Const cConsultaParametroNivel As String = "Nivel"

Const cNomeFunc As String = "GerarOrdemDeProducao"

'VARIÁVEIS

'Vetores de armazenamento de dados

Dim tEncomendas() As tEncomendaInterna

Dim tRelacionamentos() As tRelacionamentoDeOrdens

'Consulta

Dim myQRY As QueryDef

'Propriedades "estáticas" da ordem de produção

Dim nDataOrdem As Double

'Data da ordem de produção

Dim sItem As String

'Item da ordem de produção

'Propriedades "dinâmicas" da ordem de produção

Dim nQtdeAcum As Double

'Quantidade da ordem de produção

Dim nDifEPP As Double

'Diferença entre custo de inventário e EPP

Dim nCustoInvAcum As Double

'Custo acumulado de inventário

'Variáveis auxiliares

Dim i As Integer

'Contador

Dim bFirmarOrdem As Boolean

'Flag disparando ordem de produção

Dim nFatorDeTempo As Double

'Fator de conversão de tempo

Dim nCustoExtra As Double

'Custo extra para testes

Dim nGanho As Double

'Ganho para testes

Dim bPrimeiroDaOrdem As Boolean

'Flag que indica se é primeira necessidade da ordem

Dim nAjustesDeLote As Single

'Ajustes nos lotes por quantidade ou múltiplo

On Error GoTo ErroDesconhecido

'Verificar a conversão de tempo

nFatorDeTempo = VerificarFatorDeTempo(pIDB)

'Abrir recordset de necessidades brutas

Call AbrirRecordsetNecessidadesBrutas(piNivel, tEncomendas, pIDB)

'Inicilizar variáveis

i = LBound(tEncomendas)

bPrimeiroDaOrdem = True

'Loop após firmar uma ordem de produção

Do Until i > UBound(tEncomendas)

'Valores que não se alteram para uma ordem de produção (estáticos)

sItem = tEncomendas(i).sItem

nDataOrdem = tEncomendas(i).nDataMax

'Valores que serão alterados na formação da ordem de produção (dinâmicos)

```

nQtdeAcum = tEncomendas(i).nQuantidade
nDifEPP = tEncomendas(i).nEPP
nCustoInvAcum = 0

'Montar vetor de relacionamento de ordens de produção
ReDim tRelacionamento(0 To 0)
tRelacionamento(0).iReferencia = tEncomendas(i).iOrdemRelacionada
tRelacionamento(0).nQuantidade = tEncomendas(i).nQuantidade

'Acreditar necessidades de produção seguintes, em "loop" até ocorrer _
alteração de item ou que a ordem seja firmada
ProximaEncomendaInterna:
'Se for última encomenda interna existente, vai direto para look-back test
If i = UBound(tEncomendas) Then GoTo LookBackTest

'Se inclusão de nova necessidade exceder o limite da operação (lote máximo)
If nQtdeAcum + tEncomendas(i + 1).nQuantidade > tEncomendas(i).nQtdeMax _
And tEncomendas(i).nQtdeMax <> 0 Then GoTo LookBackTest

'Se for última encomenda interna do item
If sltem <> tEncomendas(i + 1).sltem Then
'Verificar se e também a primeira encomenda interna da ordem
If bPrimeiroDaOrdem = True Then
'Firmar uma ordem se for também a primeira encomenda interna da ordem
GoTo FirmarOrdem
Else
'Fazer look-back test se não for a primeira encomenda interna da ordem
GoTo LookBackTest
End If
End If

'Verificar impacto da inclusão de nova necessidade na ordem
'Não pode exceder limite máximo do lote (quando houver)
'E deve diminuir o custo OU completar limite mínimo do lote
If (nQtdeAcum + tEncomendas(i + 1).nQuantidade <= tEncomendas(i).nQtdeMax _
Or tEncomendas(i).nQtdeMax = 0) _
And nDifEPP > Abs(tEncomendas(i + 1).nEPP - nCustoInvAcum - _
tEncomendas(i + 1).nQuantidade * _
tEncomendas(i + 1).nCustoUnitInv * nFatorDeTempo * _
(tEncomendas(i + 1).nDataMax - nDataOrdem)) _
Or nQtdeAcum < tEncomendas(i).nQtdeMin Then

'Se for interessante incluir mais uma necessidade no lote
'Incluir encomenda interna na ordem
i = i + 1
nQtdeAcum = nQtdeAcum + tEncomendas(i).nQuantidade
nCustoInvAcum = nCustoInvAcum + tEncomendas(i).nQuantidade * _
tEncomendas(i).nCustoUnitInv * nFatorDeTempo * _
(tEncomendas(i).nDataMax - nDataOrdem)
nDifEPP = Abs(tEncomendas(i).nEPP - nCustoInvAcum)

'Relacionamento de ordens
ReDim Preserve tRelacionamento(0 To UBound(tRelacionamento) + 1)
tRelacionamento(UBound(tRelacionamento)).iReferencia = _
tEncomendas(i).iOrdemRelacionada
tRelacionamento(UBound(tRelacionamento)).nQuantidade = _
tEncomendas(i).nQuantidade

'Há mais de uma encomenda interna relacionada com a ordem
bPrimeiroDaOrdem = False

Else 'Se inclusão NÃO deixar custo de inventário mais próximo do EPP:
LookAheadTest:

```

'Look-ahead test (se não for o último registro e se não houver mudança de item)

'Se for última encomenda interna, não é possível fazer look-ahead test
If $i \geq \text{UBound}(\text{tEncomendas}) - 1$ Then GoTo LookBackTest

'Se exceder o limite possível para operação (lote máximo)

If $\text{nQtdeAcum} + \text{tEncomendas}(i+1).\text{nQuantidade} > \text{tEncomendas}(i).\text{nQtdeMax}$ -
And $\text{tEncomendas}(i).\text{nQtdeMax} < 0$ Then GoTo LookBackTest

'Verificar ganhos e perdas

$\text{nGanho} = \text{tEncomendas}(i+2).\text{nQuantidade} * \text{tEncomendas}(i+2).\text{nCustoUnitInv} * \text{nFatorDeTempo} * (\text{tEncomendas}(i+2).\text{nDataMax} - \text{tEncomendas}(i+1).\text{nDataMax})$
 $\text{nCustoExtra} = \text{tEncomendas}(i+1).\text{nQuantidade} * \text{tEncomendas}(i+1).\text{nCustoUnitInv} * \text{nFatorDeTempo} * (\text{tEncomendas}(i+1).\text{nDataMax} - \text{nDataOrdem})$

'Verificar se operação representa ganho

If $\text{nCustoExtra} < \text{nGanho}$ Then

'Alterar dados para inclusão da necessidade na ordem a ser firmada

$i = i + 1$

$\text{nQtdeAcum} = \text{nQtdeAcum} + \text{tEncomendas}(i).\text{nQuantidade}$

Relacionamento de ordens

$\text{ReDim Preserve tRelacionamento}(0 \text{ To } \text{UBound}(\text{tRelacionamento}) + 1)$

$\text{tRelacionamento}(\text{UBound}(\text{tRelacionamento})).\text{iReferencia} = \text{tEncomendas}(i).\text{iOrdemRelacionada}$
 $\text{tRelacionamento}(\text{UBound}(\text{tRelacionamento})).\text{nQuantidade} = \text{tEncomendas}(i).\text{nQuantidade}$

'Há mais de uma encomenda interna relacionada com a ordem
 $\text{bPrimeiroDaOrdem} = \text{False}$

Firmar a ordem com inclusão da nova necessidade

GoTo FirmarOrdem

End If

LookBackTest:

'Look-back test (se não for a primeira necessidade da ordem)

If $\text{bPrimeiroDaOrdem} = \text{True}$ Then GoTo FirmarOrdem

'Se retirada de necessidade deixar ordem abaixo do limite mínimo

If $i < \text{UBound}(\text{tEncomendas})$ Then

If $\text{nQtdeAcum} + \text{tEncomendas}(i+1).\text{nQuantidade} < \text{tEncomendas}(i).\text{nQtdeMax}$ Then GoTo FirmarOrdem

End If

'Verificar ganhos e perdas

If $i < \text{UBound}(\text{tEncomendas})$ Then

'Se não for a última necessidade

$\text{nCustoExtra} = \text{tEncomendas}(i+1).\text{nQuantidade} * \text{tEncomendas}(i+1).\text{nCustoUnitInv} * \text{nFatorDeTempo} * (\text{tEncomendas}(i+1).\text{nDataMax} - \text{tEncomendas}(i).\text{nDataMax})$

Else

'Se for a última necessidade

$\text{nCustoExtra} = \text{tEncomendas}(i).\text{nEPP}$

End If

$\text{nGanho} = \text{tEncomendas}(i).\text{nQuantidade} * \text{tEncomendas}(i).\text{nCustoUnitInv} * \text{nFatorDeTempo} * (\text{tEncomendas}(i).\text{nDataMax} - \text{nDataOrdem})$

'Verificar se operação representa ganho

If $\text{nCustoExtra} < \text{nGanho}$ Then

$\text{nQtdeAcum} = \text{nQtdeAcum} - \text{tEncomendas}(i).\text{nQuantidade}$

$i = i - 1$

```

'Relacionamento de ordens
ReDim Preserve tRelacionamento(0 To UBound(tRelacionamento) - 1)

'Firmar a ordem, excluindo a ultima necessidade
bFirmarOrdem = True
GoTo FirmarOrdem

Else
'Firmar ordem integralmente
GoTo FirmarOrdem
End If

End If

'Verificar próxima encomenda interna (necessidade de produção)
GoTo ProximaEncomendaInterna

FirmarOrdem:
'Realizar ajustes de quantidade (caracteristica do processo)
nAjustesDeLote = RealizarAjustesDeLote(nQtdeAcum, i, _
    tEncomendas, tRelacionamento)
nQtdeAcum = nQtdeAcum + nAjustesDeLote

'Firmar uma ordem de produção
Call FirmarOrdemDeProducao(sItem, nQtdeAcum, nDataOrdem, _
    tRelacionamento, pDB)

IniciarNovaOrdem:
'Iniciar nova ordem de produção
bPrimeiroDaOrdem = True
i = i + 1

'Loop após firmar uma ordem de produção
Loop

'Marcar as encomendas transferidas
Set myQRY = pDB.QueryDef(cConsultaMarcarEncomendaInterna)
myQRY.Parameters(cConsultaParametroNivel) = piNivel
myQRY.Execute
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função gera as encomendas indiretas (abertura das encomendas diretas)
Private Sub GerarOrdemDeProducaoIndireta(ByVal piMomentoInicial As Double, _
    pDB As Database)

'Constantes
Const cConsultaGerarEncomendaInternaIndireta As String = "qry_GerarEncomendaInternaIndireta"
Const cConsultaParametroNivel As String = "Nivel"
Const cNomeFunc As String = "GerarOrdemDeProducaoIndireta"

'Variaveis
Dim iNivel As Integer
Dim iRegistrosAfetados As Integer
Dim myQRY As QueryDef
'----
```

```

On Error GoTo ErroDesconhecido

'Definir a consulta
Set myQRY = pIDB.QueryDef(cConsultaGerarEncomendaIndireta)

'Definir valores iniciais
iNivel = 1
iRegistrosAfetados = 1      'Garantir a entrada no loop

'Rodar a consulta para os diversos níveis
Do Until iRegistrosAfetados = 0
    myQRY.Parameters(cConsultaParametroNivel) = iNivel
    myQRY.Execute
    iRegistrosAfetados = myQRY.RecordsAffected
    'Processar os novos registros
    If iRegistrosAfetados > 0 Then
        'Quebrar encomendas maiores que capacidade da operação
        Call LimitarEncomendaIndireta(pIDB)
        'Descontar inventário
        Call DescontarInventario(pnMomentoInicial, pIDB)
        'Agrupar encomendas internas e gerar ordem de produção
        Call GerarOrdemDeProducao(iNivel, pIDB)
        'Próximo nível I.C.C.
        iNivel = iNivel + 1
    End If
Loop

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função transfere as ordens de produção para a tabela de fila
Private Sub GerarFila(ByVal pnInicioNormalizado As Double, pIDB As Database)

'Constantes da tabela
Const cConsultaTransferirOrdemParaFila As String = "qry_TransferirOrdemParaFila"
Const cConsultaParametroMomento = "Momento"
Const cNomeFunc As String = "GerarFila"

'Variáveis
Dim myQRY As QueryDef
'----

On Error GoTo ErroDesconhecido

'Transferir as ordens de produção para Fila
Set myQRY = pIDB.QueryDef(cConsultaTransferirOrdemParaFila)
myQRY.Parameters(cConsultaParametroMomento) = pnInicioNormalizado
myQRY.Execute

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

```

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Sub-funções locais

'Esta função monta um vetor com a disponibilidade diária e retorna o
tempo total disponível de uma semana

Private Function VerificarDisponibilidade(pDB As Database, _
ptInfoDisponibilidade() As tDisponibilidadeDiaria, _
ByVal pdFatorDeConversao As Double) As Double

'Constantes

Const cConsultaDisponibilidadeDiaria As String = "qry_DisponibilidadeDiaria"
Const cCampoHoraInicio As String = "Disp_HoraInicio"
Const cCampoHoraFim As String = "Disp_HoraFim"
Const cCampo24h As String = "Disp_DiaInteiro"
Const cNomeFunc As String = "VerificarDisponibilidade"

'Variáveis

Dim mvRS As Recordset
Dim mvQRY As QueryDef

On Error GoTo ErroDesconhecido

'Abrir um recordset com as disponibilidades

Set mvQRY = pDB.QueryDef(cConsultaDisponibilidadeDiaria)
Set mvRS = mvQRY.OpenRecordset(dbOpenSnapshot, dbReadOnly, dbReadOnly)

'Guardar os dados no vetor

Do Until mvRS.EOF

'Dia da semana

ptInfoDisponibilidade(mvRS.AbsolutePosition + 1).dDia = mvRS.AbsolutePosition + 1

'Hora de início das operações

ptInfoDisponibilidade(mvRS.AbsolutePosition + 1).dHoraInicio = mvRS(cCampoHoraInicio).Value

'Hora de encerramento das operações

ptInfoDisponibilidade(mvRS.AbsolutePosition + 1).dHoraFim = mvRS(cCampoHoraFim).Value

'Horas uteis disponíveis no dia (convertido de dias para horas)

If mvRS(cCampo24h) = True Then

ptInfoDisponibilidade(mvRS.AbsolutePosition + 1).nHorasUteis = pdFatorDeConversao

Else

ptInfoDisponibilidade(mvRS.AbsolutePosition + 1).nHorasUteis = _
(mvRS(cCampoHoraFim) - mvRS(cCampoHoraInicio)) * pdFatorDeConversao

End If

'Horas uteis acumuladas na semana até o dia anterior

ptInfoDisponibilidade(mvRS.AbsolutePosition + 1).nHorasUteisAcumuladas = _
ptInfoDisponibilidade(mvRS.AbsolutePosition).nHorasUteisAcumuladas + _
ptInfoDisponibilidade(mvRS.AbsolutePosition).nHorasUteis

'Próximo dia

mvRS.MoveNext

Loop

'Retornar a disponibilidade de uma semana inteira

VerificarDisponibilidade = ptInfoDisponibilidade(vbSaturday).nHorasUteis + _
ptInfoDisponibilidade(vbSaturday).nHorasUteisAcumuladas

'Fechar o recordset

mvRS.Close

'Encerrar função

Exit Function

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função normaliza um valor de data e hora

Private Function NormalizarData(ByVal pdInicio As Date, ByVal pdDataHora As Date, _
ByVal pnTempoUtilSemanal As Double, ByVal pnTempoAcumulado As Double, _
ByVal pdDataInicioDispon As Date, ByVal pdDataFimDispon As Date, _
ByVal pdFatorDeConversao As Double) As Double

'Constantes

Const cNomeFunc As String = "NormalizarData"

'----

On Error GoTo ErroDesconhecido

'Normalizar:

Horas uteis decorridas de um inicio referencial (pdInicio) _
ate um dado momento desejado (pdDataHora)

Soma: _

Semanas completas _

Dias completos na semana _

Tempo util no dia

NormalizarData = DateDiff("w", pdInicio, _
pdDataHora, vbSunday) * pnTempoUtilSemanal + _
pnTempoAcumulado + _
If (TimeValue(pdDataHora) > pdDataInicioDispon, _
If (TimeValue(pdDataHora) < pdDataFimDispon, _
TimeValue(pdDataHora) - pdDataInicioDispon, _
pdDataFimDispon - pdDataInicioDispon), 0) * pdFatorDeConversao

'Encerrar função

Exit Function

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função verifica o fator medio de conversão de tempo

Private Function VerificarFatorDeTempo(pDB As Database) As Double

'Constantes

Const cConsultaFatorDeConversaoMensal As String = "qry_FatorDeConversaoMensal"

Const cNomeFunc As String = "VerificarFatorDeTempo"

'Variáveis

Dim myQRY As QueryDef

Dim myRS As Recordset

'----

On Error GoTo ErroDesconhecido

'Calcular o fator de conversao de tempo

Set myQRY = pDB.QueryDefs(cConsultaFatorDeConversaoMensal)

Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Definir o fator

VerificarFatorDeTempo = myRS(0)

Fechar os objetos

```
myRS.Close
myQRY.Close
```

Encerrar função

Exit Function

ErroDesconhecido:

```
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

End Function

Esta função abre um recordset com as necessidades brutas para formação das ordens

```
Private Sub AbrirRecordsetNecessidadesBrutas(ByVal piNivel As Integer, _
    ptVetor() As tIncomendaInterna, _
    pDB As Database)
```

Constantes

```
Const cConsultaNecessidadesAbertas As String = "qry_NecessidadesAbertas"
```

```
Const cConsultaParametroNivel As String = "Nivel"
```

```
Const cCampoChave As String = "Encln_Chave"
```

```
Const cCampoData As String = "Encln_DataMax"
```

```
Const cCampoItem As String = "Encln_Item"
```

```
Const cCampoQtde As String = "Encln_Qtde"
```

```
Const cCampoOrdemRelat As String = "Encln_OrdemRelacionada"
```

```
Const cCampoEPP As String = "Opr_FPP"
```

```
Const cCampoCustoUnitInv As String = "Item_CustoUnitInv"
```

```
Const cCampoQtdeMin As String = "Opr_MinLote"
```

```
Const cCampoQtdeMax As String = "MaximoReal"
```

```
Const cCampoQtdeMultiplo As String = "Opr_QtdeMultiplo"
```

```
Const cNomeFunc As String = "AbrirRecordsetNecessidadesBrutas"
```

Variaveis

```
Dim myQRY As QueryDef
```

```
Dim myRS As Recordset
```

```
Dim i As Integer
```

```
----
```

On Error GoTo ErroDesconhecido

Abrir o recordset

```
Set myQRY = pDB.QueryDef(cConsultaNecessidadesAbertas)
```

```
myQRY.Parameters(cConsultaParametroNivel) = piNivel
```

```
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)
```

Transferir os valores para um vetor

```
i = 1
```

```
Do Until myRS.EOF
```

```
ReDim Preserve ptVetor(1 To i)
```

```
ptVetor(i).iChave = myRS(cCampoChave)
```

```
ptVetor(i).nDataMax = myRS(cCampoData)
```

```
ptVetor(i).sItem = myRS(cCampoItem)
```

```
ptVetor(i).nQuantidade = myRS(cCampoQtde)
```

```
ptVetor(i).iOrdemRelacionada = myRS(cCampoOrdemRelat)
```

```
ptVetor(i).nEPP = myRS(cCampoEPP)
```

```
ptVetor(i).nCustoUnitInv = myRS(cCampoCustoUnitInv)
```

```
ptVetor(i).nQtdeMin = myRS(cCampoQtdeMin)
```

```
ptVetor(i).nQtdeMax = myRS(cCampoQtdeMax)
```

```
ptVetor(i).nQtdeMultiplo = myRS(cCampoQtdeMultiplo)
```

```
i = i + 1
```

```
myRS.MoveNext
```

```
Loop
```

Fechar os objetos

```

myRS.Close
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função realiza ajustes nos tamanhos das ordens de produção
Private Function RealizarAjustesDeLote(ByVal pnQtde As Single, _
    ByVal piIndice As Integer, _
    ptVetor() As tRecomendaçãoInterna, _
    ptRelacionamento() As tRelacionamentoDeOrdens) As Single

'Constantes
Const cNomeFunc As String = "RealizarAjustesDeLote"

'Variáveis
Dim nQuantidadeMultipla As Single
Dim nSobra As Single
'---

On Error GoTo ErroDesconhecido

'O ajuste de maximo e evitado durante o processo de formação da ordem!!!

'Ajuste de minimo
'Considerar que não há mais necessidades para ser acrescentada _
'    (se houvesse, já teria sido acrescentada)
If pnQtde < ptVetor(piIndice).nQtdeMin Then
    'Ajuste para a ordem de produção
    RealizarAjustesDeLote = ptVetor(piIndice).nQtdeMin - pnQtde
    'Correção do destino do lote
    ReDim Preserve ptRelacionamento(0 To 1)
    ptRelacionamento(1).iReferencia = 0
    ptRelacionamento(1).nQuantidade = ptVetor(piIndice).nQtdeMin - pnQtde
End If

'Ajuste de múltiplo
If ptVetor(piIndice).nQtdeMultiplo > 0 Then
    'Número de múltiplos necessario
    nQuantidadeMultipla = Int((CInt(pnQtde / ptVetor(piIndice).nQtdeMultiplo) > _
        pnQtde / ptVetor(piIndice).nQtdeMultiplo, _
        CInt(pnQtde / ptVetor(piIndice).nQtdeMultiplo), _
        CInt(pnQtde / ptVetor(piIndice).nQtdeMultiplo) + 1) * _
        ptVetor(piIndice).nQtdeMultiplo

    'Ajuste para a ordem de produção
    RealizarAjustesDeLote = nQuantidadeMultipla - pnQtde

    'Correção do destino do lote
    'Se houver mais necessidade para alocar
    If piIndice = 1/Bound(ptVetor) Then GoTo NaoAlocarOrdem
    If ptVetor(piIndice - 1).sltem = ptVetor(piIndice).sltem Then
        nSobra = nQuantidadeMultipla - pnQtde
        'Alocar gradualmente as necessidades posteriores
        piIndice = piIndice + 1
        Do Until ptVetor(piIndice).sltem <> ptVetor(piIndice - 1).sltem _
            Or nSobra = 0
            'Quantidade que falta ser alocada
            If nSobra >= ptVetor(piIndice).nQuantidade Then

```

```

'Alocação
ReDim Preserve ptRelacionamento(0 To UBound(ptRelacionamento) + 1)
ptRelacionamento(UBound(ptRelacionamento)).iReferencia = _
    ptVetor(piIndice).iOrdemRelacionada
ptRelacionamento(UBound(ptRelacionamento)).nQuantidade = _
    ptVetor(piIndice).nQuantidade
'Descontar necessidade
ptVetor(piIndice).nQuantidade = 0
'Valor que sobra para ser alocado
nSobra = nSobra - ptVetor(piIndice).nQuantidade
Else
'Alocação
ReDim Preserve ptRelacionamento(0 To UBound(ptRelacionamento) + 1)
ptRelacionamento(UBound(ptRelacionamento)).iReferencia = _
    ptVetor(piIndice).iOrdemRelacionada
ptRelacionamento(UBound(ptRelacionamento)).nQuantidade = nSobra
'Descontar necessidade
ptVetor(piIndice).nQuantidade = ptVetor(piIndice).nQuantidade - nSobra
'Valor que sobra para ser alocado
nSobra = 0
End If

'Próxima necessidade
piIndice = piIndice + 1
If piIndice > UBound(ptVetor) Then Exit Do

Loop

NaoAlocarOrdem:
'Se NÃO houver mais necessidade para alocar
Else
    ReDim Preserve ptRelacionamento(0 To UBound(ptRelacionamento) + 1)
    ptRelacionamento(UBound(ptRelacionamento)).iReferencia = 0
    ptRelacionamento(UBound(ptRelacionamento)).nQuantidade = _
        nQuantidadeMultipla - pnQtde
End If
End If

'Encerrar função
Exit Function

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função gera uma ordem de produção específica
Private Sub FirmarOrdemDeProducao(ByVal psItem As String, ByVal pnQtde As Double, _
    ByVal pnDataConclusao As Double, _
    ptRelacionamento() As tRelacionamentoDeOrdens, _
    pDB As Database)

'Constantes
Const cConsultaInserirOrdemDeProducao As String = "qry_InserirOrdemDeProducao"
Const cConsultaParametroItem As String = "Item"
Const cConsultaParametroQtde As String = "Quantidade"
Const cConsultaParametroConclusao As String = "Conclusao"

Const cConsultaInserirRelacionamentoDeOrdem As String = "qry_InserirRelacionamentoDeOrdem"
Const cConsultaParametroCriada As String = "Ordem criada"
Const cConsultaParametroReferencia As String = "Ordem de referencia"

Const cNomeFunc As String = "FirmarOrdemDeProducao"

```

```

'Variaveis
Dim myQRY As QueryDef
Dim iCodigoOrdem As Long
Dim i As Integer
'-----

On Error GoTo ErroDesconhecido

'Inserir registro na tabela de Ordens de Produção
Set myQRY = pIDB.QueryDefs(cConsultaInserirOrdemDeProducao)
myQRY.Parameters(cConsultaParametroItem) = psItem
myQRY.Parameters(cConsultaParametroQtde) = pnQtde
myQRY.Parameters(cConsultaParametroConclusao) = pnDataConclusao
myQRY.Execute

'Verificar o código da ordem inserida
iCodigoOrdem = VerificarCodigoUltimaOrdem(pIDB)

'Atualizar a tabela de relacionamento de ordens
Set myQRY = pIDB.QueryDefs(cConsultaInserirRelacionamentoDeOrdem)
myQRY.Parameters(cConsultaParametroCriada) = iCodigoOrdem
'Inserir os registros de relacionamento de ordens
For i = LBound(ptRelacionamento) To UBound(ptRelacionamento)
    myQRY.Parameters(cConsultaParametroReferencia) = ptRelacionamento(i).iReferencia
    myQRY.Parameters(cConsultaParametroQtde) = ptRelacionamento(i).nQuantidade
    myQRY.Execute
Next

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica o código da última ordem de produção inserida
Private Function VerificarCodigoUltimaOrdem(pIDB As Database) As Long

'Constantes
Const cConsultaCodigoUltimaOrdem As String = "qry_CodigoUltimaOrdem"
Const cNomeFunc As String = "VerificarCodigoUltimaOrdem"

'Variáveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'-----

On Error GoTo ErroDesconhecido

'Fazer a verificação
Set myQRY = pIDB.QueryDefs(cConsultaCodigoUltimaOrdem)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Código da próxima ordem
VerificarCodigoUltimaOrdem = myRS(0)

'Fechar os objetos
myRS.Close
myQRY.Close

```

'Encerrar função

Exit Function

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função quebra necessidades muito grandes (maiores que capacidade da operação)

Private Sub LimitarEncomendaInterna(pDB As Database)

'Constantes

Const cConsultaLimitarEncomendaInterna1 = "qry_LimitarEncomendaInterna_1"

Const cConsultaLimitarEncomendaInterna2 As String = "qry_LimitarEncomendaInterna_2"

Const cConsultaLimitarEncomendaInterna3 As String = "qry_LimitarEncomendaInterna_3"

Const cCampoChave = "EncIn_Chave"

Const cCampoLotesInteiros = "LotesInteiros"

Const cConsultaParametroChave As String = "Chave"

Const cNomeFunc As String = "LimitarEncomendaInterna"

'Variáveis

Dim mvQRY As QueryDef

Dim mvRS As Recordset

Dim i As Integer

'----

On Error GoTo ErroDesconhecido

'Abrir recordset de encomendas excedentes

Set mvQRY = pDB.QueryDef(cConsultaLimitarEncomendaInterna1)

Set mvRS = mvQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Sair da função se não houver encomenda superior ao limite

If mvRS.EOF Then GoTo EncerrarVerificacao

'Dividir encomendas

'Inserir nova(s) encomenda(s) com o limite máximo real (considerar múltiplos)

Do Until mvRS.EOF

For i = 1 To mvRS(cCampoLotesInteiros)

Set mvQRY = pDB.QueryDef(cConsultaLimitarEncomendaInterna2)

mvQRY.Parameters(cConsultaParametroChave) = mvRS(cCampoChave)

mvQRY.Execute

Next

mvRS.MoveNext

Loop

'Alterar quantidades atuais, deixando apenas o "resto"

Set mvQRY = pDB.QueryDef(cConsultaLimitarEncomendaInterna3)

mvQRY.Execute

EncerrarVerificacao:

'Fechar os objetos

mvQRY.Close

mvRS.Close

'Encerrar função

Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função desconta das encomendas internas o inventário inicial

```
Private Sub DescontarInventario(ByVal pnMomentoInicial As Double, pDB As Database)
```

```
    'Constantes
```

```
    Const cConsultaInventarioInicial As String = "qry_InventarioInicial"
```

```
    Const cCampoItem As String = "Encln_Item"
```

```
    Const cCampoNecessidade As String = "Encln_Qtde"
```

```
    Const cCampoDisponibilidade As String = "Inventario Inicial"
```

```
    Const cCampoOrdem As String = "Encln_OrdemRelacionada"
```

```
    Const cCampoChave As String = "Encln_Chave"
```

```
    Const cNomeFunc As String = "DescontarInventario"
```

```
    'Variaveis
```

```
    Dim mvQRY As QueryDef
```

```
    Dim myRS As Recordset
```

```
    Dim sltem As String
```

```
    Dim bProximoItem As Boolean
```

```
    Dim nQtdeAlocada As Double
```

```
    '-----
```

```
    On Error GoTo ErroDesconhecido
```

```
    'Abrir um recordset do inventario inicial
```

```
    Set myQRY = pDB.QueryDef(cConsultaInventarioInicial)
```

```
    Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)
```

```
    'Fechar a consulta
```

```
    myQRY.Close
```

```
    'Verificar ação para cada linha
```

```
    Do Until myRS.EOF
```

```
        'Se for novo item, zerar a quantidade alocada
```

```
        If sltem <> myRS(cCampoItem) Then nQtdeAlocada = 0
```

```
        'Definir o item
```

```
        sltem = myRS(cCampoItem)
```

```
        'Verificar relação entre necessidade e disponibilidade
```

```
        Select Case myRS(cCampoDisponibilidade) - nQtdeAlocada - myRS(cCampoNecessidade)
```

```
            'Se necessidade > disponibilidade, descontar e alocar inventário
```

```
            Case Is < 0
```

```
                'Alocar inventário inicial na tabela de inventário dinâmico
```

```
                Call AlocarInventarioInicial(sltem, myRS(cCampoDisponibilidade) - nQtdeAlocada, _  
                    myRS(cCampoOrdem), pnMomentoInicial, pDB)
```

```
                'Descontar necessidade da encomenda interna
```

```
                Call DescontarNecessidadeEncomendaInterna(myRS(cCampoChave), _  
                    myRS(cCampoDisponibilidade) - nQtdeAlocada, False, pDB)
```

```
                'Andar ate o proximo item
```

```
                bProximoItem = True
```

```
            'Se necessidade = disponibilidade, excluir necessidade e alocar inventario
```

```
            Case 0
```

```
                'Alocar inventário inicial na tabela de inventário dinâmico
```

```
                Call AlocarInventarioInicial(sltem, myRS(cCampoDisponibilidade), _  
                    myRS(cCampoOrdem), pnMomentoInicial, pDB)
```

```
                'Descontar necessidade da encomenda interna
```

```
                Call DescontarNecessidadeEncomendaInterna(myRS(cCampoChave), _  
                    myRS(cCampoDisponibilidade), True, pDB)
```

```
                'Andar ate o proximo item
```

```
                bProximoItem = True
```

```
            'Se necessidade < disponibilidade, excluir e alocar inventario
```

```
            Case Is > 0
```

```
                'Alocar inventário inicial na tabela de inventário dinâmico
```

```
                Call AlocarInventarioInicial(sltem, myRS(cCampoNecessidade), _  
                    myRS(cCampoOrdem), pnMomentoInicial, pDB)
```

```
                'Descontar necessidade da encomenda interna
```

```
                Call DescontarNecessidadeEncomendaInterna(myRS(cCampoChave), _  
                    myRS(cCampoDisponibilidade), True, pDB)
```

```
                'Registrar a quantidade alocada
```

```

        nQtdeAlocada = nQtdeAlocada + myRS(cCampoNecessidade)
        'Andar ate o próximo item
        bProximoItem = False
    End Select

    'Próximo registro
    myRS.MoveNext

    'Verificar se precisa passar para proximo item (inventário esgotado)
    If bProximoItem = True And myRS.EOF = False Then
        Do Until myRS(cCampoItem) <> sItem
            If Not myRS.EOF Then
                myRS.MoveNext
            End If
            If myRS.EOF Then Exit Do
        Loop
    End If

Loop

'Fechar objetos
myRS.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função aloca o inventário inicial no inventário dinâmico
Private Sub AlocarInventarioInicial(ByVal psItem As String, ByVal pnQtde As Double, _
    ByVal piOrdem As Long, ByVal pnMomento As Double, _
    pDB As Database)

    'Constantes
    Const cConsultaInserirInventarioInicial As String = "qry_InserirInventarioInicial"
    Const cConsultaParametroItem As String = "Item"
    Const cConsultaParametroQuantidade As String = "Quantidade"
    Const cConsultaParametroOrdem As String = "Ordem Alocada"
    Const cConsultaParametroMomento As String = "Momento"
    Const cNomeFunc As String = "AlocarInventarioInicial"

    'Variaveis
    Dim myQRY As QueryDef
    '-----

    On Error GoTo ErroDesconhecido

    'Executar a consulta de inserção
    Set myQRY = pDB.QueryDefs(cConsultaInserirInventarioInicial)
    myQRY.Parameters(cConsultaParametroItem) = psItem
    myQRY.Parameters(cConsultaParametroQuantidade) = pnQtde
    myQRY.Parameters(cConsultaParametroOrdem) = piOrdem
    myQRY.Parameters(cConsultaParametroMomento) = pnMomento
    myQRY.Execute

    'Fechar os objetos
    myQRY.Close

    'Encerrar função
    Exit Sub

```

```

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função desconta das encomendas internas o inventario inicial
Private Sub DescontarNecessidadeEncomendaInterna ByVal piEncomenda As Long, _
    ByVal pnQtde As Double, pbExcluir As Boolean, pDB As Database)

'Constantes
Const cConsultaDescontarInventarioInicial As String = "qry_DescontarInventarioInicial"
Const cConsultaExcluirEncomenda As String = "qry_ExcluirEncomendaPorInventario"
Const cConsultaParametroDesconto As String = "Desconto"
Const cConsultaParametroEncomenda As String = "Encomenda"
Const cNomeFunc As String = "DescontarNecessidadeEncomendaInterna"

'Variaveis
Dim myQRY As QueryDef
'-----

On Error GoTo ErroDesconhecido

'Verificar se desconta necessidade ou exclui necessidade
If pbExcluir = True Then
    Set myQRY = pDB.QueryDef(cConsultaExcluirEncomenda)
    myQRY.Parameters(cConsultaParametroEncomenda) = piEncomenda
Else
    Set myQRY = pDB.QueryDef(cConsultaDescontarInventarioInicial)
    myQRY.Parameters(cConsultaParametroEncomenda) = piEncomenda
    myQRY.Parameters(cConsultaParametroDesconto) = pnQtde
End If

'Executar atualização
myQRY.Execute

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

```

Fase 2:

```

Option Explicit
Const mcModulo As String = "Fase 2"

'Esta função faz a seqüenciação das ordens de produção
Sub SequenciarOrdensDeProducao ByVal pnMomentoInicial As Double, _
    ByVal pDataHoraInicioSimulacao As Date, _
    pDB As Database, pForm As Form)

'Constantes de nome de campo dos recordset
Const cCampoCodRT = "Fila_CodRT"

Const cCampoRegraSeq_1 As String = "CampoDeSequenciacao_1"
Const cCampoOrdemSeq_1 As String = "OrdemDeSequenciacao_1"
Const cCampoRegraSeq_2 As String = "CampoDeSequenciacao_2"
Const cCampoOrdemSeq_2 As String = "OrdemDeSequenciacao_2"

```

```

Const cCampoDisponibilidade As String = "RecursoDisponivel"
Const cCampoOrdemDeProducao As String = "Fila_CodOrdem"
Const cCampoUnidadesRT As String = "RT_Qtde"

Const cRotuloSubStatus As String = "IblSubStatus"
Const cNomeFunc As String = "SequenciarOrdensDeProducao"

'Variáveis
Dim rsRecursoTrabalho As Recordset
Dim rsFila As Recordset
Dim i As Integer
Dim nContaTempo As Double
Dim bContinuarSimulacao As Boolean
Dim iOrdensNaoAgendadas As Integer
Dim iTotalDeOrdens As Integer
'-----

On Error GoTo ErroDesconhecido

'Calcular propriedades dinâmicas iniciais
Call AtualizarPropriedadesDinamicas(pnMomentoInicial, pDB)

'Contar total de ordens para serem programadas
iTotalDeOrdens = ContarOrdensNaoRealizadas(pDB)

'Iniciar sequenciação
'Momento de inicio da simulação
nContaTempo = pnMomentoInicial
'Garantir entrada no loop
bContinuarSimulacao = True

'Loop ate que todas as ordens sejam programadas
Do While bContinuarSimulacao = True

'Abrir um recordset com os recursos de trabalho onde há fila
Call DefinirRecursosDeTrabalho(nContaTempo, rsRecursoTrabalho, pDB)

'Iniciar a sequenciação
Do Until rsRecursoTrabalho.EOF
'Abrir o recordset ordenado da fila para determinado recurso de trabalho
Call AbrirRecordsetFila(rsRecursoTrabalho(cCampoCodRT), _
rsRecursoTrabalho(cCampoRegraSeq_1), _
rsRecursoTrabalho(cCampoOrdemSeq_1), _
rsRecursoTrabalho(cCampoRegraSeq_2), _
rsRecursoTrabalho(cCampoOrdemSeq_2), _
rsFila, pDB)

'Realizar operações (na quantidade de recursos disponiveis)
For i = 1 To rsRecursoTrabalho(cCampoDisponibilidade)
If rsFila.EOF = False Then
'Realizar operação
Call RealizarOperacao(rsFila(cCampoOrdemDeProducao), nContaTempo, _
rsRecursoTrabalho(cCampoUnidadesRT), pDB)
'Próxima operação do recurso de trabalho
rsFila.MoveNext
End If
Next
'Proximo recurso de trabalho disponivel
rsRecursoTrabalho.MoveNext
Loop

'Verificar se todas as operações já foram realizadas
iOrdensNaoAgendadas = ContarOrdensNaoRealizadas(pDB)

```

```

bContinuarSimulacao = CBool(iOrdensNaoAgendadas)

'Atualizar mensagem de status
pForm.Controls(cRotuloSubStatus).Caption = "Ordens programadas: " & _
    iTotalDeOrdens - iOrdensNaoAgendadas & " / " & iTotalDeOrdens
pForm.Repaint

'Calcular proximo momento
nContaTempo = AndarContadorDeTempo(nContaTempo, pIDB)
Loop

'Fechar os objetos
rsRecursoTrabalho.Close
rsFila.Close

'Desnormalizar dados do programa de produção
Call GerarProgramaDataHora(pIDataHoraInicioSimulacao, pIDB)

'Garantir que itens de origem externa estão cadastrados
Call CadastrarLeadTimeMP(pIDB)

'Encerrar função
Exit Sub

'Erro desconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

Funções locais
=====

'Esta função calcula as propriedades estaticas dos elementos da fila
Private Sub AtualizarPropriedadesDinamicas(ByVal pnMomento As Double, _
    pIDB As Database)

'Constantes
Const cNomeFunc As String = "AtualizarPropriedadesDinamicas"
'-----

On Error GoTo ErroDesconhecido

'Atualizar tempo de set up
Call AtualizarTempoSetUp(pnMomento, pIDB)

'Atualizar disponibilidade de materiais
Call AtualizarDisponibilidadeMateriais(pnMomento, pIDB)

'Atualizar demais propriedades dinâmicas
Call AtualizarTemposDinamicos(pnMomento, pIDB)

'Encerrar função
Exit Sub

'Erro desconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função conta o total de ordens que devem ser programadas
Private Function ContarTotalDeOrdens(pIDB As Database) As Integer

'Constantes

```

```

Const cConsultaContarTotalDeOrdens = "qry_ContarTotalDeOrdens"
Const cNomeFunc As String = "ContarTotalDeOrdens"

'Variáveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'----

On Error GoTo ErroDesconhecido

'Executar consulta
Set myQRY = pDB.QueryDefs(cConsultaContarTotalDeOrdens)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)
ContarTotalDeOrdens = myRS(0)

'Fechar objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Function

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função monta um recordset com todos os recursos de trabalho
'    disponíveis que têm fila
Private Sub DefinirRecursosDeTrabalho ByVal pnMomento As Double, _
    pRS As Recordset, pDB As Database)

'Constantes
Const cConsultaRecursoDisponivel As String = "qry_RecursoDisponivel"
Const cConsultaParametro As String = "Momento"
Const cNomeFunc As String = "DefinirRecursosDeTrabalho"

'Variáveis
Dim myQRY As QueryDef
'----

On Error GoTo ErroDesconhecido

'Definir o recordset
Set myQRY = pDB.QueryDefs(cConsultaRecursoDisponivel)
myQRY.Parameters(cConsultaParametro) = pnMomento
Set pRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função abre um recordset indicando a ordem de preferência para cada
'    elemento da fila de determinado Recurso de Trabalho
Private Sub AbrirRecordsetFila ByVal psRecurso As String, _
    ByVal psCampoCritério_1 As String, ByVal psAscDesc_1 As String, _
    ByVal psCampoCritério_2 As String, ByVal psAscDesc_2 As String, _

```

```

    pRS As Recordset, pDB As Database)

'Constantes da tabela de Fila para montagem da instrução SQL
Const cTabFila As String = "Fila"
Const cFilaCampoCodRT = "Fila_CodRT"
Const cFilaCampoOrdem As String = "Fila_CodOrdem"
Const cFilaCampoVerificacao As String = "Fila_InicioOperacao"
Const cFilaCampoDisponibilidade As String = "Fila_DisponMat"
Const cNomeFunc As String = "AbrirRecordsetFila"

'Variáveis
Dim sSQL As String
'-----

On Error GoTo ErroDesconhecido

'Montar e executar a instrução de SQL
sSQL = "SELECT " & cFilaCampoOrdem & ", " & _
    psCampoCritério_1 & ", " & psCampoCritério_2 & _
    " FROM " & cTabFila & _
    " WHERE " & cFilaCampoCodRT & " = " & psRecurso & _
    " AND " & cFilaCampoDisponibilidade & " = True" & _
    " AND " & cFilaCampoVerificacao & " = Null" & _
    " ORDER BY " & psCampoCritério_1 & " " & psAscDesc_1 & _
    " " & psCampoCritério_2 & " " & psAscDesc_2

    Set pRS = pDB.OpenRecordset(sSQL, dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função realiza uma operação, dado um recurso de trabalho
Private Sub RealizarOperacao(BvVal piCodOrdem As Long, BvVal pnMomento As Double, _
    BvVal piRTIndiceMaximo As Integer, pDB As Database)

'Constantes
Const cNomeFunc As String = "RealizarOperacao"

'Variáveis
Dim iIndiceRT As Integer
'-----

On Error GoTo ErroDesconhecido

'Definir índice do recurso de trabalho utilizado
iIndiceRT = DefinirIndiceRT(piCodOrdem, pnMomento, piRTIndiceMaximo, pDB)

'Corrigir o tempo de set-up (de acordo com o índice do recurso utilizado)
Call VerificarTempoSetUp(piCodOrdem, iIndiceRT, pnMomento, pDB)

'Definir início e término da ordem de produção
Call DefinirOrdemRealizada(piCodOrdem, iIndiceRT, pnMomento, pDB)

'Inserir no programa de produção
Call InserirPrograma(piCodOrdem, pDB)

'Atualizar inventário
Call AtualizarInventario(piCodOrdem, pnMomento, pDB)

```

```

'Atualizar propriedades dinâmicas
    Call AtualizarPropriedadesDinamicas(pnMomento, pIDB)

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função insere os dados na tabela Programa_DataHora,
' gerando datas e horário reais (desnormalizados) para as ordens
Private Sub GerarProgramaDataHora(ByVal pnMomentoInicial As Date, pIDB As Database)

'Constantes
Const cConsultaProgramaDataHora As String = "qry_GerarProgramaDataHora"
Const cConsultaParametroDataInicio As String = "InicioSemana"
Const cNomeFunc As String = "GerarProgramaDataHora"

'Variáveis
Dim mvQRY As QueryDef
Dim dInicioPrimeiraSemana As Date
'---

On Error GoTo ErroDesconhecido

'Definir o momento inicial da primeira semana (referência de tempo)
dInicioPrimeiraSemana = DateValue(pnMomentoInicial - _
    WeekDay(pnMomentoInicial, vbSunday) + 1)

'Definir a consulta de inserção de dados
Set mvQRY = pIDB.QueryDefs(cConsultaProgramaDataHora)
mvQRY.Parameters(cConsultaParametroDataInicio) = dInicioPrimeiraSemana
mvQRY.Execute

'Fechar os objetos
mvQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
    Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica itens de origem externa que não têm lead-time cadastrado
Private Sub CadastrarLeadTimeMP(pIDB As Database)

'Constantes
Const cConsultaInserirMP_Prazos As String = "qry_InserirMP_Prazos"
Const cNomeFunc As String = "CadastrarLeadTimeMP"

'Variáveis
Dim mvQRY As QueryDef
'---

On Error GoTo ErroDesconhecido

'Executar a consulta
Set mvQRY = pIDB.QueryDefs(cConsultaInserirMP_Prazos)
mvQRY.Execute

```

'Fechar os objetos

myQRY.Close

'Encerrar função

Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Sub-funções locais

=====

'Esta função atualiza o tempo de set up da tabela Fila

Private Sub AtualizarTempoSetUp(ByVal pnMomento As Double, pDB As Database)

'Constantes

Const cConsultaGerarTabelaProvisoria As String = "qry_DefinirSetUpDinamico_3"

Const cConsultaParametroMomento As String = "Momento"

Const cConsultaAtualizarTempoSetUp As String = "qry_DefinirSetUpDinamico_4"

Const cConsultaApagarProvTempoSetUp As String = "qry_ApagarProv_TempoSetUp"

Const cNomeFunc As String = "AtualizarTempoSetUp"

'Variaveis

Dim mvQRY As QueryDef

Dim sSQL As String

On Error GoTo ErroDesconhecido

'Executar a consulta que gera uma tabela provisória

Set myQRY = pDB.QueryDefs(cConsultaGerarTabelaProvisoria)

myQRY.Parameters(cConsultaParametroMomento) = pnMomento

myQRY.Execute

'Executar a consulta que atualiza a tabela de fila

Set mvQRY = pDB.QueryDefs(cConsultaAtualizarTempoSetUp)

mvQRY.Execute

'Apagar a tabela provisoria

Set mvQRY = pDB.QueryDefs(cConsultaApagarProvTempoSetUp)

mvQRY.Execute

'Fechar os objetos

myQRY.Close

'Encerrar função

Exit Sub

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função atualiza a disponibilidade de materiais para os elementos das filas

Private Sub AtualizarDisponibilidadeMateriais(ByVal pnMomento As Double, _
pDB As Database)

'Constantes

Const cConsultaAtualizarDisponibilidade = "qry_AtualizarDisponibilidadeDeMateriais"

Const cConsultaParametroMomento = "Momento"

```

Const cNomeFunc As String = "AtualizarDisponibilidadeMateriais"

'Variáveis
Dim myQRY As QueryDef
'-----

On Error GoTo ErroDesconhecido

'Executar a consulta de atualização
Set myQRY = pDB.QueryDef(cConsultaAtualizarDisponibilidade)
myQRY.Parameters(cConsultaParametroMomento) = pnMomento
myQRY.Execute

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função calcula o tempo de processo que falta
Private Sub AtualizarTemposDinamicos(ByVal pnMomento As Double, pDB As Database)

'Constantes das consultas executadas
Const cConsultaPropriedadesDinamicas As String = "qry_AtualizarPropriedadesDinamicas"
Const cConsultaParametroMomento As String = "Momento"
Const cNomeFunc As String = "AtualizarTemposDinamicos"

'Variáveis
Dim myQRY As QueryDef
Dim sSQL As String
'-----

On Error GoTo ErroDesconhecido

'Atualizar tabela de Fila
Set myQRY = pDB.QueryDef(cConsultaPropriedadesDinamicas)
myQRY.Parameters(cConsultaParametroMomento) = pnMomento
myQRY.Execute

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função define o índice do recurso de trabalho a ser utilizado
Private Function DefinirIndiceRT(ByVal piCodOrdem As Long, ByVal pnMomento As Single, _
ByVal piIndiceMaximo As Integer, pDB As Database) As Integer

'Constantes
Const cConsultaVerificarIndiceRT_Usados As String = "qry_VerificarIndiceRT_Usados"
Const cConsultaVerificarIndiceRT_EmUso As String = "qry_VerificarIndiceRT_EmUso"
Const cConsultaParametroOrdemProd As String = "Ordem de Producao"
Const cConsultaParametroMomento As String = "Momento"

```

```

Const cCampoIndice As String = "Prog_IndiceRT"
Const cNomeFunc As String = "DefinirIndiceRT"

'Variáveis
Dim i As Integer
Dim mvQRY As QueryDef
Dim mvRS As Recordset
'-----

On Error GoTo ErroDesconhecido

'Teste inicial
'Se só houver uma unidade do recurso, o índice é 1
If piIndiceMaximo = 1 Then
    DefinirIndiceRT = 1
    Exit Function
End If

'Abrir a consulta
Set mvQRY = pIDB.QueryDefs(cConsultaVerificarIndiceRT_Usados)
mvQRY.Parameters(cConsultaParametroOrdemProd) = piCodOrdem
mvQRY.Parameters(cConsultaParametroMomento) = pnMomento
Set mvRS = mvQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Se retornar algum registro, utilizar o primeiro índice retornado
If Not mvRS.EOF Then
    DefinirIndiceRT = mvRS(cCampoIndice)
    GoTo FecharObjetos

'Se não retornar registros, índice que ainda não foi utilizado
Else
    'Consulta de índices em uso
    Set mvQRY = pIDB.QueryDefs(cConsultaVerificarIndiceRT_EmUso)
    mvQRY.Parameters(cConsultaParametroOrdemProd) = piCodOrdem
    mvQRY.Parameters(cConsultaParametroMomento) = pnMomento
    Set mvRS = mvQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

    'Procura o primeiro índice disponível
    DefinirIndiceRT = 1
    Do Until mvRS.EOF
        If mvRS(cCampoIndice) > DefinirIndiceRT Then GoTo FecharObjetos
        DefinirIndiceRT = DefinirIndiceRT + 1
        mvRS.MoveNext
    Loop

End If

FecharObjetos:

'Fecha os objetos
mvRS.Close
mvQRY.Close

'Encerrar função
Exit Function

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função verifica o tempo de set-up correto e, se necessário, corrige a fila
Private Sub VerificarTempoSetUp(ByVal piCodOrdem As Long, piIndiceRT As Integer, _
    ByVal pnMomento As Double, pIDB As Database)

```

'Constantes

```
Const cConsultaVerificarSetUpReal As String = "qry_VerificarSetUpReal"
Const cConsultaCorrigirSetUp As String = "qry_CorrigirTempoSetUp"
Const cConsultaParametroOrdem As String = "Ordem de Producao"
Const cConsultaParametroIndice As String = "Indice do Recurso"
Const cConsultaParametroMomento As String = "Momento"
Const cConsultaParametroTempoCorrigido As String = "Tempo Corrigido"
Const cConsultaCampoSetUp As String = "SetUpReal"
Const cFilaCampoOrdem As String = "Fila_CodOrdem"
Const cNomeFunc As String = "VerificarTempoSetUp"
```

'Variaveis

```
Dim mvQRY As QueryDef
Dim mvRS As Recordset
Dim nTempoSetUp As Double
'----
```

On Error Go To ErroDesconhecido

'Verificar o tempo real de set-up

```
Set mvQRY = pIDB.QueryDef(cConsultaVerificarSetUpReal)
mvQRY.Parameters(cConsultaParametroOrdem) = piCodOrdem
mvQRY.Parameters(cConsultaParametroIndice) = piIndiceRT
mvQRY.Parameters(cConsultaParametroMomento) = pnMomento
Set mvRS = mvQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)
```

'Definir a consulta de atualização

```
Set mvQRY = pIDB.QueryDef(cConsultaCorrigirSetUp)
mvQRY.Parameters(cConsultaParametroOrdem) = piCodOrdem
mvQRY.Parameters(cConsultaParametroTempoCorrigido) = mvRS(cConsultaCampoSetUp)
```

'Fechar o recordset

```
mvRS.Close
```

'Executar a atualização

```
mvQRY.Execute
```

'Fechar a consulta

```
mvQRY.Close
```

'Encerrar função

```
Exit Sub
```

ErroDesconhecido:

```
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

```
End Sub
```

'Esta função define como realizada uma ordem de produção na fila

```
Private Sub DefinirOrdemRealizada(BvVal piCodOrdem As Long, _
    BvVal piIndiceRT As Integer, _
    BvVal pnMomento As Double, pDB As Database)
```

'Constantes das consultas que definem a ordem de produção como realizada

```
Const cConsultaRealizarOperacao As String = "qry_RealizarOperacao"
Const cConsultaParametroMomento As String = "Momento"
Const cConsultaParametroOrdemDeProducao As String = "Ordem de Producao"
Const cConsultaParametroIndiceRT As String = "Indice do RT"
Const cNomeFunc As String = "DefinirOrdemRealizada"
```

'Variaveis

```
Dim mvQRY As QueryDef
'----
```

On Error GoTo ErroDesconhecido

Definir a consulta

```
Set myQRY = pDB.QueryDefs(cConsultaRealizarOperacao)
myQRY.Parameters(cConsultaParametroMomento) = pnMomento
myQRY.Parameters(cConsultaParametroOrdemDeProducao) = piCodOrdem
myQRY.Parameters(cConsultaParametroIndiceRT) = piIndiceRT
```

Executar a atualização

```
myQRY.Execute
```

Fechar os objetos

```
myQRY.Close
```

Encerrar função

```
Exit Sub
```

ErroDesconhecido:

```
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

```
End Sub
```

Esta função insere uma ordem de produção na tabela de Programação

```
Private Sub InserirPrograma(ByVal piCodOrdem As Long, pDB As Database)
```

Constantes da consulta de inserção

```
Const cConsultaInserirPrograma As String = "qry_InserirPrograma"
Const cConsultaParametroOrdemDeProducao As String = "Ordem de Producao"
Const cNomeFunc As String = "InserirPrograma"
```

Variáveis

```
Dim myQRY As QueryDef
```

```
----
```

On Error GoTo ErroDesconhecido

Definir a consulta

```
Set myQRY = pDB.QueryDefs(cConsultaInserirPrograma)
myQRY.Parameters(cConsultaParametroOrdemDeProducao) = piCodOrdem
```

Executar a inserção

```
myQRY.Execute
```

Fechar os objetos

```
myQRY.Close
```

Encerrar função

```
Exit Sub
```

ErroDesconhecido:

```
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

```
End Sub
```

Esta função atualiza o inventário ao se realizar uma operação

```
Private Sub AtualizarInventario(ByVal piCodOrdem As Long, _
    ByVal pnMomento As Single, pDB As Database)
```

Constantes

```
Const cConsultaInserirProducaoInventario As String = "qry_InserirProducaoInventario"
Const cConsultaConsumirInventario As String = "qry_ConsumirInventario"
Const cConsultaParametroOrdemDeProducao As String = "Ordem de Producao"
Const cConsultaParametroMomento As String = "Momento"
```

```

Const cNomeFunc As String = "AtualizarInventario"

'Variáveis
Dim myQRY As QueryDef
'----

On Error GoTo ErroDesconhecido

'Consumir o inventario alocado para a ordem
Set myQRY = pDB.QueryDefs(cConsultaConsumirInventario)
myQRY.Parameters(cConsultaParametroOrdemDeProducao) = piCodOrdem
myQRY.Parameters(cConsultaParametroMomento) = pnMomento
myQRY.Execute

'Inserir a produção na tabela de Inventario Dinâmico
Set myQRY = pDB.QueryDefs(cConsultaInserirProducaoInventario)
myQRY.Parameters(cConsultaParametroOrdemDeProducao) = piCodOrdem
myQRY.Execute

'Fechar os objetos
myQRY.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função verifica se há ordens de produção não agendadas
Private Function ContarOrdensNaoRealizadas(pDB As Database) As Integer

'Constantes
Const cConsultaContarOrdensNaoRealizadas As String = "qry_ContarOrdensNaoRealizadas"
Const cNomeFunc As String = "ContarOrdensNaoRealizadas"

'Variáveis
Dim myRS As Recordset
Dim myQRY As QueryDef
'----

On Error GoTo ErroDesconhecido

'Abriu um recordset de contagem
Set myQRY = pDB.QueryDefs(cConsultaContarOrdensNaoRealizadas)
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Resultado da verificação
ContarOrdensNaoRealizadas = myRS(0)

'Fechar os objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Function

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função identifica o momento para o qual a simulação deve andar

```

```

Private Function AndarContadorDeTempo(By Val pnUltimoMomento As Double, _
    pDB As Database) As Single

'Constantes
Const cConsultaVerificarProximoMomento As String = "qry_VerificarProximoMomento"
Const cConsultaParametroUltimoMomento As String = "Ultimo Momento"
Const cNomeFunc As String = "AndarContadorDeTempo"

'Variaveis
Dim myQRY As QueryDef
Dim myRS As Recordset
'-----

On Error GoTo ErroDesconhecido

'Executar a consulta
Set myQRY = pDB.QueryDefs(cConsultaVerificarProximoMomento)
myQRY.Parameters(cConsultaParametroUltimoMomento) = pnUltimoMomento
Set myRS = myQRY.OpenRecordset(dbOpenForwardOnly, dbReadOnly, dbReadOnly)

'Resultado da verificação
AndarContadorDeTempo = CSng(myRS(0))

'Fechar os objetos
myQRY.Close
myRS.Close

'Encerrar função
Exit Function

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

```

Sequenciador

```

Option Explicit
Const mcModulo As String = "Sequenciador"

'Função mestre para a simulação
Sub SimularProgramacao(By Val pdDataHoraInicial As Date, pForm As Form)

'Constantes
Const cRotuloStatus As String = "lblStatus"
Const cRotuloSubStatus As String = "lblSubStatus"
Const cMensDB As String = "Status: Acessando banco de dados"
Const cMens0 As String = "Status: Inicializando modelo"
Const cMens1 As String = "Status: Gerando ordens de produção"
Const cMens2 As String = "Status: Gerando programa de produção"
Const cMensEmBranco As String = ""

'Variáveis
Dim dbSimulador As Database
Dim dDataHoraInicial As Date
Dim nInicioNormalizado As Single
Const cNomeFunc As String = "SimularProgramacao"
'-----

On Error GoTo ErroDesconhecido

```

Definir o banco de dados do simulador

```
pForm.Controls(cRotuloStatus).Caption = cMensDB
pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco
pForm.Repaint
Set dbSimulador = CurrentDB()
```

FASE 0

Importar o inventário atual do banco de dados da organização

Verificar a consistência do modelo

```
pForm.Controls(cRotuloStatus).Caption = cMens()
pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco
pForm.Repaint
Call InicializarModelo(pForm, dbSimulador)
```

FASE 1

Gerar as encomendas internas diretas (entradas do usuário)
e indiretas (necessidades para as encomendas diretas)

Normalização das datas em relação ao início da simulação

```
pForm.Controls(cRotuloStatus).Caption = cMens1
pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco
pForm.Repaint
Call PrepararElementosDeSimulacao(pdlDataHorainicial, inicioNormalizado, _
dbSimulador)
```

FASE 2

Gerar as ordens de produção que compõem as filas nos recursos de trabalho

Faz agrupamento de encomendas internas

```
pForm.Controls(cRotuloStatus).Caption = cMens2
pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco
pForm.Repaint
Call SequenciarOrdensDeProducao(inicioNormalizado, pdlDataHorainicial, _
dbSimulador, pForm)
```

Fechar o objeto

```
pForm.Controls(cRotuloStatus).Caption = cMensEmBranco
pForm.Controls(cRotuloSubStatus).Caption = cMensEmBranco
pForm.Repaint
dbSimulador.Close
```

Encerrar função

Exit Sub

ErroDesconhecido:

```
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

End Sub

Esta função encerra a simulação por algum erro desconhecido

```
Public Sub EncerrarPorErroDesconhecido(ByVal piErro As Long, ByVal psErro As String, _
ByVal psFuncao As String, psModulo As String)
```

Constantes

```
Const cLblStatus = "lblStatus"
Const cLblSubStatus = "lblSubStatus"
Const cNomeForm = "Principal"
```

```
Const cMensStatus As String = "ERRO FATAL!!!"
```

```
Const cMensSubStatus As String = "Não é possível prosseguir - erro desconhecido"
```

```
Const cMensEmBranco As String = ""
```

```
Const cTitulo As String = "Erro desconhecido"
```

```

'Variáveis
Dim sMens As String
'-----

'Atualizar mensagem de status
Application.Forms(cNomeForm).Controls(cLblStatus).Caption = cMensStatus
Application.Forms(cNomeForm).Controls(cLblSubStatus).Caption = cMensSubStatus

'Mensagem de erro
sMens = "Ocorreu um erro desconhecido na função '" & psFuncao & "' (módulo '" & _
psModulo & "' ). Não é possível prosseguir a operação! " & _
"-- ERRO: " & piErro & " - " & psiErro
MsgBox sMens, vbCritical, cTitulo

'Apagar mensagens de status
Application.Forms(cNomeForm).Controls(cLblStatus).Caption = cMensEmBranco
Application.Forms(cNomeForm).Controls(cLblSubStatus).Caption = cMensEmBranco

'Encerrar aplicação
End

End Sub

```

Funções Auxiliares

```

Option Explicit
Const mcModulo As String = "Funcoes_Auxiliares"

'Esta função mantém atualizada tabela de campos da Fila
Sub AtualizarCamposFila()

'Constantes das tabela
Const cTabFilaNome As String = "Fila"           'Nome da tabela de Fila
Const cTabCamposFilaNome As String = "CamposFila" 'Nome da tabela de campos de Fila
Const cNumCamposIniciais As Integer = 7         '# campos que não são critérios
Const cNomeCampo As String = "CmFl_NomeCampo"   'Nome dos campos
Const cLegCampo As String = "CmFl_Legenda"
Const cDescCampo As String = "CmFl_Desc"
Const cNomeFunc As String = "AtualizarCamposFila"

'Variáveis
Dim myDB As Database      'Banco de dados atual
Dim myTBD As TableDef     'Tabela de Fila
Dim myFLD As Field        'Campos da tabela de fila
Dim sSQL As String        'String de SQL
Dim i As Integer          'Índice para o número dos campos
'-----

On Error GoTo ErroDesconhecido

'Seta as variáveis de objeto
Set myDB = CurrentDb()
Set myTBD = myDB.TableDefs(cTabFilaNome)

'Apaga o conteúdo previo da tabela de campos de fila
sSQL = "DELETE * FROM " & cTabCamposFilaNome
myDB.Execute (sSQL)

'Insere os valores atuais, verificando campo por campo
For i = cNumCamposIniciais To myTBD.Fields.Count - 1
Set myFLD = myTBD.Fields(i)
sSQL = "INSERT INTO " & cTabCamposFilaNome & "(" & cNomeCampo & _
"," & cLegCampo & ", " & cDescCampo & ") VALUES (" & _
myTBD.Fields(i).Name & ", " & _

```

```

myTBD.Fields(i).Properties("Caption") & " " & _
myTBD.Fields(i).Properties("Description") & " ")

myDB.Execute (sSQL)

Next

'Fecha os objetos
myDB.Close

'Encerrar função
Exit Sub

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Sub

'Esta função faz a conversão de unidades e retorna 0 em caso de incoerência de grandezas
Public Function ConverterUnidade(ByVal pnValor As Double, _
    ByVal psUnidadeOriginal As String, _
    ByVal psUnidadeDestino As String, _
    ByVal psGrandeza As String) As Double

'Constantes da consulta
Const cConsultaConversao As String = "qry_ConversaoUnidade"
Const cConsultaParametroValor As String = "Valor"
Const cConsultaParametroUnidOriginal As String = "Unidade Original"
Const cConsultaParametroUnidDestino As String = "Unidade Destino"
Const cConsultaParametroGrandeza As String = "Grandeza"
Const cNomeFunc As String = "ConverterUnidade"

'Variáveis para execução da consulta
Dim myDB As Database
Dim myQRY As QueryDef
Dim myRS As Recordset
Dim sSQL As String
'----

On Error GoTo ErroDesconhecido

'Define os objetos
Set myDB = CurrentDb()
Set myQRY = myDB.QueryDefs(cConsultaConversao)
myQRY.Parameters(cConsultaParametroValor) = pnValor
myQRY.Parameters(cConsultaParametroUnidOriginal) = psUnidadeOriginal
myQRY.Parameters(cConsultaParametroUnidDestino) = psUnidadeDestino
myQRY.Parameters(cConsultaParametroGrandeza) = psGrandeza

'Faz a conversão
Set myRS = myQRY.OpenRecordset
ConverterUnidade = If(IsNull(myRS(0)), 0, myRS(0))

'Fecha os objetos
myRS.Close
myQRY.Close
myDB.Close

'Encerrar função
Exit Function

ErroDesconhecido:
Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

```

End Function

'Esta função retorna o valor de um tempo em uma unidade convertido para dias

Public Function ConverterTempoParaDia(ByVal pnValor As Double, _
ByVal psUnidade As String) As Double

'Constantes dos nomes das variáveis

Const cNomeSegundo As String = "seg"

Const cNomeMinuto As String = "min"

Const cNomeHora As String = "h"

Const cNomeDia As String = "dia"

Const cNomeSemana As String = "sem"

'Constantes dos valores de conversão das variáveis

Const cConvSegundo As Long = 86400

Const cConvMinuto As Integer = 1440

Const cConvHora As Integer = 24

Const cConvDia As Integer = 1

Const cConvSemana As Double = 0.142857142857

Const cNomeFunc As String = "ConverterTempoParaDia"

'----

On Error GoTo ErroDesconhecido

'Faz a conversão

Select Case psUnidade

Case cNomeSegundo

ConverterTempoParaDia = pnValor / cConvSegundo

Case cNomeMinuto

ConverterTempoParaDia = pnValor / cConvMinuto

Case cNomeHora

ConverterTempoParaDia = pnValor / cConvHora

Case cNomeDia

ConverterTempoParaDia = pnValor / cConvDia

Case cNomeSemana

ConverterTempoParaDia = pnValor / cConvSemana

End Select

'Encerrar função

Exit Function

ErroDesconhecido:

Call EncerrarPorErroDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)

End Function

'Esta função retorna o valor de um tempo em uma unidade convertido para dias

Public Function ConverterTempoParaHora(ByVal pnValor As Double, _
ByVal psUnidade As String) As Long

'Constantes dos nomes das variáveis

Const cNomeSegundo As String = "seg"

Const cNomeMinuto As String = "min"

Const cNomeHora As String = "h"

Const cNomeDia As String = "dia"

Const cNomeSemana As String = "sem"

'Constantes dos valores de conversão das variáveis

Const cConvSegundo As Long = 3600

Const cConvMinuto As Integer = 60

Const cConvHora As Integer = 1

Const cConvDia As Double = 1 / 24

Const cConvSemana As Double = 1 / 7 / 24

```
Const cNomeFunc As String = "ConverterTempoParaHora"
```

```
'Variaveis
```

```
Dim nConversao As Double
```

```
'-----
```

```
On Error GoTo ErroiDesconhecido
```

```
'Fazer a conversão
```

```
    Select Case psUnidade
```

```
        Case cNomeSegundo
```

```
            nConversao = pnValor / cConvSegundo
```

```
        Case cNomeMinuto
```

```
            nConversao = pnValor / cConvMinuto
```

```
        Case cNomeHora
```

```
            nConversao = pnValor / cConvHora
```

```
        Case cNomeDia
```

```
            nConversao = pnValor / cConvDia
```

```
        Case cNomeSemana
```

```
            nConversao = pnValor / cConvSemana
```

```
    End Select
```

```
'Arredondar e transformar em inteiro
```

```
    ConverterTempoParaHora = CInt(1000 * nConversao)
```

```
'Encerrar função
```

```
Exit Function
```

```
ErroiDesconhecido:
```

```
    Call EncerrarPorErroiDesconhecido(Err, Err.Description, cNomeFunc, mcModulo)
```

```
End Function
```

ANEXO B - FORMULÁRIOS

Item

Código: CD1110

Nome: cadr prb, pe smp, est smp, verniz

Descrição: Cadeira em peroba, pés simples, estrutura de encosto simples, acabamento em verniz

Categoria: PA Produto Acabado

Família: CADR cadeiras

Grandeza: un Unidade

Unidade de medida: un Unidade

Custo de inventário: 0,63

Registro: 1 de 171

Figura D.1 – Formulário de item

Operação

Código da operação: OPCD1110

Operação: Acab final - CD1110

Produto final: CD1110 cadr prb, pe smp, est smp, ve

Descrição: Dar acabamento final ao item CD1110 - cadr prb, pe smp, est smp, verniz

Recurso de Trabalho: MSA001 Mesa op

Tempos

Fila esperada: 480,00

Set-Up (mesmo item): 0,00

Set-Up (mesma família): 0,00

Set-Up (item diverso): 10,00

Tempo de processo: 10,80

Tempo constante: 0,00

Clean-Up: 8,00

Unidade de tempo: min Minuto

Limitações

Lote mínimo: 0,00

Lote máximo: 0,00

Quantidade múltipla: 0,00

EPP: 5,74

Registros: 1 de 144

Figura D.2 – Formulário de operação

Entrada de operações

Operação: DPCD1110 Acab final - CD1110

Quantidade Base: 1.00

Ordem de entrada: 1

Produto de Entrada: CDSA110 cadr S/A prb. pe smp. est smi

Quantidade de Entrada: 1.00

Registro: 1 de 296

Figura D.3 – Formulário de entrada de operações

Recurso de Trabalho

Código: FRM001

Recurso de Trabalho: Forma

Centro de Trabalho: MQPS Maq pesadas

Descrição: Prensa para dar forma "envergada" a encosto de cadeira

Máquinas disponíveis: 1

Regra de Sequênciação: FLDN Menor folga dinâmica

Registro: 1 de 8

Figura D.4 – Formulário de recurso de trabalho

Disponibilidade de trabalho

Disponibilidade diária

Dia	Início	Fim	24 horas
Domingo	00:00	00:00	☐
Segunda	08:00	18:00	☐
Terça	08:00	18:00	☐
Quarta	08:00	18:00	☐
Quinta	08:00	18:00	☐
Sexta	08:00	18:00	☐
Sábado	08:00	12:00	☐

OK Cancelar

Figura D.5 – Formulário de disponibilidade

Família

Código: CADR

Família: cadeiras

Descrição: Produto acabado (pronto para comercialização) do tipo "cadeira"

Registro: 1 de 28

Figura D.6 – Formulário de família

Centro de trabalho

Código: BNCD

Centro de Trabalho: Bancada de montagem

Descrição: Bancada de mesas de operações de montagem

Registro: 1 de 5

Figura D.7 – Formulário de centro de trabalho

Categoria

Código: ME

Nome: Mat. Embalagem

Origem externa: ☒

Descrição: Todo tipo de material de embalagem: plástico, papelão

Registro: 1 de 4

Figura D.8 – Formulário de categoria

Grandeza

Sigla: A

Grandeza: Área

Expoente M: 0

Expoente L: 2

Expoente T: 0

Registro: 1 de 6

Figura D.9 – Formulário de grandeza

Unidade

Sigla: cm

Unidade: centímetro

Grandeza: comp Comprimento

Unidade Padrão: ☐

Equivalência Padrão: 0,01

Registro: 1 de 10

Figura D.10 – Formulário de unidade

Regra de Sequenciação

Código: DTLM

Descrição: Menor data de conclusão

1º critério: Fila_DataMaxOP Data limite para ordem ☐ Maximizar

2º critério: Fila_TempoProcesso Tempo de processo ☐ Maximizar

Registro: 1 de 5

Figura D.11 – Formulário de regra de sequenciação

Verificacao

Tabela:

Campo chave:

Registro:

Complemento:

Descrição do erro:

Registro: 1 de 1

Figura D.12 – Formulário de verificação

The image shows a software window titled "LeadTime_MP". Inside the window, there is a form with the following fields:

- Item:** A text box containing "XAC0100" followed by a dropdown menu showing "verniz".
- Prazo de entrega:** A text box containing the number "5".
- Registro:** A section at the bottom with navigation buttons (back, forward, etc.) and a status indicator showing "1" and "de 27".

Figura D.13 – Formulário de *lead-time* de matéria-prima

ANEXO C - RELATÓRIOS

Os relatórios seguintes são, respectivamente:

- Cumprimento de datas de entrega – 1ª configuração
- Cumprimento de datas de entrega – 2ª configuração
- Cumprimento de datas de entrega – 3ª configuração
- Performance de recursos de trabalho – 1ª configuração
- Performance de recursos de trabalho – 2ª configuração
- Performance de recursos de trabalho – 3ª configuração
- Ocupação de recursos de trabalho – 1ª configuração
- Ocupação de recursos de trabalho – 2ª configuração
- Ocupação de recursos de trabalho – 3ª configuração
- Programação de produção centro de trabalho – 1ª configuração
- Programação de produção por data – 1ª configuração
- Necessidade de disponibilidade de matéria-prima – 1ª configuração

Cumprimento de datas de entrega

Encomenda	Ordem	Item	Quantidade	Data de entrega	Entrega programada	Atraso (dias)
37	8 092	CD1420 - cadr pub. pe trib. est trib. beco	60.000	08 dez 00 12.00	13 dez 00 08.51	4,9
38	8 093	CD2110 - cadr pub. pe sup. est sup. verniz	150.000	08 dez 00 12.00	04 dez 00 12.07	0,0
39	8 094	CD1110 - cadr pub. pe sup. est sup. verniz	120.000	08 dez 00 12.00	06 dez 00 14.01	0,0
40	8 094	CD2310 - cadr pub. pe sup. est trib. verniz	90.000	08 dez 00 12.00	08 dez 00 10.31	0,0
41	8 098	MS6220 - mesa prb red esp. pe prb red. beco	15.000	08 dez 00 12.00	11 dez 00 17.45	3,2
42	8 097	MS3410 - mesa pub quad fin. pe prb red. verniz	25.000	08 dez 00 12.00	11 dez 00 14.01	3,1
43	8 095	MS2210 - mesa prb red fin. pe prb red. verniz	30.000	08 dez 00 12.00	12 dez 00 14.16	4,1
44	8 096	MS3310 - mesa pub quad fin. pe prb quad. verniz	15.000	08 dez 00 12.00	12 dez 00 10.15	3,9
45	8 134	MST A110 - tanpo prb quad fin	10.000	01 dez 00 12.00	07 dez 00 14.22	6,1
46	8 135	MST A120 - tanpo prb quad fin	10.000	01 dez 00 12.00	07 dez 00 13.18	6,1
47	8 139	MS1 A410 - tanpo prb red esp	10.000	01 dez 00 12.00	07 dez 00 12.14	6,0
48	8 121	CDP A200 - pe cadr prb trib	320.000	01 dez 00 12.00	08 dez 00 09.35	6,9
49	8 122	CDP A210 - pe cadr prb trib	160.000	01 dez 00 12.00	06 dez 00 10.19	4,9
50	8 116	CDP A100 - pe cadr prb smp	400.000	01 dez 00 12.00	05 dez 00 14.32	4,1
51	8 119	CDP A110 - pe cadr prb smp	320.000	01 dez 00 12.00	04 dez 00 16.28	3,2

Total de Encomendas: 15

Encomendas com atraso: 12 (80%)

Atraso médio: 3,8

Atraso máximo: 6,9

Cumprimento de datas de entrega

Encomenda	Ordem	Item	Quantidade	Data de entrega	Entrega programada	Atraso (dias)
37	8 566	C1D1420 - cadr pub, pe trib, est trib, beo	60,0000	08 dez 00 12:00	08 dez 00 15:53	0,2
38	8 567	C1D2110 - cadr pub, pe snup, est snup, verniz	150,0000	08 dez 00 12:00	04 dez 00 11:52	0,0
39	8 565	C1D1110 - cadr pub, pe snup, est snup, verniz	120,0000	08 dez 00 12:00	05 dez 00 08:48	0,0
40	8 568	C1D2310 - cadr pub, pe snup, est trib, verniz	90,0000	08 dez 00 12:00	09 dez 00 11:53	1,0
41	8 572	MS6220 - mesa pub red esp, pe pub red, beo	15,0000	08 dez 00 12:00	07 dez 00 14:47	0,0
42	8 571	MS3110 - mesa pub quad fin, pe pub red, verniz	25,0000	08 dez 00 12:00	08 dez 00 08:36	0,0
43	8 569	MS2210 - mesa pub red fin, pe pub red, verniz	30,0000	08 dez 00 12:00	08 dez 00 12:09	0,0
44	8 570	MS3310 - mesa pub quad fin, pe pub quad, verniz	15,0000	08 dez 00 12:00	07 dez 00 15:23	0,0
45	8 608	MST1110 - lampo pub quad fin	10,0000	01 dez 00 12:00	27 nov 00 12:26	0,0
46	8 609	MST1120 - lampo pub quad fin	10,0000	01 dez 00 12:00	27 nov 00 11:13	0,0
47	8 613	MST1110 - lampo pub red esp	10,0000	01 dez 00 12:00	27 nov 00 10:00	0,0
48	8 595	C1DP1200 - pe cadr pub trib	320,0000	01 dez 00 12:00	05 dez 00 17:55	4,2
49	8 596	C1DP1210 - pe cadr pub trib	160,0000	01 dez 00 12:00	01 dez 00 15:07	0,1
50	8 590	C1DP1100 - pe cadr pub snp	400,0000	01 dez 00 12:00	05 dez 00 15:44	4,2
51	8 593	C1DP1110 - pe cadr pub snp	320,0000	01 dez 00 12:00	04 dez 00 17:40	3,2

Total de Encomendas: 15

Encomendas com atraso: 7 (47%)

Atraso médio: 0,9

Atraso máximo: 4,2

Cumprimento de datas de entrega

Encomenda	Ordem	Item	Quantidade	Data de entrega	Entrega programada	Atraso (dias)
37	8 923	CD1420 - cadr prb, pe tub, est trb, beo	60,000	08 dez 00 12:00	08 dez 00 17:10	0.2
38	8 924	CD2110 - cadr prb, pe smp, est smp, verniz	150,000	08 dez 00 12:00	04 dez 00 12:08	0.0
39	8 922	CD1110 - cadr pub, pe smp, est smp, verniz	120,000	08 dez 00 12:00	05 dez 00 15:07	0.0
40	8 925	CD2310 - cadr prb, pe smp, est trb, verniz	90,000	08 dez 00 12:00	08 dez 00 17:39	0.2
41	8 929	MS6220 - mesa prb red esp, pe prb red, beo	15,000	08 dez 00 12:00	09 dez 00 08:29	0.9
42	8 928	MS3410 - mesa prb quad fin, pe prb red, verniz	25,000	08 dez 00 12:00	08 dez 00 15:15	0.1
43	8 926	MS2210 - mesa prb red fin, pe prb red, verniz	30,000	08 dez 00 12:00	08 dez 00 14:35	0.1
44	8 927	MS3310 - mesa prb quad fin, pe prb quad, verniz	15,000	08 dez 00 12:00	09 dez 00 09:09	0.9
45	8 965	MST A110 - tempo prb quad fin	10,000	01 dez 00 12:00	28 nov 00 09:13	0.0
46	8 966	MST A120 - tempo prb quad fin	10,000	01 dez 00 12:00	27 nov 00 11:13	0.0
47	8 970	MST A410 - tempo prb red esp	10,000	01 dez 00 12:00	27 nov 00 10:00	0.0
48	8 952	CDP A200 - pe cadr prb tub	320,000	01 dez 00 12:00	05 dez 00 09:14	3.9
49	8 953	CDP A210 - pe cadr prb tub	160,000	01 dez 00 12:00	01 dez 00 13:50	0.1
50	8 947	CDP A100 - pe cadr prb smp	400,000	01 dez 00 12:00	05 dez 00 11:04	4.0
51	8 950	CDP A110 - pe cadr prb smp	320,000	01 dez 00 12:00	05 dez 00 17:32	4.2

Total de Encomendas: 15

Encomendas com atraso: 10 (67%)

Atraso médio: 1.0

Atraso máximo: 4.2

Performance de recursos de trabalho

Centro de Trabalho	Recurso de Trabalho	Tempo em fila	Tempo em operação	# médio ordens
MOPIS - Maq pesadas	FRM001 - Forma (1)	1.03	1.2	0.5
	IND001 - Usadeira plana (1)	49.92	4.7	11.6
	IND002 - Us cilíndrica (1)	4.70	3.1	2.5
BNCID - Bancada de montagem	MIS A001 - Mesa op (1)	5.48	16.6	1.3
	MIS A001 - Mesa op (2)	4.55	9.9	1.3
	MIS A001 - Mesa op (3)	3.53	10.7	1.0
	SRA001 - Serra de fita (1)	35.35	1.7	22.3
TORNO - Tornos	SRA002 - Serra de mesa (1)	1.55	0.6	3.7
	TRN001 - Torno manual (1)	17.22	12.4	2.4
	TRN002 - Torno automatico (1)	8.77	7.8	2.1

Performance de recursos de trabalho

Centro de Trabalho	Recurso de Trabalho	Tempo em fila	Tempo em operação	# médio ordens
MOPPS - Maq pesadas	FRA001 - Forma (1)	1,21	1,2	0,6
	IND001 - Lixadeira plana (1)	6,60	7,2	1,8
	IND001 - Lixadeira plana (2)	6,17	3,5	2,7
	IND002 - Lix cilíndrica (1)	1,70	3,1	2,5
BNCID - Bancada de montagem	MIS A001 - Mesa op (1)	5,59	11,4	1,4
	MIS A001 - Mesa op (2)	9,85	13,1	1,4
	MIS A001 - Mesa op (3)	9,04	11,5	1,3
	MIS A001 - Mesa op (4)	7,99	8,9	1,3
SERRA - Serras	SRA001 - Serra de fita (1)	35,35	1,7	22,3
	SRA002 - Serra de mesa (1)	1,55	0,6	3,7
TORNO - Tornos	TRN001 - Torno manual (1)	2,84	12,8	1,2
	TRN001 - Torno manual (2)	5,80	12,2	1,5
	TRN002 - Torno automatic (1)	9,96	7,8	2,3

Performance de recursos de trabalho

Centro de Trabalho	Recurso de Trabalho	Tempo em fila	Tempo em operação	# médio ordens
ALQPS - Maq pesadas	TRM001 - Formã (1)	1,32	1,2	0,7
	LND001 - Livadeira plana (1)	7,75	4,5	2,7
	LND001 - Livadeira plana (2)	5,37	5,0	1,9
	LND002 - Liv. cilíndrica (1)	4,62	3,4	2,5
LINC D - Bancada de montagem	MIS A001 - Mesa op (1)	6,20	17,0	1,4
	MIS A001 - Mesa op (2)	12,09	13,7	1,5
	MIS A001 - Mesa op (3)	8,51	10,2	1,3
	MIS A001 - Mesa op (4)	8,55	8,3	1,3
SERRA - Seras	SRA001 - Serra de fila (1)	35,81	1,7	22,5
	SRA002 - Serra de mesa (1)	1,55	0,6	4,7
FORNO - Fornos	FRN001 - Forno manual (1)	6,65	11,0	1,6
	FRN001 - Forno manual (2)	3,21	14,6	1,2
	FRN002 - Forno automático (1)	12,70	7,9	2,6

Ocupação de recursos de trabalho

Centro de Trabalho	Recurso de Trabalho	Tempo em uso	Utilização local	Utilização global
MQPS - Maquinadas	PRM001 - Forma(1)	12.07	28,2%	9,4%
LIND - Lindeiras	LXD001 - Lindeira plana(1)	112.00	100,0%	87,1%
	LXD002 - Lindeira cilíndrica(1)	12.32	100,0%	9,6%
BNC D - Bancada de montagem	MISX001 - Mesa op(1)	127.73	99,4%	99,1%
	MISX001 - Mesa op(2)	108.71	87,8%	84,4%
	MISX001 - Mesa op(3)	85.38	71,5%	66,3%
SERRA - Serras	SRX001 - Serra de fita(1)	56.45	100,0%	43,8%
	SRX002 - Serra de mesa(1)	3.97	100,0%	3,1%
TORNO - Tornos	TRN001 - Torno manual(1)	62.07	100,0%	48,2%
	TRN002 - Torno automático(1)	54.87	100,0%	42,6%

Ocupação local - parcela de tempo ocupado entre o início da primeira operação e a conclusão da última operação do recurso de trabalho

Ocupação global - parcela de tempo ocupado entre o início da primeira operação e a conclusão da última operação de todos os recursos de trabalho

Ocupação de recursos de trabalho

Centro de Trabalho	Recurso de Trabalho	Tempo em uso	Utilização local	Utilização global
ALQPS - Maq pesadas	FRN001 - Forno(1)	12,07	33,1%	11,5%
	JND001 - Travadeira plana(1)	58,82	98,5%	55,9%
	JND001 - Travadeira plana(2)	54,59	91,3%	51,9%
	JND002 - Trav. cilíndrica(1)	12,32	100,0%	14,7%
BNC D - Bancada de montagem	MIS A001 - Mesa opt(1)	102,27	99,1%	97,3%
	MIS A001 - Mesa opt(2)	82,48	80,2%	78,3%
	MIS A001 - Mesa opt(3)	71,25	68,7%	67,8%
	MIS A001 - Mesa opt(4)	66,18	63,6%	62,9%
SEERRA - Seras	SR A001 - Serra de fita(1)	56,60	100,0%	53,8%
	SR A002 - Serra de mesa(1)	3,97	100,0%	3,8%
TORXO - Fornos	TRN001 - Forno manual(1)	33,00	100,0%	31,4%
	TRN001 - Forno manual(2)	29,20	100,0%	27,8%
	TRN002 - Forno automatic(1)	55,13	100,0%	52,4%

Ocupação local: parcela de tempo ocupado entre o início da primeira operação e a conclusão da última operação do recurso de trabalho

Ocupação global: parcela de tempo ocupado entre o início da primeira operação e a conclusão da última operação de todos os recursos de trabalho

Ocupação de recursos de trabalho

Centro de Trabalho	Recurso de Trabalho	Tempo em uso	Utilização local	Utilização global
MQPS - Maq pesadas	FRM001 - Format(1)	12,07	28,1%	11,2%
	IND001 - Fixadeira planat(1)	57,94	95,4%	53,7%
IND - Fixadeiras	IND001 - Fixadeira planat(2)	55,17	95,6%	51,4%
	IND002 - Fix cilíndrica(1)	12,32	100,0%	11,4%
BNCID - Bancada de montagem	MISX001 - Mesa op(1)	100,77	99,3%	93,4%
	MISX001 - Mesa op(2)	78,78	80,6%	73,0%
	MISX001 - Mesa op(3)	80,25	74,9%	74,4%
	MISX001 - Mesa op(4)	62,28	66,4%	57,7%
SILRRV - Serpis	SRX001 - Serpa de fixat(1)	56,45	100,0%	52,3%
	SRX002 - Serpa de mesa(1)	3,97	100,0%	3,7%
TORNO - Tornos	TRX001 - Torno manual(1)	25,60	100,0%	23,7%
	TRX001 - Torno manual(2)	36,60	100,0%	33,9%
	TRX002 - Torno automatico(1)	54,87	100,0%	50,9%

(Ocupação local: parcela de tempo ocupado entre o início da primeira operação e a conclusão da última operação do recurso de trabalho)

(Ocupação global: parcela de tempo ocupado entre o início da primeira operação e a conclusão da última operação de todos os recursos de trabalho)

Programação de produção por centro de trabalho

Centro de Trabalho: BNCD - Bancada de montagem

Recurso de Trabalho: MSA001 - Mesa op (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8149	CDAS110	asento S \ A pinho	150,000	27 11 00 08:25	28 11 00 10:48
8147	CDAS100	asento S \ A peroba	120,000	28 11 00 10:48	29 11 00 10:31
8150	CDAS110	asento S \ A pinho	90,000	29 11 00 11:14	30 11 00 08:34
8101	CDS\A210	cadr S \ A pub, pe simp, est simp	150,000	30 11 00 08:34	06 12 00 12:47
8102	CDS\A230	cadr S \ A pub, pe simp, est arb	90,000	06 12 00 12:47	09 12 00 11:43
8097	MS3410	mesa pub quad lin, pe pub red, vcmz	25,000	09 12 00 11:43	11 12 00 14:01
8098	MS6220	mesa pub red esp, pe pub red, beo	15,000	11 12 00 14:01	11 12 00 17:45
8092	CD1420	cadr pub, pe arb, est arb, beo	60,000	11 12 00 17:45	13 12 00 08:51

Programação de produção por centro de trabalho

Recurso de Trabalho: MSA001 - Mesa op (2)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 184	MS1S410	tampo S A prb red esp	10,000	27 11 00 08:25	27 11 00 11:01
8 179	MS1S410	tampo S A prb quad fin	10,000	27 11 00 11:01	27 11 00 13:38
8 181	MS1S120	tampo S A prb quad fin	25,000	27 11 00 13:38	28 11 00 09:51
8 182	MS1S120	tampo S A prb quad fin	15,000	28 11 00 09:51	28 11 00 13:40
8 093	CD2110	cadi pulh. pe smp. est smp. verniz	150,000	30 11 00 08:49	04 12 00 12:07
8 091	CD1110	cadi prb. pe smp. est smp. verniz	120,000	04 12 00 12:07	06 12 00 14:01
8 094	CD2310	cadi pulh. pe smp. est trib. verniz	90,000	06 12 00 14:01	08 12 00 10:34
8 103	MS1A220	mesa S A prb red fin. pe prb red. verniz	30,000	08 12 00 10:31	09 12 00 09:08
8 105	MS1A340	mesa S A prb quad fin. pe prb red. verniz	25,000	09 12 00 09:08	11 12 00 12:21
8 104	MS1A330	mesa S A prb quad fin. pe prb quad. verniz	15,000	11 12 00 12:21	11 12 00 16:46
8 095	MS2210	mesa prb red fin. pe prb red. verniz	30,000	11 12 00 16:46	12 12 00 14:16

Programação de produção por centro de trabalho

Recurso de Trabalho: MSA001 - Mesa op (3)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8180	MSTS120	tampo S A pnh quad fin	10,000	27 11 00 08:47	27 11 00 11:24
8183	MSTS210	tampo S A prb red fin	30,000	27 11 00 11:24	28 11 00 08:48
8185	MSTS410	tampo S A prb red esp	15,000	28 11 00 08:48	28 11 00 12:37
8148	CDSAS100	asento S A peroba	60,000	30 11 00 14:28	01 12 00 09:39
8099	CDSAS110	cadr S A prb, pe smp, est smp	120,000	02 12 00 11:55	07 12 00 16:31
8100	CDSAS140	cadr S A prb, pe tub, est tub	60,000	07 12 00 16:31	11 12 00 11:56
8106	MSSA620	mesa S A prb red esp, pe prb red, verniz	15,000	11 12 00 11:56	11 12 00 16:21
8096	MSS3310	mesa pnh quad fin, pe pnh quad, verniz	15,000	11 12 00 16:21	12 12 00 10:15

Programação de produção por centro de trabalh

Centro de Trabalho: LIXD - Lixadeiras

Recurso de Trabalho: LXD001 - Lixadeira plana (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 109	CDA V110	asento pinho	150,000	27 11 00 08:47	28 11 00 08:00
8 113	CDE V110	ene cada pnh	150,000	28 11 00 08:00	28 11 00 17 13
8 129	CDV V110	trav cada pnh	960,000	28 11 00 17:13	30 11 00 16:38
8 111	CDE V100	ene cada pnb	120,000	30 11 00 16:38	01 12 00 14:03
8 107	CDA V100	asento peroba	120,000	01 12 00 14:03	02 12 00 11 28
8 127	CDV V100	trav cada pnb	480,000	02 12 00 11 28	04 12 00 17 17
8 110	CDA V110	asento pinho	90,000	04 12 00 17:17	05 12 00 12:54
8 114	CDE V110	ene cada pnh	90,000	05 12 00 12 54	06 12 00 08:31
8 112	CDE V100	ene cada pnb	60,000	06 12 00 08:31	06 12 00 12:11
8 108	CDA V100	asento peroba	60,000	06 12 00 12 11	06 12 00 16:00
8 128	CDV V100	trav cada pnb	240,000	06 12 00 16:00	07 12 00 11 01
8 139	MST A410	tampo pub red esp	10,000	07 12 00 11:01	07 12 00 12 14
8 135	MST V120	tampo pnh quad fin	10,000	07 12 00 12:14	07 12 00 13:18
8 134	MST V110	tampo pub quad fin	10,000	07 12 00 13:18	07 12 00 14:22
8 138	MST A210	tampo pub red fin	30,000	07 12 00 14:22	07 12 00 17:26
8 131	MST A210	pe ms pnh red	120,000	07 12 00 17:26	08 12 00 14 39
8 141	MST V120	trv ms pnh quad crt	50,000	08 12 00 14 39	08 12 00 14:52

Programação de produção por centro de trabalho

Recurso de Trabalho: LXD001 - Lixadeira plana (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8143	MISVA220	trv ms pub quad lug	50,000	08 12 00 14:52	08 12 00 16:26
8136	MISVA120	tampa pub quad fin	25,000	08 12 00 16:26	09 12 00 09 09
8142	MISVA120	trv ms pub quad crt	30,000	09 12 00 09 09	09 12 00 09 58
8144	MISVA220	trv ms pub quad lug	30,000	09 12 00 09 58	09 12 00 10 56
8140	MISVA410	tampa pub red esp	15,000	09 12 00 10 56	11 12 00 08 39
8137	MISVA120	tampa pub quad fin	15,000	11 12 00 08 39	11 12 00 10 13
8132	MISPA210	pe ms pub red	60,000	11 12 00 10 13	11 12 00 13 26

Recurso de Trabalho: LXD002 - Lix cilíndrica (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8133	MISPA220	pe ms pub red	100,000	02 12 00 11:04	04 12 00 12 17
8145	MISVA310	trv ms pub red	120,000	04 12 00 12 17	04 12 00 14 54
8146	MISVA310	trv ms pub red	60,000	04 12 00 14 54	04 12 00 16 10
8130	MISPA120	pe ms pub quad	60,000	04 12 00 16 10	05 12 00 09 23

Programação de produção por centro de trabalho

Centro de Trabalho: MQPS - Maq pesadas

Recurso de Trabalho: FRM001 - Forma (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8159	C'DL'S110	ene cada S A prb	50,000	27 11 00 11:24	27 11 00 12:40
8158	C'DL'S110	ene cada S A prb	50,000	27 11 00 12:40	27 11 00 13:52
8157	C'DL'S110	ene cada S A prb	50,000	27 11 00 13:52	27 11 00 15:04
8151	C'DL'S100	ene cada S A prb	20,000	29 11 00 08:50	29 11 00 10:02
8153	C'DL'S100	ene cada S A prb	50,000	29 11 00 10:02	29 11 00 11:14
8152	C'DL'S100	ene cada S A prb	50,000	29 11 00 11:14	29 11 00 12:26
8161	C'DL'S110	ene cada S A prb	50,000	30 11 00 12:01	30 11 00 13:16
8160	C'DL'S110	ene cada S A prb	40,000	30 11 00 13:16	30 11 00 14:28
8154	C'DL'S100	ene cada S A prb	10,000	01 12 00 11:44	01 12 00 12:56
8155	C'DL'S100	ene cada S A prb	50,000	01 12 00 12:56	01 12 00 14:08

Programação de produção por centro de trabalho

Centro de Trabalho: SERRA - Serras

Recurso de Trabalho: SRA001 - Serra de fita (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 194	CDAN110	asento N A pinho	150,000	27 11 00 08:00	27 11 00 11:07
8 200	CDEN110	ene cadr N A pulv	50,000	27 11 00 11:07	27 11 00 12:14
8 201	CDEN110	ene cadr N A pulv	100,000	27 11 00 12:14	27 11 00 14:18
8 174	CDVN110	trav cadr N A pulv	960,000	27 11 00 14:18	28 11 00 10:10
8 192	CDAN100	asento N A peroba	120,000	28 11 00 10:10	28 11 00 12:41
8 165	CDPN110	pe cadr N A pulv snp	600,000	28 11 00 12:41	28 11 00 16:24
8 170	CDEN110	est ene cadr N A pulv snp	300,000	28 11 00 16:24	29 11 00 08:19
8 196	CDEN100	ene cadr N A prb	20,000	29 11 00 08:19	29 11 00 08:50
8 197	CDEN100	ene cadr N A prb	100,000	29 11 00 08:50	29 11 00 10:54
8 195	CDAN110	asento N A pinho	90,000	29 11 00 10:54	29 11 00 12:49
8 162	CDPN100	pe cadr N A prb snp	480,000	29 11 00 12:49	29 11 00 15:49
8 172	CDVN100	trav cadr N A prb	480,000	29 11 00 15:49	30 11 00 08:49
8 168	CDPN100	est ene cadr N A prb snp	240,000	30 11 00 08:49	30 11 00 10:22
8 171	CDEN110	est ene cadr N A pulv snp	180,000	30 11 00 10:22	30 11 00 11:31
8 202	CDEN110	ene cadr N A pulv	90,000	30 11 00 11:31	30 11 00 13:26
8 193	CDAN100	asento N A peroba	60,000	30 11 00 13:26	30 11 00 14:45
8 166	CDPN110	pe cadr N A pulv snp	520,000	30 11 00 14:45	30 11 00 17:59

Programação de produção por centro de trabalho

Recurso de Trabalho: SRA001 - Serra de fita (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 163	CDPN100	pe cadr N A prb smp	500,000	30 11 00 17:59	01 12 00 11:25
8 198	CDEN100	ene cadr N A prb	10,000	01 12 00 11:25	01 12 00 11:44
8 199	CDEN100	ene cadr N A prb	50,000	01 12 00 11:44	01 12 00 12:48
8 169	CDTN100	est ene cadr N A pub smp	120,000	01 12 00 12:48	01 12 00 13:38
8 173	CDVN100	trav cadr N A prb	240,000	01 12 00 13:38	01 12 00 15:12
8 164	CDPN100	pe cadr N A prb smp	400,000	01 12 00 15:12	01 12 00 17:43
8 167	CDPN110	pe cadr N A pub smp	320,000	01 12 00 17:43	02 12 00 09:42
8 176	MSPN210	pe N A ms prb red	108,000	02 12 00 09:42	02 12 00 10:28
8 178	MSPN220	pe N A ms pub red	90,000	02 12 00 10:28	02 12 00 11:04
8 190	MSPN310	trv ms N A prb red	120,000	02 12 00 11:04	02 12 00 11:55
8 188	MSPN220	trv ms N A pub quad lng	50,000	02 12 00 11:55	04 12 00 08:17
8 186	MSPN120	trv ms N A pub quad crt	50,000	04 12 00 08:17	04 12 00 08:39
8 175	MSPN120	pe N A ms pub quad	54,000	04 12 00 08:39	04 12 00 09:05
8 177	MSPN210	pe N A ms pub red	54,000	04 12 00 09:05	04 12 00 09:28
8 191	MSPN310	trv ms N A prb red	60,000	04 12 00 09:28	04 12 00 09:57
8 189	MSPN220	trv ms N A pub quad lng	30,000	04 12 00 09:57	04 12 00 10:12
8 187	MSPN120	trv ms N A pub quad crt	30,000	04 12 00 10:12	04 12 00 10:27

Programação de produção por centro de trabalho

Recurso de Trabalho: SRA002 - Serra de mesa (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 208	MSTN410	tempo N A pub red esp	10,000	27 11 00 08:00	27 11 00 08:25
8 201	MSTN120	tempo N A pub quad fin	10,000	27 11 00 08:25	27 11 00 08:47
8 203	MSTN110	tempo N A pub quad fin	10,000	27 11 00 08:47	27 11 00 09 09
8 207	MSTN210	tempo N A pub red fin	30,000	27 11 00 09:09	27 11 00 10:07
8 205	MSTN120	tempo N A pub quad fin	25,000	27 11 00 10:07	27 11 00 10:56
8 209	MSTN410	tempo N A pub red esp	15,000	27 11 00 10:56	27 11 00 11 27
8 206	MSTN120	tempo N A pub quad fin	15,000	27 11 00 11:27	27 11 00 11:58

Programação de produção por centro de trabalho

Centro de Trabalho: TORNO - Tornos

Recurso de Trabalho: TRN001 - Torno manual (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 126	CDT A210	est enc cadr pub trib	180,000	30 11 00 11:31	01 12 00 12:31
8 120	CDP A200	pe cadr pub trib	240,000	01 12 00 12:31	04 12 00 13:07
8 125	CDT A200	est enc cadr pub trib	120,000	04 12 00 13:07	05 12 00 10:31
8 122	CDP A210	pe cadr pub trib	160,000	05 12 00 10:31	06 12 00 10:19
8 121	CDP A200	pe cadr pub trib	320,000	06 12 00 10:19	08 12 00 09:35

Recurso de Trabalho: TRN002 - Torno automático (1)

Ordem de Produção	Código de item	Item	Quantidade	Início	Conclusão
8 117	CDP A110	pe cadr pub sup	600,000	28 11 00 13:40	29 11 00 15:52
8 124	CDT A110	est enc cadr pub sup	300,000	29 11 00 15:52	30 11 00 12:04
8 123	CDT A100	est enc cadr pub sup	240,000	30 11 00 12:04	30 11 00 16:56
8 115	CDP A100	pe cadr pub sup	480,000	30 11 00 16:56	01 12 00 16:44
8 118	CDP A110	pe cadr pub sup	360,000	01 12 00 16:44	04 12 00 10:00
8 119	CDP A110	pe cadr pub sup	320,000	04 12 00 10:00	04 12 00 16:28
8 116	CDP A100	pe cadr pub sup	400,000	04 12 00 16:28	05 12 00 14:32

Programação de produção por data

Data: segunda-feira, 27 de novembro de 2000

Recurso de Trabalho	Ordem de produçã	Item	Quantidade	Início	Conclusão
FRM001 - Forma (1)	8 159,00	CDES110 - ene cadr S A pub	50,000	27 11 00 11:24	27 11 00 12:40
	8 158,00	CDES110 - ene cadr S A pub	50,000	27 11 00 12:40	27 11 00 13:52
	8 157,00	CDES110 - ene cadr S A pub	50,000	27 11 00 13:52	27 11 00 15:04
LXD001 - Lixadeira plana (	8 109,00	CDAAL10 - aseto pulho	150,000	27 11 00 08:47	28 11 00 08:00
	8 149,00	CDAAL10 - aseto S A pulho	150,000	27 11 00 08:25	28 11 00 10:48
MIS A001 - Mesa op (1)	8 184,00	MISTS410 - tempo S A pub red esp	10,000	27 11 00 08:25	27 11 00 11:01
MIS A001 - Mesa op (2)	8 179,00	MISTS110 - tempo S A pub quad fin	10,000	27 11 00 11:01	27 11 00 13:38
	8 181,00	MISTS120 - tempo S A pub quad fin	25,000	27 11 00 13:38	28 11 00 09:51
MIS A001 - Mesa op (3)	8 180,00	MISTS120 - tempo S A pub quad fin	10,000	27 11 00 08:47	27 11 00 11:24
	8 183,00	MISTS210 - tempo S A pub red fin	30,000	27 11 00 11:24	28 11 00 08:48
SR A001 - Seta de fita (1)	8 194,00	CDAAL10 - aseto N A pulho	150,000	27 11 00 08:00	27 11 00 11:07
	8 200,00	CDEN110 - ene cadr N A pub	50,000	27 11 00 11:07	27 11 00 12:14
	8 201,00	CDEN110 - ene cadr N A pub	100,000	27 11 00 12:14	27 11 00 14:18
SR A002 - Seta de mesa (1)	8 174,00	CDAAL10 - trav cadr N A pulh	960,000	27 11 00 14:18	28 11 00 10:10
	8 208,00	MISTN410 - tempo N A pub red esp	10,000	27 11 00 08:00	27 11 00 08:25
	8 204,00	MISTN120 - tempo N A pub quad fin	10,000	27 11 00 08:25	27 11 00 08:17
	8 203,00	MISTN410 - tempo N A pub quad fin	10,000	27 11 00 08:47	27 11 00 09:09
	8 207,00	MISTN210 - tempo N A pub red fin	30,000	27 11 00 09:09	27 11 00 10:07
	8 205,00	MISTN120 - tempo N A pub quad fin	25,000	27 11 00 10:07	27 11 00 10:56

Programação de produção por data

Data: segunda-feira, 27 de novembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
SR A002 - Serra de mesa (1)	8.209,00	MISTN410 - tempo N A prb red esp	15,000	27/11 00 10:56	27/11 00 11:27
	8.206,00	MISTN120 - tempo N A prb quad lin	15,000	27/11 00 11:27	27/11 00 11:58

Data: terça-feira, 28 de novembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
LND001 - Fivadeira plana (	8.113,00	CDFX110 - ene cada pnh	150,000	28/11 00 08:00	28/11 00 17:13
	8.129,00	CIDV110 - trav cada pnh	960,000	28/11 00 17:13	30/11 00 16:38
MIS A001 - Mesa op (1)	8.147,00	CIDV100 - asito S A peroba	120,000	28/11 00 10:48	29/11 00 10:31
MIS A001 - Mesa op (2)	8.182,00	MISTN120 - tempo S A pnh quad lin	15,000	28/11 00 09:51	28/11 00 13:40
MIS A001 - Mesa op (3)	8.185,00	MISTN410 - tempo S A prb red esp	15,000	28/11 00 08:18	28/11 00 12:37
SR A001 - Serra de fita (1)	8.192,00	CDAN100 - asito N A peroba	120,000	28/11 00 10:10	28/11 00 12:41
	8.165,00	CDPN110 - pe cada N A pnh snp	600,000	28/11 00 12:41	28/11 00 16:24
	8.170,00	CDTN110 - est ene cada N A pnh snp	300,000	28/11 00 16:24	29/11 00 08:19
FRN002 - Forno automatico (	8.117,00	CDPA110 - pe cada pnh snp	600,000	28/11 00 13:40	29/11 00 15:52

Programação de produção por data

Data: quarta-feira, 29 de novembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
IRM001 - Forma (1)	8 151,00	CDES100 - ene cadr S A prb	20,000	29 11 00 08:50	29 11 00 10:02
	8 153,00	CDES100 - ene cadr S A prb	50,000	29 11 00 10:02	29 11 00 11 14
	8 152,00	CDES100 - ene cadr S A prb	50,000	29 11 00 11:14	29 11 00 12:26
	8 150,00	CIDAS110 - asuto S A pinho	90,000	29 11 00 11:14	30 11 00 08:34
MSA001 - Mesa op (1)	8 196,00	CDEEN100 - ene cadr N A prb	20,000	29 11 00 08:19	29 11 00 08:50
	8 197,00	CDEEN100 - ene cadr N A prb	100,000	29 11 00 08:50	29 11 00 10 54
SRX001 - Serra de fita (1)	8 195,00	CIDAN110 - asuto N A pinho	90,000	29 11 00 10:54	29 11 00 12:49
	8 162,00	CIDEN100 - pe cadr N A prb smp	480,000	29 11 00 12:49	29 11 00 15 19
	8 172,00	CIDAN100 - trav cadr N A prb	480,000	29 11 00 15 19	30 11 00 08:49
	8 124,00	CIDAN110 - est ene cadr pinh smp	300,000	29 11 00 15:52	30 11 00 12:04
IRM002 - Forno automatico (1)					

Programação de produção por data

Data: quinta-feira, 30 de novembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
FRX001 - Forno (1)	8 161,00	CDES110 - ene cadr S A pub	50,000	30 11 00 12:04	30 11 00 13:16
	8 160,00	CDES110 - ene cadr S A pub	40,000	30 11 00 13:16	30 11 00 14:28
FXD001 - Favadena plana (	8 111,00	CDEA100 - ene cadr pub	120,000	30 11 00 16:38	01 12 00 14:03
MISA001 - Mesa op (1)	8 101,00	CDS A210 - cadr S A pub, pe sup, est sup	150,000	30 11 00 08:34	06 12 00 12:47
MISA001 - Mesa op (2)	8 093,00	CD2110 - cadr pub, pe sup, est sup, verniz	150,000	30 11 00 08:49	04 12 00 12 07
MISA001 - Mesa op (3)	8 148,00	CDAS100 - asnto S A peroba	60,000	30 11 00 14:28	01 12 00 09:39
SRA001 - Serra de fita (1)	8 168,00	CDEN100 - est ene cadr N A pub sup	240,000	30 11 00 08:49	30 11 00 10:22
	8 171,00	CDEN110 - est ene cadr N A pub sup	180,000	30 11 00 10:22	30 11 00 11:31
	8 202,00	CDEA110 - ene cadr N A pub	90,000	30 11 00 11:31	40 11 00 13:26
	8 193,00	CDAN100 - asnto N A peroba	60,000	30 11 00 13:26	30 11 00 14:45
	8 166,00	CDPN110 - pe cadr N A pub sup	520,000	30 11 00 14:45	30 11 00 17:59
	8 163,00	CDPN100 - pe cadr N A pub sup	560,000	30 11 00 17:59	01 12 00 11 25
URN001 - Torno manual (1)	8 126,00	CDFA210 - est ene cadr pub tub	180,000	30 11 00 11:31	01 12 00 12:31
URN002 - Torno automático (	8 123,00	CDFA100 - est ene cadr pub sup	240,000	30 11 00 12:04	30 11 00 16:56
	8 115,00	CDPA100 - pe cadr pub sup	480,000	30 11 00 16:56	01 12 00 16:44

Programação de produção por data

Data: sexta-feira, 1 de dezembro de 2000

Recurso de Trabalho	Ordem de produçã	Item	Quantidade	Início	Conclusão
FRM001 - Fôrma (1)	8 154,00	CDES100 - ene cadr S A prb	10,000	01 12 00 11:44	01 12 00 12:56
	8 155,00	CDES100 - ene cadr S A prb	50,000	01 12 00 12:56	01 12 00 14:08
	8 107,00	CDA A160 - asento peroba	120,000	01 12 00 14:03	02 12 00 11:28
	8 198,00	CDE N100 - ene cadr N A prb	10,000	01 12 00 11:25	01 12 00 11:44
	8 199,00	CDE N100 - ene cadr N A prb	50,000	01 12 00 11:44	01 12 00 12:48
	8 169,00	CDE N100 - ene cadr N A prb snap	120,000	01 12 00 12:48	01 12 00 13:38
	8 173,00	CDA N100 - trav cadr N A prb	240,000	01 12 00 13:38	01 12 00 15:12
	8 164,00	CDPN100 - pe cadr N A prb snap	400,000	01 12 00 15:12	01 12 00 17:43
	8 167,00	CDPN110 - pe cadr N A prb snap	320,000	01 12 00 17:43	02 12 00 09:42
	8 120,00	CDPA200 - pe cadr prb ub	240,000	01 12 00 12:31	04 12 00 13:07
FRN001 - Torne manual (1)					
FRN002 - Torne automatic (1)	8 118,00	CDPA110 - pe cadr prb snap	360,000	01 12 00 16:44	04 12 00 10:00

Programação de produção por data

Data: sábado, 2 de dezembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
1 NDX001 - Fixadeira plana (	8 127,00	CDVA100 - trav cadr prb	480,000	02 12 00 11:28	04 12 00 17:17
1 NDX002 - Fix cilíndrica (1)	8 133,00	MSP A220 - pe ms pub red	100,000	02 12 00 11:04	04 12 00 12:17
MISA001 - Mesa op (3)	8 099,00	CDS A110 - cadr S A pub, pe smp, est smp	120,000	02 12 00 11:55	07 12 00 16:31
SR A001 - Seta de fila (1)	8 176,00	MSPN210 - pe N A ms prb red	108,000	02 12 00 09:42	02 12 00 10:28
	8 178,00	MSPN220 - pe N A ms pub red	90,000	02 12 00 10:28	02 12 00 11:04
	8 190,00	MISA N310 - trav ms N A prb red	120,000	02 12 00 11:04	02 12 00 11:55
	8 188,00	MISA N220 - trav ms N A pub quad log	50,000	02 12 00 11:55	04 12 00 08:17

Programação de produção por data

Data: segunda-feira, 4 de dezembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
TRN001 - Fuxadeira plana (	8.110,00	CD.A110 - usento pinho	90,000	04/12/00 17:17	05/12/00 12:54
TRN002 - Lix cilíndrica (1)	8.145,00	MISV A310 - trv ms prb red	120,000	04/12/00 12:17	04/12/00 14:54
	8.146,00	MISV A310 - trv ms prb red	60,000	04/12/00 14:54	01/12/00 16:10
	8.130,00	MSP A120 - pe ms prb quad	60,000	04/12/00 16:10	05/12/00 09:23
MIS A001 - Mesa op (2)	8.091,00	CDI1110 - cadr prb - pe snp - est snp - vertuz	120,000	04/12/00 12:07	06/12/00 14:01
SR A001 - Serra de fita (1)	8.186,00	MISV N120 - trv ms N A prb quad ent	50,000	04/12/00 18:17	04/12/00 08:39
	8.175,00	MSPN120 - pe N A ms prb quad	54,000	04/12/00 08:39	04/12/00 09:05
	8.177,00	MSPN210 - pe N A ms prb red	54,000	04/12/00 09:05	04/12/00 09:28
	8.191,00	MISV N310 - trv ms N A prb red	60,000	04/12/00 09:28	04/12/00 09:57
	8.189,00	MISV N220 - trv ms N A prb quad lag	30,000	04/12/00 09:57	04/12/00 10:12
	8.187,00	MISV N120 - trv ms N A prb quad ent	30,000	04/12/00 10:12	04/12/00 10:27
TRN001 - Torno manual (1)	8.125,00	CDI1A200 - est ene cadr prb tub	120,000	04/12/00 13:07	05/12/00 10:31
TRN002 - Torno automatico (	8.119,00	CDP A110 - pe cadr prb snp	320,000	04/12/00 10:00	04/12/00 16:28
	8.116,00	CDP A100 - pe cadr prb snp	400,000	04/12/00 16:28	05/12/00 14:32

Data: terça-feira, 5 de dezembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
TRN001 - Fuxadeira plana (	8.114,00	CDE A110 - ene cadr prb	90,000	05/12/00 12:54	06/12/00 08:31
TRN001 - Torno manual (1)	8.122,00	CDP A210 - pe cadr prb tub	160,000	05/12/00 10:31	06/12/00 10:19

Programação de produção por data

Data: quarta-feira, 6 de dezembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
1 XD001 - Fixadeira plana (	8 112,00	CDE A100 - ene cadr prb	60,000	06/12/00 08:31	06/12/00 12:11
	8 108,00	CD A A100 - asato peroba	60,000	06/12/00 12:11	06/12/00 16:00
	8 128,00	CDV A100 - trav cadr prb	2 40,000	06/12/00 16:00	07/12/00 11:01
MIS A001 - Mesa op (1)	8 102,00	CDS A230 - cadr S A prb, pe smp est trb	90,000	06/12/00 12:47	09/12/00 11:43
MIS A001 - Mesa op (2)	8 094,00	CD2310 - cadr prb, pe smp, est trb, veniz	90,000	06/12/00 14:01	08/12/00 10:31
1 RN001 - Torpo manual (1)	8 121,00	CDP A200 - pe cadr prb trb	320,000	06/12/00 10:19	08/12/00 09:35

Data: quinta-feira, 7 de dezembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
1 XD001 - Fixadeira plana (	8 139,00	MST A410 - tampo prb red esp	10,000	07/12/00 11:01	07/12/00 12:14
	8 135,00	MST A120 - tampo prb quad fin	10,000	07/12/00 12:14	07/12/00 13:18
	8 134,00	MST A110 - tampo prb quad fin	10,000	07/12/00 13:18	07/12/00 14:22
	8 138,00	MST A210 - tampo prb red fin	30,000	07/12/00 14:22	07/12/00 17:26
	8 131,00	MSP A210 - pe ms prb red	120,000	07/12/00 17:26	08/12/00 13:39
MIS A001 - Mesa op (3)	8 100,00	CDS A140 - cadr S A prb, pe trb, est trb	60,000	07/12/00 16:31	11/12/00 11:56

Programação de produção por data

Data: sexta-feira, 8 de dezembro de 2000

Recurso de Trabalho	Ordem de produçã	Item	Quantidade	Início	Conclusão
LND001 - Lixadeira plana (	8 141,00	MISV A120 - trv ms pnh quad art	50,000	08 12 00 13:39	08 12 00 14:52
	8 143,00	MISV A220 - trv ms pnh quad lng	50,000	08 12 00 14:52	08 12 00 16:26
	8 136,00	MIST A120 - tampo pnh quad fin	25,000	08 12 00 16:26	09 12 00 09:09
MISA001 - Mesa op (2)	8 103,00	MISS A220 - mesa S A pnh red fin, pe pnh red, verniz	30,000	08 12 00 10:31	09 12 00 09:08

Data: sábado, 9 de dezembro de 2000

Recurso de Trabalho	Ordem de produçã	Item	Quantidade	Início	Conclusão
LND001 - Lixadeira plana (	8 142,00	MISV A120 - trv ms pnh quad art	30,000	09 12 00 09:09	09 12 00 09:58
	8 144,00	MISV A220 - trv ms pnh quad lng	30,000	09 12 00 09:58	09 12 00 10:56
	8 140,00	MISF A410 - tampo pnh red esp	15,000	09 12 00 10:56	11 12 00 08:39
MISA001 - Mesa op (1)	8 097,00	MIS3410 - mesa pnh quad fin, pe pnh red, verniz	25,000	09 12 00 11:43	11 12 00 14:01
MISA001 - Mesa op (2)	8 105,00	MISS A340 - mesa S A pnh quad fin, pe pnh red, v	25,000	09 12 00 09:08	11 12 00 12:21

Programação de produção por data

Data: segunda-feira, 11 de dezembro de 2000

Recurso de Trabalho	Ordem de produção	Item	Quantidade	Início	Conclusão
FND001 - Lixadeira plana (	8 137,00	MSTA120 - tambo pnb quad fin	15,000	11 12 00 08:39	11 12 00 10:13
	8 132,00	MSPA210 - pe ms pnb red	60,000	11 12 00 10:13	11 12 00 13:26
MIS A001 - Mesa op (1)	8 098,00	MIS6220 - mesa pnb red esp. pe pnb red. bco	15,000	11 12 00 14:01	11 12 00 17:45
	8 092,00	CD1420 - cade pnb. pe tub. est tub. bco	60,000	11 12 00 17:45	13 12 00 08:51
MIS A001 - Mesa op (2)	8 104,00	MIS5A330 - mesa S A pnb quad fin. pe pnb quad.	15,000	11 12 00 12:21	11 12 00 16:46
	8 095,00	MIS2210 - mesa pnb red fin. pe pnb red. verniz	30,000	11 12 00 16:46	12 12 00 14:16
MIS A001 - Mesa op (3)	8 106,00	MIS5A620 - mesa S A pnb red esp. pe pnb red. ver	15,000	11 12 00 11:56	11 12 00 16:21
	8 096,00	MIS3310 - mesa pnb quad fin. pe pnb quad. verniz	15,000	11 12 00 16:21	12 12 00 10:15

Necessidade de disponibilidade de matéria-prima

Item	Data requerida	Prazo	Data de compra	Quantidade
NAC0100 - verniz	30 nov/2000	0	30 nov/2000	30,00
	04 dez/2000	0	04 dez/2000	24,00
	06 dez/2000	0	06 dez/2000	18,00
	09 dez/2000	0	09 dez/2000	18,75
	11 dez/2000	0	11 dez/2000	45,75
NAC0200 - tinta branca	11 dez/2000	0	11 dez/2000	11,25
NFX0100 - cola	27 nov/2000	5	22 nov/2000	25,50
	28 nov/2000	5	23 nov/2000	9,00
	30 nov/2000	5	25 nov/2000	7,50
	02 dez/2000	5	27 nov/2000	6,00
	06 dez/2000	5	01 dez/2000	4,50
	07 dez/2000	5	02 dez/2000	3,00
	08 dez/2000	5	03 dez/2000	4,50
	09 dez/2000	5	04 dez/2000	3,75
	11 dez/2000	5	06 dez/2000	4,50
NRP0200 - ripa pinho, estreita	29 nov/2000	5	24 nov/2000	60,00
	01 dez/2000	5	26 nov/2000	30,00
NRP1200 - ripa pinho, larga	27 nov/2000	5	22 nov/2000	75,00
	30 nov/2000	5	25 nov/2000	45,00
NTB0100 - tabua peroba, estreita e fina	28 nov/2000	5	23 nov/2000	72,00
	30 nov/2000	5	25 nov/2000	36,00
NTB0200 - tabua peroba, larga, fina	27 nov/2000	5	22 nov/2000	80,00
NTB0400 - tabua peroba, larga, espessa	27 nov/2000	5	22 nov/2000	50,00
NTB1100 - tabua pinho, estreita, fina	27 nov/2000	5	22 nov/2000	90,00
	29 nov/2000	5	24 nov/2000	54,00
NTB1200 - tabua pinho, larga, fina	27 nov/2000	5	22 nov/2000	100,00
XVG0100 - viga peroba, quadrada, pequena	29 nov/2000	5	24 nov/2000	180,00
	01 dez/2000	5	26 nov/2000	90,00
XVG0300 - viga peroba, circular, pequena	29 nov/2000	5	24 nov/2000	240,00
	30 nov/2000	5	25 nov/2000	460,00
	01 dez/2000	5	26 nov/2000	290,00
	02 dez/2000	5	27 nov/2000	126,00

Necessidade de disponibilidade de matéria-prima

Item	Data requerida	Prazo	Data de compra	Quantidade
NVG0300 - viga peroba, circular, pequena	04 dez 2000	5	29 nov 2000	63,00
NVG0400 - viga peroba, circular, grande	04 dez 2000	5	29 nov 2000	67,50
NVG1100 - viga pinho, quadrada, pequena	27 nov 2000	5	22 nov 2000	360,00
	02 dez 2000	5	27 nov 2000	60,00
	04 dez 2000	5	29 nov 2000	108,00
NVG1200 - viga pinho, quadrada, grande	02 dez 2000	5	27 nov 2000	135,00
	04 dez 2000	5	29 nov 2000	67,50
NVG1300 - viga pinho, circular, pequena	28 nov 2000	5	23 nov 2000	525,00
	30 nov 2000	5	25 nov 2000	395,00
	01 dez 2000	5	26 nov 2000	160,00
NVG1400 - viga pinho, circular, grande	02 dez 2000	5	27 nov 2000	112,50

ANEXO D - DADOS DO EXEMPLO

As próximas páginas contêm os dados utilizados no exemplo (respectivamente):

- Item
- Operações
- Entrada de operações
- Recurso de Trabalho
- Disponibilidade de Trabalho
- Família
- Centro de Trabalho
- Categoria
- Grandeza
- Unidade
- Regra de Sequenciação
- Lead-time de matérias-primas

Código	Nome	Descrição	Categoria	Família	Grand	Unid	CustInv
CD1110	cadr prb, pe smp, est smp, verniz	Cadeira em peroba, pés simples, estrutura de encosto simples, acabamento em verniz	PA	CADR	un	un	0,63
CD1120	cadr prb, pe smp, est smp, bco	Cadeira em peroba, pés simples, estrutura de encosto simples, acabamento com pintura branca	PA	CADR	un	un	0,63
CD1210	cadr prb, pe trb, est smp, verniz	Cadeira em peroba, pés trabalhados, estrutura de encosto simples, acabamento em verniz	PA	CADR	un	un	0,69
CD1220	cadr prb, pe trb, est smp, bco	Cadeira em peroba, pés trabalhados, estrutura de encosto simples, acabamento com pintura branca	PA	CADR	un	un	0,69
CD1310	cadr prb, pe smp, est trb, verniz	Cadeira em peroba, pés simples, estrutura de encosto trabalhada, acabamento em verniz	PA	CADR	un	un	0,66
CD1320	cadr prb, pe smp, est trb, bco	Cadeira em peroba, pés simples, estrutura de encosto trabalhada, acabamento com pintura branca	PA	CADR	un	un	0,66
CD1410	cadr prb, pe trb, est trb, verniz	Cadeira em peroba, pés trabalhados, estrutura de encosto trabalhada, acabamento em verniz	PA	CADR	un	un	0,71
CD1420	cadr prb, pe trb, est trb, bco	Cadeira em peroba, pés trabalhados, estrutura de encosto trabalhada, acabamento com pintura branca	PA	CADR	un	un	0,71
CD2110	cadr pnh, pe smp, est smp, verniz	Cadeira em pinho, pés simples, estrutura de encosto simples, acabamento em verniz	PA	CADR	un	un	0,55
CD2120	cadr pnh, pe smp, est smp, bco	Cadeira em pinho, pés simples, estrutura de encosto simples, acabamento com pintura branca	PA	CADR	un	un	0,55
CD2210	cadr pnh, pe trb, est smp, verniz	Cadeira em pinho, pés trabalhados, estrutura de encosto simples, acabamento em verniz	PA	CADR	un	un	0,61
CD2220	cadr pnh, pe trb, est smp, bco	Cadeira em pinho, pés trabalhados, estrutura de encosto simples, acabamento com pintura branca	PA	CADR	un	un	0,61
CD2310	cadr pnh, pe smp, est trb, verniz	Cadeira em pinho, pés simples, estrutura de encosto trabalhada, acabamento em verniz	PA	CADR	un	un	0,58
CD2320	cadr pnh, pe smp, est trb, bco	Cadeira em pinho, pés simples, estrutura de encosto trabalhada, acabamento com pintura branca	PA	CADR	un	un	0,58
CD2410	cadr pnh, pe trb, est trb, verniz	Cadeira em pinho, pés trabalhados, estrutura de encosto trabalhada, acabamento em verniz	PA	CADR	un	un	0,63
CD2420	cadr pnh, pe trb, est trb, bco	Cadeira em pinho, pés trabalhados, estrutura de encosto trabalhada, acabamento com pintura branca	PA	CADR	un	un	0,63
CDA100	asento peroba	Assento de cadeira em peroba	SA	CDA	un	un	0,08
CDA110	asento pinho	Assento de cadeira em pinho	SA	CDA	un	un	0,07
CDAN100	asento N/A peroba	Corte para ssento de cadeira em peroba	SA	CDAN	un	un	0,03
CDAN110	asento N/A pinho	Corte para ssento de cadeira em pinho	SA	CDAN	un	un	0,02

Código	Nome	Descrição	Categoria	Família	Grand	Unid	Custolnv
CDAS100	asnto S/A peroba	Assento de cadeira sem acabamento em peroba	SA	CDAS	un	un	0,05
CDAS110	asnto S/A pinho	Assento de cadeira sem acabamento em pinho	SA	CDAS	un	un	0,04
CDEA100	enc cadr prb	Encosto acabado de cadeira em peroba, sem acabamento	SA	CDEA	un	un	0,04
CDEA110	enc cadr pnh	Encosto acabado de cadeira em pinho, sem acabamento	SA	CDEA	un	un	0,05
CDEN100	enc cadr N/A prb	Corte para encosto de cadeira em peroba, sem acabamento	SA	CDEN	un	un	0,01
CDEN110	enc cadr N/A pnh	Corte para encosto de cadeira em pinho, sem acabamento	SA	CDEN	un	un	0,02
CDES100	enc cadr S/A prb	Encosto sem acabamento de cadeira em peroba (já com curvatura ergonômica)	SA	CDES	un	un	0,01
CDES110	enc cadr S/A pnh	Encosto sem acabamento de cadeira em pinho (já com curvatura ergonômica)	SA	CDES	un	un	0,02
CDPA100	pe cadr prb smp	Pé de cadeira acabado, em peroba, simples	SA	CDPA	un	un	0,03
CDPA110	pe cadr pnh smp	Pé de cadeira acabado, em pinho, simples	SA	CDPA	un	un	0,02
CDPA200	pe cadr prb trb	Pé de cadeira acabado, em peroba, trabalhado	SA	CDPA	un	un	0,05
CDPA210	pe cadr pnh trb	Pé de cadeira acabado, em pinho, trabalhado	SA	CDPA	un	un	0,04
CDPN100	pe cadr N/A prb smp	Corte para pé de cadeira sem acabamento, em peroba	SA	CDPN	un	un	0,02
CDPN110	pe cadr N/A pnh smp	Corte para pé de cadeira sem acabamento, em pinho	SA	CDPN	un	un	0,01
CDSA110	cadr S/A prb, pe smp, est smp	Cadeira sem acabamento em peroba, pés simples, estrutura de encosto simples	SA	CDSA	un	un	0,51
CDSA120	cadr S/A prb, pe trb, est smp	Cadeira sem acabamento em peroba, pés trabalhados, estrutura de encosto simples	SA	CDSA	un	un	0,57
CDSA130	cadr S/A prb, pe smp, est trb	Cadeira sem acabamento em peroba, pés simples, estrutura de encosto trabalhada	SA	CDSA	un	un	0,54
CDSA140	cadr S/A prb, pe trb, est trb	Cadeira sem acabamento em peroba, pés trabalhados, estrutura de encosto trabalhada	SA	CDSA	un	un	0,60
CDSA210	cadr S/A pnh, pe smp, est smp	Cadeira sem acabamento em pinho, pés simples, estrutura de encosto simples	SA	CDSA	un	un	0,43
CDSA220	cadr S/A pnh, pe trb, est smp	Cadeira sem acabamento em pinho, pés trabalhados, estrutura de encosto simples	SA	CDSA	un	un	0,49
CDSA230	cadr S/A pnh, pe smp, est trb	Cadeira sem acabamento em pinho, pés simples, estrutura de encosto trabalhada	SA	CDSA	un	un	0,46
CDSA240	cadr S/A pnh, pe trb, est trb	Cadeira sem acabamento em pinho, pés trabalhados, estrutura de encosto trabalhada	SA	CDSA	un	un	0,52
CDTA100	est enc cadr prb smp	Estrutura acabada para encosto de cadeira, em peroba, simples	SA	CDTA	un	un	0,04

Código	Nome	Descrição	Categoria	Família	Grand	Unid	Custolnv
CDTA110	est enc cadr pnh smp	Estrutura acabada para encosto de cadeira, em pinho, simples	SA	CDTA	un	un	0,03
CDTA200	est enc cadr prb trb	Estrutura acabada para encosto de cadeira, em peroba, trabalhado	SA	CDTA	un	un	0,06
CDTA210	est enc cadr pnh trb	Estrutura acabada para encosto de cadeira, em pinho, trabalhado	SA	CDTA	un	un	0,04
CDTN100	est enc cadr N/A prb smp	Corte para estrutura para encosto de cadeira sem acabamento, em peroba	SA	CDTN	un	un	0,03
CDTN110	est enc cadr N/A pnh smp	Corte para estrutura para encosto de cadeira sem acabamento, em pinho	SA	CDTN	un	un	0,02
CDVA100	trav cadr prb	Travessa acabada para apoio dos pés de cadeira, em peroba	SA	CDVA	un	un	0,02
CDVA110	trav cadr pnh	Travessa acabada para apoio dos pés de cadeira, em pinho	SA	CDVA	un	un	0,02
CDVN100	trav cadr N/A prb	Corte para travessa para apoio dos pés de cadeira, em peroba	SA	CDVN	un	un	0,01
CDVN110	trav cadr N/A pnh	Corte para travessa para apoio dos pés de cadeira, em pinho	SA	CDVN	un	un	0,01
MS1110	mesa prb quad fin, pe prb quad, verniz	Mesa com tampo fino quadrado em peroba, pé quadrado em peroba, acabamento em verniz	PA	MESA	un	un	1,12
MS1120	mesa prb quad fin, pe prb quad, bco	Mesa com tampo fino quadrado em peroba, pé quadrado em peroba, acabamento com pintura branca	PA	MESA	un	un	1,12
MS1210	mesa prb quad fin, pe prb red, verniz	Mesa com tampo fino quadrado em peroba, pé redondo em peroba, acabamento em verniz	PA	MESA	un	un	1,03
MS1220	mesa prb quad fin, pe prb red, bco	Mesa com tampo fino quadrado em peroba, pé redondo em peroba, acabamento com pintura branca	PA	MESA	un	un	1,03
MS2110	mesa prb red fin, pe prb quad, verniz	Mesa com tampo fino redondo em peroba, pé quadrado em peroba, acabamento em verniz	PA	MESA	un	un	1,22
MS2120	mesa prb red fin, pe prb quad, bco	Mesa com tampo fino redondo em peroba, pé quadrado em peroba, acabamento com pintura branca	PA	MESA	un	un	1,22
MS2210	mesa prb red fin, pe prb red, verniz	Mesa com tampo fino redondo em peroba, pé redondo em peroba, acabamento em verniz	PA	MESA	un	un	1,13
MS2220	mesa prb red fin, pe prb red, bco	Mesa com tampo fino redondo em peroba, pé redondo em peroba, acabamento com pintura branca	PA	MESA	un	un	1,13
MS3310	mesa pnh quad fin, pe pnh quad, verniz	Mesa com tampo fino quadrado em pinho, pé quadrado em pinho, acabamento em verniz	PA	MESA	un	un	1,16

Código	Nome	Descrição	Categoria	Família	Grand	Unid	Custolnv
MS3320	mesa pnh quad fin, pe pnh quad, bco	Mesa com tampo fino quadrado em pinho, pé quadrado em pinho, acabamento com pintura branca	PA	MESA	un	un	1,16
MS3410	mesa pnh quad fin, pe pnh red, verniz	Mesa com tampo fino quadrado em pinho, pé redondo em pinho, acabamento em verniz	PA	MESA	un	un	1,03
MS3420	mesa pnh quad fin, pe pnh red, bco	Mesa com tampo fino quadrado em pinho, pé redondo em pinho, acabamento com pintura branca	PA	MESA	un	un	1,03
MS4310	mesa pnh red fin, pe pnh quad, verniz	Mesa com tampo fino redondo em pinho, pé quadrado em pinho, acabamento em verniz	PA	MESA	un	un	1,19
MS4320	mesa pnh red fin, pe pnh quad, bco	Mesa com tampo fino redondo em pinho, pé quadrado em pinho, acabamento com pintura branca	PA	MESA	un	un	1,19
MS4410	mesa pnh red fin, pe pnh red, verniz	Mesa com tampo fino redondo em pinho, pé redondo em pinho, acabamento em verniz	PA	MESA	un	un	1,06
MS4420	mesa pnh red fin, pe pnh red, bco	Mesa com tampo fino redondo em pinho, pé redondo em pinho, acabamento com pintura branca	PA	MESA	un	un	1,06
MS5110	mesa prb quad esp, pe prb quad, verniz	Mesa com tampo espesso quadrado em peroba, pé quadrado em peroba, acabamento em verniz	PA	MESA	un	un	1,28
MS5120	mesa prb quad esp, pe prb quad, bco	Mesa com tampo espesso quadrado em peroba, pé quadrado em peroba, acabamento com pintura branca	PA	MESA	un	un	1,28
MS5210	mesa prb quad esp, pe prb red, verniz	Mesa com tampo espesso quadrado em peroba, pé redondo em peroba, acabamento em verniz	PA	MESA	un	un	1,19
MS5220	mesa prb quad esp, pe prb red, bco	Mesa com tampo espesso quadrado em peroba, pé redondo em peroba, acabamento com pintura branca	PA	MESA	un	un	1,19
MS6110	mesa prb red esp, pe prb quad, verniz	Mesa com tampo espesso redondo em peroba, pé quadrado em peroba, acabamento em verniz	PA	MESA	un	un	1,38
MS6120	mesa prb red esp, pe prb quad, bco	Mesa com tampo espesso redondo em peroba, pé quadrado em peroba, acabamento com pintura branca	PA	MESA	un	un	1,38
MS6210	mesa prb red esp, pe prb red, verniz	Mesa com tampo espesso redondo em peroba, pé redondo em peroba, acabamento em verniz	PA	MESA	un	un	1,29
MS6220	mesa prb red esp, pe prb red, bco	Mesa com tampo espesso redondo em peroba, pé redondo em peroba, acabamento com pintura branca	PA	MESA	un	un	1,29
MS7310	mesa pnh quad esp, pe pnh quad, verniz	Mesa com tampo espesso quadrado em pinho, pé quadrado em pinho, acabamento em verniz	PA	MESA	un	un	1,25

Código	Nome	Descrição	Categoria	Família	Grand	Unid	Custolnv
MS7320	mesa pnh quad esp, pe pnh quad, bco	Mesa com tampo espesso quadrado em pinho, pé quadrado em pinho, acabamento com pintura branca	PA	MESA	un	un	1,25
MS7410	mesa pnh quad esp, pe pnh red, verniz	Mesa com tampo espesso quadrado em pinho, pé redondo em pinho, acabamento em verniz	PA	MESA	un	un	1,13
MS7420	mesa pnh quad esp, pe pnh red, bco	Mesa com tampo espesso quadrado em pinho, pé redondo em pinho, acabamento com pintura branca	PA	MESA	un	un	1,13
MS8310	mesa pnh red esp, pe pnh quad, verniz	Mesa com tampo espesso redondo em pinho, pé quadrado em pinho, acabamento em verniz	PA	MESA	un	un	1,28
MS8320	mesa pnh red esp, pe pnh quad, bco	Mesa com tampo espesso redondo em pinho, pé quadrado em pinho, acabamento com pintura branca	PA	MESA	un	un	1,28
MS8410	mesa pnh red esp, pe pnh red, verniz	Mesa com tampo espesso redondo em pinho, pé redondo em pinho, acabamento em verniz	PA	MESA	un	un	1,16
MS8420	mesa pnh red esp, pe pnh red, bco	Mesa com tampo espesso redondo em pinho, pé redondo em pinho, acabamento com pintura branca	PA	MESA	un	un	1,16
MSPA110	pe ms prb quad	Pé de mesa quadrado em peroba	SA	MSPA	un	un	0,07
MSPA120	pe ms pnh quad	Pé de mesa quadrado em pinho	SA	MSPA	un	un	0,10
MSPA210	pe ms prb red	Pé de mesa redondo em peroba	SA	MSPA	un	un	0,04
MSPA220	pe ms pnh red	Pé de mesa redondo em pinho	SA	MSPA	un	un	0,06
MSPN110	pe N/A ms prb quad	Pé de mesa sem acabamento quadrado em peroba	SA	MSPN	un	un	0,05
MSPN120	pe N/A ms pnh quad	Pé de mesa sem acabamento quadrado em pinho	SA	MSPN	un	un	0,07
MSPN210	pe N/A ms prb red	Pé de mesa sem acabamento redondo em peroba	SA	MSPN	un	un	0,03
MSPN220	pe N/A ms pnh red	Pé de mesa sem acabamento redondo em pinho	SA	MSPN	un	un	0,04
MSSA110	mesa S/A prb quad fin, pe prb quad, vnz	Mesa sem acabamento com tampo fino quadrado em peroba, pé quadrado em peroba, acabamento em verniz	SA	MSSA	un	un	0,81
MSSA120	mesa S/A prb quad fin, pe prb red, vnz	Mesa sem acabamento com tampo fino quadrado em peroba, pé redondo em peroba, acabamento em verniz	SA	MSSA	un	un	0,72
MSSA210	mesa S/A prb red fin, pe prb quad, vnz	Mesa sem acabamento com tampo fino redondo em peroba, pé quadrado em peroba, acabamento em verniz	SA	MSSA	un	un	0,91
MSSA220	mesa S/A prb red fin, pe prb red, verniz	Mesa sem acabamento com tampo fino redondo em peroba, pé redondo em peroba, acabamento em verniz	SA	MSSA	un	un	0,82
MSSA330	mesa S/A pnh quad fin, pe pnh quad, vnz	Mesa sem acabamento com tampo fino quadrado em pinho, pé quadrado em pinho, acabamento em verniz	SA	MSSA	un	un	0,84

Código	Nome	Descrição		Categoria	Família	Grand	Unid	CustotInv
MSSA340	mesa S/A pnh quad fin, pe pnh red, vnz	Mesa sem acabamento com tampo fino quadrado em pinho, pé redondo em pinho, acabamento em verniz	SA	MSSA	un	un		0,72
MSSA430	mesa S/A pnh red fin, pe pnh quad, vnz	Mesa sem acabamento com tampo fino redondo em pinho, pé quadrado em pinho, acabamento em verniz	SA	MSSA	un	un		0,88
MSSA440	mesa S/A pnh red fin, pe pnh red, verniz	Mesa sem acabamento com tampo fino redondo em pinho, pé redondo em pinho, acabamento em verniz	SA	MSSA	un	un		0,75
MSSA510	mesa S/A prb quad esp, pe prb quad, vnz	Mesa sem acabamento com tampo espesso quadrado em peroba, pé quadrado em peroba, acabamento em verniz	SA	MSSA	un	un		0,97
MSSA520	mesa S/A prb quad esp, pe prb red, vnz	Mesa sem acabamento com tampo espesso quadrado em peroba, pé redondo em peroba, acabamento em verniz	SA	MSSA	un	un		0,88
MSSA610	mesa S/A prb red esp, pe prb quad, vnz	Mesa sem acabamento com tampo espesso redondo em peroba, pé quadrado em peroba, acabamento em verniz	SA	MSSA	un	un		1,07
MSSA620	mesa S/A prb red esp, pe prb red, verniz	Mesa sem acabamento com tampo espesso redondo em peroba, pé redondo em peroba, acabamento em verniz	SA	MSSA	un	un		0,98
MSSA730	mesa S/A pnh quad esp, pe pnh quad, vnz	Mesa sem acabamento com tampo espesso quadrado em pinho, pé quadrado em pinho, acabamento em verniz	SA	MSSA	un	un		0,94
MSSA740	mesa S/A pnh quad esp, pe pnh red, vnz	Mesa sem acabamento com tampo espesso quadrado em pinho, pé redondo em pinho, acabamento em verniz	SA	MSSA	un	un		0,81
MSSA830	mesa S/A pnh red esp, pe pnh quad, vnz	Mesa sem acabamento com tampo espesso redondo em pinho, pé quadrado em pinho, acabamento em verniz	SA	MSSA	un	un		0,97
MSSA840	mesa S/A pnh red esp, pe pnh red, verniz	Mesa sem acabamento com tampo espesso redondo em pinho, pé redondo em pinho, acabamento em verniz	SA	MSSA	un	un		0,85
MSTA110	tampo prb quad fin	Tampo acabado de mesa, quadrado em peroba, fino	SA	MSTA	un	un		0,32
MSTA120	tampo pnh quad fin	Tampo acabado de mesa, quadrado em pinho, fino	SA	MSTA	un	un		0,25
MSTA210	tampo prb red fin	Tampo acabado de mesa, redondo em peroba, fino	SA	MSTA	un	un		0,32
MSTA220	tampo pnh red fin	Tampo acabado de mesa, redondo em pinho, fino	SA	MSTA	un	un		0,25
MSTA310	tampo prb quad esp	Tampo acabado de mesa, quadrado em peroba, espesso	SA	MSTA	un	un		0,48
MSTA320	tampo pnh quad esp	Tampo acabado de mesa, quadrado em pinho, espesso	SA	MSTA	un	un		0,35

Código	Nome	Descrição	Categoria	Família	Grand	Unid	CustotInv
MSTA410	tampo prb red esp	Tampo acabado de mesa, redondo em peroba, espesso	SA	MSTA	un	un	0,48
MSTA420	tampo pnh red esp	Tampo acabado de mesa, redondo em pinho, espesso	SA	MSTA	un	un	0,35
MSTN110	tampo N/A prb quad fin	Corte para tampo de mesa, quadrado em peroba, fino	SA	MSTN	un	un	0,17
MSTN120	tampo N/A pnh quad fin	Corte para tampo de mesa, quadrado em pinho, fino	SA	MSTN	un	un	0,11
MSTN210	tampo N/A prb red fin	Corte para tampo de mesa, redondo em peroba, fino	SA	MSTN	un	un	0,17
MSTN220	tampo N/A pnh red fin	Corte para tampo de mesa, redondo em pinho, fino	SA	MSTN	un	un	0,11
MSTN310	tampo N/A prb quad esp	Corte para tampo de mesa, quadrado em peroba, espesso	SA	MSTN	un	un	0,33
MSTN320	tampo N/A pnh quad esp	Corte para tampo de mesa, quadrado em pinho, espesso	SA	MSTN	un	un	0,20
MSTN410	tampo N/A prb red esp	Corte para tampo de mesa, redondo em peroba, espesso	SA	MSTN	un	un	0,33
MSTN420	tampo N/A pnh red esp	Corte para tampo de mesa, redondo em pinho, espesso	SA	MSTN	un	un	0,20
MSTS110	tampo S/A prb quad fin	Tampo sem acabamento de mesa, quadrado em peroba, fino	SA	MSTS	un	un	0,28
MSTS120	tampo S/A pnh quad fin	Tampo sem acabamento de mesa, quadrado em pinho, fino	SA	MSTS	un	un	0,21
MSTS210	tampo S/A prb red fin	Tampo sem acabamento de mesa, redondo em peroba, fino	SA	MSTS	un	un	0,28
MSTS220	tampo S/A pnh red fin	Tampo sem acabamento de mesa, redondo em pinho, fino	SA	MSTS	un	un	0,21
MSTS310	tampo S/A prb quad esp	Tampo sem acabamento de mesa, quadrado em peroba, espesso	SA	MSTS	un	un	0,44
MSTS320	tampo S/A pnh quad esp	Tampo sem acabamento de mesa, quadrado em pinho, espesso	SA	MSTS	un	un	0,31
MSTS410	tampo S/A prb red esp	Tampo sem acabamento de mesa, redondo em peroba, espesso	SA	MSTS	un	un	0,44
MSTS420	tampo S/A pnh red esp	Tampo sem acabamento de mesa, redondo em pinho, espesso	SA	MSTS	un	un	0,31
MSVA110	trv ms prb quad crt	Travessa acabada de mesa, quadrada curta, em peroba	SA	MSVA	un	un	0,03
MSVA120	trv ms pnh quad crt	Travessa acabada de mesa, quadrada curta, em pinho	SA	MSVA	un	un	0,02
MSVA210	trv ms prb quad lng	Travessa acabada de mesa, quadrada longa, em peroba	SA	MSVA	un	un	0,04

Código	Nome	Descrição	Categoria	Família	Grand	Unid	Custolnv
MSVA220	tv ms pnh quad lng	Travessa acabada de mesa, quadrada longa, em pinho	SA	MSVA	un	un	0,03
MSVA310	tv ms prb red	Travessa acabada de mesa, seção circular, em peroba	SA	MSVA	un	un	0,06
MSVA320	tv ms pnh red	Travessa acabada de mesa, seção circular, em pinho	SA	MSVA	un	un	0,04
MSVN110	tv ms N/A prb quad crt	Travessa de mesa sem acabamento, quadrada curta, em peroba	SA	MSVN	un	un	0,02
MSVN120	tv ms N/A pnh quad crt	Travessa de mesa sem acabamento, quadrada curta, em pinho	SA	MSVN	un	un	0,02
MSVN210	tv ms N/A prb quad lng	Travessa de mesa sem acabamento, quadrada longa, em peroba	SA	MSVN	un	un	0,03
MSVN220	tv ms N/A pnh quad lng	Travessa de mesa sem acabamento, quadrada longa, em pinho	SA	MSVN	un	un	0,02
MSVN310	tv ms N/A prb red	Travessa de mesa sem acabamento, seção circular, em peroba	SA	MSVN	un	un	0,04
MSVN320	tv ms N/A pnh red	Travessa de mesa sem acabamento, seção circular, em pinho	SA	MSVN	un	un	0,02
XAC0100	verniz	Verniz incolor	MP	MPAC	m	kg	0,32
XAC0200	tinta branca	Tinta branca impermeável	MP	MPAC	m	kg	0,32
XFX0100	cola	Cola para madeira	MP	MPEX	m	kg	0,12
XF0210	parafuso pequeno	Parafuso pequeno, incluindo porca correspondente	MP	MPEX	un	un	0,00
		Dimensão: M2x2					
XF0220	parafuso grande	Parafuso grande, incluindo porca correspondente	MP	MPEX	un	un	0,00
		Dimensão: M6x6					
XF0310	prego pequeno	Prego pequeno	MP	MPEX	un	un	0,00
		Comprimento: 20 mm					
XF0320	prego grande	Prego grande	MP	MPEX	un	un	0,00
		Comprimento: 60 mm					
XRP0100	ripa peroba, estreita	Ripa de peroba, estreita	MP	MPRP	comp	m	0,02
		Largura: 30 mm					
XRP0200	ripa pinho, estreita	Ripa de pinho, larga	MP	MPRP	comp	m	0,01
		Largura: 70 mm					
XRP1100	ripa peroba, larga	Ripa de peroba, larga	MP	MPRP	comp	m	0,04
		Largura: 70 mm					
XRP1200	ripa pinho, larga	Ripa de pinho, larga	MP	MPRP	comp	m	0,02
		Largura: 70 mm					
XTB0100	tabua peroba, estreita e fina	Tábua de peroba, estreita e fina	MP	MPTB	comp	m	0,04
		Espessura: 15 mm					
		Largura: 150 mm					

ANEXO D - Cadastro de itens

Código	Nome	Descrição	Categoria	Família	Grand	Unid	Custolnv
XTB0200	tabua peroba, larga, fina	Tábua de peroba, larga e fina Espessura: 15 mm Largura: 400 mm	MP	MPTB	comp	m	0,08
XTB0300	tabua peroba, estreita, espessa	Tábua de peroba, estreita e espessa Espessura: 30 mm Largura: 150 mm	MP	MPTB	comp	m	0,08
XTB0400	tabua peroba, larga, espessa	Tábua de peroba, larga e espessa Espessura: 30 mm Largura: 400 mm	MP	MPTB	comp	m	0,16
XTB1100	tabua pinho, estreita, fina	Tábua de pinho, estreita e fina Espessura: 15 mm Largura: 150 mm	MP	MPTB	comp	m	0,02
XTB1200	tabua pinho, larga, fina	Tábua de pinho, larga e fina Espessura: 15 mm Largura: 400 mm	MP	MPTB	comp	m	0,05
XTB1300	tabua pinho, estreita, espessa	Tábua de pinho, estreita e espessa Espessura: 30 mm Largura: 150 mm	MP	MPTB	comp	m	0,05
XTB1400	tabua pinho, larga, espessa	Tábua de pinho, larga e espessa Espessura: 30 mm Largura: 400 mm	MP	MPTB	comp	m	0,10
XVG0100	viga peroba, quadrada, pequena	Viga de peroba, seção retangular, pequena Dimensões: 25 x 15	MP	MPVG	comp	m	0,02
XVG0200	viga peroba, quadrada, grande	Viga de peroba, seção retangular, grande Dimensões: 50 x 35 mm	MP	MPVG	comp	m	0,04
XVG0300	viga peroba, circular, pequena	Viga de peroba, seção circular, pequena Diâmetro: 20 mm	MP	MPVG	comp	m	0,04
XVG0400	viga peroba, circular, grande	Viga de peroba, seção circular, grande Diâmetro: 40	MP	MPVG	comp	m	0,06
XVG1100	viga pinho, quadrada, pequena	Viga de pinho, seção retangular, pequena Dimensões: 25 x 15	MP	MPVG	comp	m	0,02
XVG1200	viga pinho, quadrada, grande	Viga de pinho, seção retangular, grande Dimensões: 50 x 35 mm	MP	MPVG	comp	m	0,02
XVG1300	viga pinho, circular, pequena	Viga de pinho, seção circular, pequena Diâmetro: 20 mm	MP	MPVG	comp	m	0,02
XVG1400	viga pinho, circular, grande	Viga de pinho, seção circular, grande Diâmetro: 40	MP	MPVG	comp	m	0,03

Código da Operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo item):	Set-Up (mesma família):
OPCD1110	Acab final - CD1110	CD1110	Dar acabamento final ao item CD1110 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1120	Acab final - CD1120	CD1120	cad. prb. pe smp. est smp. verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1210	Acab final - CD1210	CD1210	cad. prb. pe smp. est smp. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1220	Acab final - CD1220	CD1220	Dar acabamento final ao item CD1220 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1310	Acab final - CD1310	CD1310	cad. prb. pe trb. est smp. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1320	Acab final - CD1320	CD1320	Dar acabamento final ao item CD1320 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1410	Acab final - CD1410	CD1410	cad. prb. pe smp. est trb. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD1420	Acab final - CD1420	CD1420	Dar acabamento final ao item CD1420 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2110	Acab final - CD2110	CD2110	cad. prb. pe trb. est trb. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2120	Acab final - CD2120	CD2120	Dar acabamento final ao item CD2120 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2210	Acab final - CD2210	CD2210	cad. prb. pe smp. est smp. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2220	Acab final - CD2220	CD2220	Dar acabamento final ao item CD2220 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2310	Acab final - CD2310	CD2310	cad. prb. pe trb. est smp. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2320	Acab final - CD2320	CD2320	Dar acabamento final ao item CD2320 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2410	Acab final - CD2410	CD2410	cad. prb. pe smp. est trb. bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPCD2420	Acab final - CD2420	CD2420	Dar acabamento final ao item CD2420 - verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDAA100	Acab asnto peroba	CDA100	cad. prb. pe trb. est trb. bco	LXD001	0,00	0,00	0,00	0,00	0,00
OPCDAA110	Acab asnto pinho	CDA110	asnto peroba	LXD001	0,00	0,00	0,00	0,00	0,00
OPCDAN100	Cort asnto N/A peroba	CDAN100	Dar acabamento ao item CDA110 - asnto pinho	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDAN110	Cort asnto N/A pinho	CDAN110	Corte de matéria-prima para o item CDAN100 - asnto N/A peroba	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDAS100	Form asnto S/A peroba	CDAS100	Corte de matéria-prima para o item CDAN110 - asnto N/A pinho	SMSA001	0,00	0,00	0,00	0,00	0,00
OPCDAS110	Form asnto S/A pinho	CDAS110	Dar formato ao item CDAS100 - asnto S peroba	SMSA001	0,00	0,00	0,00	0,00	0,00
OPCDEA100	Acab enc cad. prb	CDEA100	Dar formato ao item CDAS110 - asnto S pinho	en: LXD001	0,00	0,00	0,00	0,00	0,00
OPCDEA110	Acab enc cad. prb	CDEA110	Dar acabamento ao item CDEA100 - en: LXD001	en: LXD001	0,00	0,00	0,00	0,00	0,00

Código da operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo item):	Set-Up (mesma família):
OPCDEN100	Cort enc cadr N/A prb	C DEN100	Corte de matéria-prima para o item CDEN100 - enc cadr N/A prb	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDEN110	Cort enc cadr N/A pnh	C DEN110	Corte de matéria-prima para o item CDEN110 - enc cadr N/A pnh	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDES100	Form enc cadr S/A prb	C DES100	Dar formato ao item CDES100 - enc cac FRM001 S/A prb	FRM001	0,00	50,00	0,00	0,00	0,00
OPCDES110	Form enc cadr S/A pnh	C DES110	Dar formato ao item CDES110 - enc cac FRM001 S/A pnh	FRM001	0,00	50,00	0,00	0,00	0,00
OPCDPA100	Acab pe cadr prb smp	C DPA100	Dar acabamento ao item CDPA100 - pe TRN002 cadr prb smp	TRN002	0,00	0,00	0,00	0,00	0,00
OPCDPA110	Acab pe cadr pnh smp	C DPA110	Dar acabamento ao item CDPA110 - pe TRN002 cadr pnh smp	TRN002	0,00	0,00	0,00	0,00	0,00
OPCDPA200	Acab pe cadr prb trb	C DPA200	Dar acabamento ao item CDPA200 - pe TRN001 cadr prb trb	TRN001	0,00	0,00	0,00	0,00	0,00
OPCDPA210	Acab pe cadr pnh trb	C DPA210	Dar acabamento ao item CDPA210 - pe TRN001 cadr pnh trb	TRN001	0,00	0,00	0,00	0,00	0,00
OPCDPN100	Cort pe cadr N/A prb smp	C DPN100	Corte de matéria-prima para o item CDPN100 - pe cadr N/A prb smp	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDPN110	Cort pe cadr N/A pnh smp	C DPN110	Corte de matéria-prima para o item CDPN110 - pe cadr N/A pnh smp	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDSA110	Montar cadr S/A prb, pe smp, est smp	C DSA110	Montagem do item CDSA110 - cadr S/A prb, pe smp, est smp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA120	Montar cadr S/A prb, pe trb, est smp	C DSA120	Montagem do item CDSA120 - cadr S/A prb, pe trb, est smp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA130	Montar cadr S/A prb, pe smp, est trb	C DSA130	Montagem do item CDSA130 - cadr S/A prb, pe smp, est trb (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA140	Montar cadr S/A prb, pe trb, est trb	C DSA140	Montagem do item CDSA140 - cadr S/A prb, pe trb, est trb (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA210	Montar cadr S/A pnh, pe smp, est smp	C DSA210	Montagem do item CDSA210 - cadr S/A pnh, pe smp, est smp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA220	Montar cadr S/A pnh, pe trb, est smp	C DSA220	Montagem do item CDSA220 - cadr S/A pnh, pe trb, est smp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA230	Montar cadr S/A pnh, pe smp, est trb	C DSA230	Montagem do item CDSA230 - cadr S/A pnh, pe smp, est trb (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDSA240	Montar cadr S/A pnh, pe trb, est trb	C DSA240	Montagem do item CDSA240 - cadr S/A pnh, pe trb, est trb (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPCDTA100	Acab est enc cadr prb smp	C DTA100	Dar acabamento ao item CDTA100 - est TRN002 enc cadr prb smp	TRN002	0,00	0,00	0,00	0,00	0,00
OPCDTA110	Acab est enc cadr pnh smp	C DTA110	Dar acabamento ao item CDTA110 - est TRN002 enc cadr pnh smp	TRN002	0,00	0,00	0,00	0,00	0,00
OPCDTA200	Acab est enc cadr prb trb	C DTA200	Dar acabamento ao item CDTA200 - est TRN001 enc cadr prb trb	TRN001	0,00	0,00	0,00	0,00	0,00
OPCDTA210	Acab est enc cadr pnh trb	C DTA210	Dar acabamento ao item CDTA210 - est TRN001 enc cadr pnh trb	TRN001	0,00	0,00	0,00	0,00	0,00
OPCDTN100	Cort est enc cadr N/A prb smp	C DTN100	Corte de matéria-prima para o item CDTN100 - est enc cadr N/A prb smp	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDTN110	Cort est enc cadr N/A pnh smp	C DTN110	Corte de matéria-prima para o item CDTN110 - est enc cadr N/A pnh smp	SRA001	0,00	0,00	0,00	0,00	0,00

Código da operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo item):	Set-Up (mesma família):
OPCDVA100	Acab trav cadr prb	CDVA100	Dar acabamento ao item CDVA100 - tra	LXD001	0,00	0,00	0,00	0,00	0,00
OPCDVA110	Acab trav cadr pnh	CDVA110	cadr prb	LXD001	0,00	0,00	0,00	0,00	0,00
OPCDVN100	Cort trav cadr N/A prb	CDVN100	cadr pnh	SRA001	0,00	0,00	0,00	0,00	0,00
OPCDVN110	Cort trav cadr N/A pnh	CDVN110	Corte de matéria-prima para o item CDVN100 - trav cadr N/A prb	SRA001	0,00	0,00	0,00	0,00	0,00
OPMS1110	Acab final - MS1110	MS1110	Corte de matéria-prima para o item CDVN110 - trav cadr N/A pnh	SRA001	0,00	0,00	0,00	0,00	0,00
OPMS1120	Acab final - MS1120	MS1120	Dar acabamento final ao item MS1110 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS1210	Acab final - MS1210	MS1210	mesa prb quad fin, pe prb quad, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS1220	Acab final - MS1220	MS1220	Dar acabamento final ao item MS1120 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS2110	Acab final - MS2110	MS2110	mesa prb quad fin, pe prb quad, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS2120	Acab final - MS2120	MS2120	Dar acabamento final ao item MS1210 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS2210	Acab final - MS2210	MS2210	mesa prb quad fin, pe prb red, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS2220	Acab final - MS2220	MS2220	Dar acabamento final ao item MS2110 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS3310	Acab final - MS3310	MS3310	mesa prb quad fin, pe prb red, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS3320	Acab final - MS3320	MS3320	Dar acabamento final ao item MS2120 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS3410	Acab final - MS3410	MS3410	mesa pnh quad fin, pe pnh quad, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS3420	Acab final - MS3420	MS3420	Dar acabamento final ao item MS3310 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS4310	Acab final - MS4310	MS4310	mesa pnh quad fin, pe pnh red, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS4320	Acab final - MS4320	MS4320	Dar acabamento final ao item MS3410 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS4410	Acab final - MS4410	MS4410	mesa pnh quad fin, pe pnh quad, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS4420	Acab final - MS4420	MS4420	Dar acabamento final ao item MS4310 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS5110	Acab final - MS5110	MS5110	mesa pnh red fin, pe pnh red, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS5120	Acab final - MS5120	MS5120	Dar acabamento final ao item MS4410 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS5210	Acab final - MS5210	MS5210	mesa prb quad esp, pe prb quad, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS5220	Acab final - MS5220	MS5220	Dar acabamento final ao item MS5110 - MSA001	MSA001	0,00	0,00	0,00	0,00	0,00
			mesa prb quad esp, pe prb red, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
			mesa prb quad esp, pe prb red, bco	MSA001	0,00	0,00	0,00	0,00	0,00

Código da operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo item):	Set-Up (mesma família):
OPMS6110	Acab final - MS6110	MS6110	Dar acabamento final ao item MS6110 - MSA001 mesa prb red esp, pe prb quad, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS6120	Acab final - MS6120	MS6120	Dar acabamento final ao item MS6120 - MSA001 mesa prb red esp, pe prb quad, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS6210	Acab final - MS6210	MS6210	Dar acabamento final ao item MS6210 - MSA001 mesa prb red esp, pe prb red, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS6220	Acab final - MS6220	MS6220	Dar acabamento final ao item MS6220 - MSA001 mesa prb red esp, pe prb red, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS7310	Acab final - MS7310	MS7310	Dar acabamento final ao item MS7310 - MSA001 mesa prb quad esp, pe prb quad, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS7320	Acab final - MS7320	MS7320	Dar acabamento final ao item MS7320 - MSA001 mesa prb quad esp, pe prb quad, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS7410	Acab final - MS7410	MS7410	Dar acabamento final ao item MS7410 - MSA001 mesa prb quad esp, pe prb red, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS7420	Acab final - MS7420	MS7420	Dar acabamento final ao item MS7420 - MSA001 mesa prb quad esp, pe prb red, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS8310	Acab final - MS8310	MS8310	Dar acabamento final ao item MS8310 - MSA001 mesa prb red esp, pe prb quad, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS8320	Acab final - MS8320	MS8320	Dar acabamento final ao item MS8320 - MSA001 mesa prb red esp, pe prb quad, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS8410	Acab final - MS8410	MS8410	Dar acabamento final ao item MS8410 - MSA001 mesa prb red esp, pe prb red, verniz	MSA001	0,00	0,00	0,00	0,00	0,00
OPMS8420	Acab final - MS8420	MS8420	Dar acabamento final ao item MS8420 - MSA001 mesa prb red esp, pe prb red, bco	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSPA110	Acab pe ms prb quad	MSPA110	Dar acabamento ao item MSPA110 - pe LX0001 ms prb quad	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSPA120	Acab pe ms prb quad	MSPA120	Dar acabamento ao item MSPA120 - pe LX0002 ms prb quad	LXD002	0,00	0,00	0,00	0,00	0,00
OPMSPA210	Acab pe ms prb red	MSPA210	Dar acabamento ao item MSPA210 - pe LX0001 ms prb red	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSPA220	Acab pe ms prb red	MSPA220	Dar acabamento ao item MSPA220 - pe LX0002 ms prb red	LXD002	0,00	0,00	0,00	0,00	0,00
OPMSPN110	Cort pe N/A ms prb quad	MSPN110	Corte de matéria-prima para o item MSPN110 - pe N/A ms prb quad	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSPN120	Cort pe N/A ms prb quad	MSPN120	Corte de matéria-prima para o item MSPN120 - pe N/A ms prb quad	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSPN210	Cort pe N/A ms prb red	MSPN210	Corte de matéria-prima para o item MSPN210 - pe N/A ms prb red	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSPN220	Cort pe N/A ms prb red	MSPN220	Corte de matéria-prima para o item MSPN220 - pe N/A ms prb red	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSSA110	mont ms S/A prb qd fin, f prb qd, vnz	MSSA110	Montagem do item MSSA110 - mesa S/A MSA001 prb quad fin, pe prb quad, vnz (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA120	mont ms S/A prb qd fin, f prb red, vnz	MSSA120	Montagem do item MSSA120 - mesa S/A MSA001 prb quad fin, pe prb red, vnz (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00

ANEXO D - Operações

Código da operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo item):	Set-Up (mesma família):
OPMSSA210	mont ms S/A prb red fin, MSSA210 pe prb qd, vnz		Monagem do item MSSA210 - mesa prb red fin, pe prb quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA220	mont ms S/A prb red fin, MSSA220 pe prb red, vnz		Monagem do item MSSA220 - mesa prb red fin, pe prb red, verniz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA330	mont ms S/A pnh qd fin, MSSA330 pe pnh qd, vnz		Monagem do item MSSA330 - mesa pnh quad fin, pe pnh quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA340	mont ms S/A pnh qd fin, MSSA340 pe pnh red, vnz		Monagem do item MSSA340 - mesa pnh quad fin, pe pnh red, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA430	mont ms S/A pnh red fin, MSSA430 pe pnh qd, vnz		Monagem do item MSSA430 - mesa pnh red fin, pe pnh quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA440	mont ms S/A pnh red fin, MSSA440 pe pnh red, vnz		Monagem do item MSSA440 - mesa pnh red fin, pe pnh red, verniz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA510	mont ms S/A prb qd esp, MSSA510 pe prb qd, vnz		Monagem do item MSSA510 - mesa prb quad esp, pe prb quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA520	mont ms S/A prb qd esp, MSSA520 pe prb red, vnz		Monagem do item MSSA520 - mesa prb quad esp, pe prb red, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA610	mont ms S/A prb red esp MSSA610 pe prb qd, vnz		Monagem do item MSSA610 - mesa prb red esp, pe prb quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA620	mont ms S/A prb red esp MSSA620 pe prb red, vnz		Monagem do item MSSA620 - mesa prb red esp, pe prb red, verniz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA730	mont ms S/A pnh qd esp, MSSA730 pe pnh qd, vnz		Monagem do item MSSA730 - mesa pnh quad esp, pe pnh quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA740	mont ms S/A pnh qd esp, MSSA740 pe pnh red, vnz		Monagem do item MSSA740 - mesa pnh quad esp, pe pnh red, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA830	mont ms S/A pnh red esp MSSA830 pe pnh qd, vnz		Monagem do item MSSA830 - mesa pnh red esp, pe pnh quad, vnz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSSA840	mont ms S/A pnh red esp MSSA840 pe pnh red, vnz		Monagem do item MSSA840 - mesa pnh red esp, pe pnh red, verniz (sem acabamento)	S/A MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTA110	Acab tampo prb quad fin MSTA110		Dar acabamento ao item MSTA110 - tampo prb quad fin	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTA120	Acab tampo pnh quad fin MSTA120		Dar acabamento ao item MSTA120 - tampo pnh quad fin	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTA210	Acab tampo prb red fin MSTA210		Dar acabamento ao item MSTA210 - tampo prb red fin	LXD001	0,00	0,00	0,00	0,00	0,00

Código da Operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo Item):	Set-Up (mesma família):
OPMSTA220	Acab tampo pnh red fin	MSTA220	Dar acabamento ao item MST220 - tampo pnh red fin	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTA310	Acab tampo prb quad espMSTA310	MSTA310	Dar acabamento ao item MST310 - tampo prb quad esp	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTA320	Acab tampo pnh quad espMSTA320	MSTA320	Dar acabamento ao item MST320 - tampo pnh quad esp	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTA410	Acab tampo prb red esp	MSTA410	Dar acabamento ao item MST410 - tampo prb red esp	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTA420	Acab tampo pnh red esp	MSTA420	Dar acabamento ao item MST420 - tampo pnh red esp	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSTN110	Cort tampo N/A prb quad MSTN110 fin	MSTN110	Corte de matéria-prima para o item MSTN110 - tampo N/A prb quad fin	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN120	Cort tampo N/A pnh quac MSTN120 fin	MSTN120	Corte de matéria-prima para o item MSTN120 - tampo N/A pnh quad fin	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN210	Cort tampo N/A prb red finMSTN210	MSTN210	Corte de matéria-prima para o item MSTN210 - tampo N/A prb red fin	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN220	Cort tampo N/A pnh red	MSTN220	Corte de matéria-prima para o item MSTN220 - tampo N/A pnh red fin	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN310	Cort tampo N/A prb quad MSTN310 esp	MSTN310	Corte de matéria-prima para o item MSTN310 - tampo N/A prb quad esp	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN320	Cort tampo N/A pnh quac MSTN320 esp	MSTN320	Corte de matéria-prima para o item MSTN320 - tampo N/A pnh quad esp	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN410	Cort tampo N/A prb red	MSTN410	Corte de matéria-prima para o item MSTN410 - tampo N/A prb red esp	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTN420	Cort tampo N/A pnh red	MSTN420	Corte de matéria-prima para o item MSTN420 - tampo N/A pnh red esp	SRA002	0,00	0,00	0,00	0,00	0,00
OPMSTS110	Montar tampo S/A prb	MSTS110	Monagem do item MST110 - tampo S//MSA001 prb quad fin (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS120	Montar tampo S/A pnh	MSTS120	Monagem do item MST120 - tampo S//MSA001 prb quad fin (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS210	Montar tampo S/A prb rei	MSTS210	Monagem do item MST210 - tampo S//MSA001 prb red fin (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS220	Montar tampo S/A pnh re	MSTS220	Monagem do item MST220 - tampo S//MSA001 prb red fin (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS310	Montar tampo S/A prb	MSTS310	Monagem do item MST310 - tampo S//MSA001 prb quad esp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS320	Montar tampo S/A pnh	MSTS320	Monagem do item MST320 - tampo S//MSA001 prb quad esp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS410	Montar tampo S/A prb rei	MSTS410	Monagem do item MST410 - tampo S//MSA001 prb red esp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSTS420	Montar tampo S/A pnh re	MSTS420	Monagem do item MST420 - tampo S//MSA001 prb red esp (sem acabamento)	MSA001	0,00	0,00	0,00	0,00	0,00
OPMSVA110	Acab trv ms prb quad crt	MSVA110	Dar acabamento ao item MSVA110 - trv ms prb quad crt	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSVA120	Acab trv ms pnh quad crt	MSVA120	Dar acabamento ao item MSVA120 - trv ms pnh quad crt	LXD001	0,00	0,00	0,00	0,00	0,00
OPMSVA210	Acab trv ms prb quad lng	MSVA210	Dar acabamento ao item MSVA210 - trv ms prb quad lng	LXD001	0,00	0,00	0,00	0,00	0,00

Código da operação:	Operação:	Produto final:	Descrição:	Recurso de Trabalho:	Lote mínimo:	Lote máximo:	Quantidade múltipla:	Set-Up (mesmo item):	Set-Up (mesma família):
OPMSVA220	Acab trv ms pnh quad lng MSVA220	MSVA220	Dar acabamento ao item ms pnh quad lng	MSVA220 - trv LXD001	0,00	0,00	0,00	0,00	0,00
OPMSVA310	Acab trv ms prb red MSVA310	MSVA310	Dar acabamento ao item ms prb red	MSVA310 - trv LXD002	0,00	0,00	0,00	0,00	0,00
OPMSVA320	Acab trv ms pnh red MSVA320	MSVA320	Dar acabamento ao item ms pnh red	MSVA320 - trv LXD002	0,00	0,00	0,00	0,00	0,00
OPMSVN110	Cort trv ms N/A prb quad MSVN110 crt	MSVN110	Corte de matéria-prima para o item MSVN110 - trv ms N/A prb quad crt	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSVN120	Cort trv ms N/A pnh quad MSVN120 crt	MSVN120	Corte de matéria-prima para o item MSVN120 - trv ms N/A pnh quad crt	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSVN210	Cort trv ms N/A prb quad MSVN210 lng	MSVN210	Corte de matéria-prima para o item MSVN210 - trv ms N/A prb quad lng	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSVN220	Cort trv ms N/A pnh quad MSVN220 lng	MSVN220	Corte de matéria-prima para o item MSVN220 - trv ms N/A pnh quad lng	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSVN310	Cort trv ms N/A prb red MSVN310	MSVN310	Corte de matéria-prima para o item MSVN310 - trv ms N/A prb red	SRA001	0,00	0,00	0,00	0,00	0,00
OPMSVN320	Cort trv ms N/A pnh red MSVN320	MSVN320	Corte de matéria-prima para o item MSVN320 - trv ms N/A pnh red	SRA001	0,00	0,00	0,00	0,00	0,00

Código da operação:	Operação:	Set-Up (item diverso):	Fila esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPCD1110	Acab final - CD1110	10,00	480,00	10,80	8,00	FALSO	min	5,74
OPCD1120	Acab final - CD1120	10,00	480,00	10,80	8,00	FALSO	min	5,74
OPCD1210	Acab final - CD1210	10,00	480,00	10,80	8,00	FALSO	min	5,25
OPCD1220	Acab final - CD1220	10,00	480,00	10,80	8,00	FALSO	min	5,25
OPCD1310	Acab final - CD1310	10,00	480,00	10,80	8,00	FALSO	min	5,49
OPCD1320	Acab final - CD1320	10,00	480,00	10,80	8,00	FALSO	min	5,49
OPCD1410	Acab final - CD1410	10,00	480,00	10,80	8,00	FALSO	min	5,04
OPCD1420	Acab final - CD1420	10,00	480,00	10,80	8,00	FALSO	min	5,04
OPCD2110	Acab final - CD2110	10,00	480,00	10,80	8,00	FALSO	min	6,58
OPCD2120	Acab final - CD2120	10,00	480,00	10,80	8,00	FALSO	min	6,58
OPCD2210	Acab final - CD2210	10,00	480,00	10,80	8,00	FALSO	min	5,95
OPCD2220	Acab final - CD2220	10,00	480,00	10,80	8,00	FALSO	min	5,95
OPCD2310	Acab final - CD2310	10,00	480,00	10,80	8,00	FALSO	min	6,25
OPCD2320	Acab final - CD2320	10,00	480,00	10,80	8,00	FALSO	min	6,25
OPCD2410	Acab final - CD2410	10,00	480,00	10,80	8,00	FALSO	min	5,68
OPCD2420	Acab final - CD2420	10,00	480,00	10,80	8,00	FALSO	min	5,68
OPCDAA100	Acab asnto peroba	9,00	480,00	3,60	4,00	FALSO	min	45,02
OPCDAA110	Acab asnto pinho	9,00	480,00	3,60	4,00	FALSO	min	51,43
OPCDAN100	Cort asnto N/A peroba	3,00	300,00	1,20	4,00	FALSO	min	40,23
OPCDAN110	Cort asnto N/A pinho	3,00	300,00	1,20	4,00	FALSO	min	60,14
OPCDAS100	Form asnto S/A peroba	15,00	720,00	4,80	8,00	FALSO	min	86,79
OPCDAS110	Form asnto S/A pinho	15,00	720,00	4,80	8,00	FALSO	min	105,99
OPCDEA100	Acab enc cadr prb	9,00	420,00	3,60	4,00	FALSO	min	90,04
OPCDEA110	Acab enc cadr pnh	9,00	420,00	3,60	4,00	FALSO	min	76,19

Código da operação:	Operação:	Set-Up (Item diverso):	Fila esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPCDEN100	Cort enc cadr N/A prb	3,00	300,00	1,20	4,00	FALSO	min	116,67
OPCDEN110	Cort enc cadr N/A pnh	3,00	300,00	1,20	4,00	FALSO	min	68,63
OPCDES100	Form enc cadr S/A prb	4,00	480,00	72,00	0,00	VERDADEIRO	min	34,48
OPCDES110	Form enc cadr S/A pnh	4,00	480,00	72,00	0,00	VERDADEIRO	min	23,26
OPCDPA100	Acab pe cadr prb smp	8,00	360,00	1,20	4,00	FALSO	min	147,06
OPCDPA110	Acab pe cadr pnh smp	8,00	360,00	1,20	4,00	FALSO	min	208,33
OPCDPA200	Acab pe cadr prb trb	8,00	720,00	3,60	4,00	FALSO	min	74,23
OPCDPA210	Acab pe cadr pnh trb	8,00	720,00	3,60	4,00	FALSO	min	93,51
OPCDPN100	Cort pe cadr N/A prb smp3,00		300,00	0,36	4,00	FALSO	min	54,26
OPCDPN110	Cort pe cadr N/A pnh smp3,00		300,00	0,36	4,00	FALSO	min	101,45
OPCDSA110	Montar cadr S/A prb, pe smp, est smp	5,00	720,00	19,20	8,00	FALSO	min	5,11
OPCDSA120	Montar cadr S/A prb, pe trb, est smp	5,00	720,00	19,20	8,00	FALSO	min	4,58
OPCDSA130	Montar cadr S/A prb, pe smp, est trb	5,00	720,00	19,20	8,00	FALSO	min	4,83
OPCDSA140	Montar cadr S/A prb, pe trb, est trb	5,00	720,00	19,20	8,00	FALSO	min	4,36
OPCDSA210	Montar cadr S/A pnh, pe smp, est smp	5,00	720,00	19,20	8,00	FALSO	min	6,06
OPCDSA220	Montar cadr S/A pnh, pe trb, est smp	5,00	720,00	19,20	8,00	FALSO	min	5,34
OPCDSA230	Montar cadr S/A pnh, pe smp, est trb	5,00	720,00	19,20	8,00	FALSO	min	5,67
OPCDSA240	Montar cadr S/A pnh, pe trb, est trb	5,00	720,00	19,20	8,00	FALSO	min	5,04
OPCDTA100	Acab est enc cadr prb smp	8,00	360,00	1,20	4,00	FALSO	min	113,64
OPCDTA110	Acab est enc cadr pnh smp	8,00	360,00	1,20	4,00	FALSO	min	172,41
OPCDTA200	Acab est enc cadr prb trb	8,00	720,00	3,60	4,00	FALSO	min	61,54
OPCDTA210	Acab est enc cadr pnh trb	8,00	720,00	3,60	4,00	FALSO	min	82,76
OPCDTN100	Cort est enc cadr N/A prb smp	3,00	300,00	0,36	4,00	FALSO	min	37,04
OPCDTN110	Cort est enc cadr N/A pnh smp	3,00	300,00	0,36	4,00	FALSO	min	70,71

Código da operação:	Operação:	Set-Up (item diverso):	Fla esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPCDVA100	Acab trav cadr prb	9,00	300,00	1,20	4,00	FALSO	min	203,92
OPCDVA110	Acab trav cadr pnh	9,00	300,00	1,20	4,00	FALSO	min	229,20
OPCDVN100	Cort trav cadr N/A prb	3,00	300,00	0,36	4,00	FALSO	min	129,63
OPCDVN110	Cort trav cadr N/A pnh	3,00	300,00	0,36	4,00	FALSO	min	163,74
OPMS1110	Acab final - MS1110	10,00	480,00	14,40	8,00	FALSO	min	3,20
OPMS1120	Acab final - MS1120	10,00	480,00	14,40	8,00	FALSO	min	3,20
OPMS1210	Acab final - MS1210	10,00	480,00	14,40	8,00	FALSO	min	3,48
OPMS1220	Acab final - MS1220	10,00	480,00	14,40	8,00	FALSO	min	3,48
OPMS2110	Acab final - MS2110	10,00	480,00	14,40	8,00	FALSO	min	2,95
OPMS2120	Acab final - MS2120	10,00	480,00	14,40	8,00	FALSO	min	2,95
OPMS2210	Acab final - MS2210	10,00	480,00	14,40	8,00	FALSO	min	3,19
OPMS2220	Acab final - MS2220	10,00	480,00	14,40	8,00	FALSO	min	3,19
OPMS3310	Acab final - MS3310	10,00	480,00	14,40	8,00	FALSO	min	3,11
OPMS3320	Acab final - MS3320	10,00	480,00	14,40	8,00	FALSO	min	3,11
OPMS3410	Acab final - MS3410	10,00	480,00	14,40	8,00	FALSO	min	3,49
OPMS3420	Acab final - MS3420	10,00	480,00	14,40	8,00	FALSO	min	3,49
OPMS4310	Acab final - MS4310	10,00	480,00	14,40	8,00	FALSO	min	3,03
OPMS4320	Acab final - MS4320	10,00	480,00	14,40	8,00	FALSO	min	3,03
OPMS4410	Acab final - MS4410	10,00	480,00	14,40	8,00	FALSO	min	3,39
OPMS4420	Acab final - MS4420	10,00	480,00	14,40	8,00	FALSO	min	3,39
OPMS5110	Acab final - MS5110	10,00	480,00	14,40	8,00	FALSO	min	2,80
OPMS5120	Acab final - MS5120	10,00	480,00	14,40	8,00	FALSO	min	2,80
OPMS5210	Acab final - MS5210	10,00	480,00	14,40	8,00	FALSO	min	3,01
OPMS5220	Acab final - MS5220	10,00	480,00	14,40	8,00	FALSO	min	3,01

Código da operação:	Operação:	Set-Up (Item diverso):	Fila esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPMS6110	Acab final - MS6110	10,00	480,00	14,40	8,00	FALSO	min	2,61
OPMS6120	Acab final - MS6120	10,00	480,00	14,40	8,00	FALSO	min	2,61
OPMS6210	Acab final - MS6210	10,00	480,00	14,40	8,00	FALSO	min	2,79
OPMS6220	Acab final - MS6220	10,00	480,00	14,40	8,00	FALSO	min	2,79
OPMS7310	Acab final - MS7310	10,00	480,00	14,40	8,00	FALSO	min	2,87
OPMS7320	Acab final - MS7320	10,00	480,00	14,40	8,00	FALSO	min	2,87
OPMS7410	Acab final - MS7410	10,00	480,00	14,40	8,00	FALSO	min	3,20
OPMS7420	Acab final - MS7420	10,00	480,00	14,40	8,00	FALSO	min	3,20
OPMS8310	Acab final - MS8310	10,00	480,00	14,40	8,00	FALSO	min	2,80
OPMS8320	Acab final - MS8320	10,00	480,00	14,40	8,00	FALSO	min	2,80
OPMS8410	Acab final - MS8410	10,00	480,00	14,40	8,00	FALSO	min	3,11
OPMS8420	Acab final - MS8420	10,00	480,00	14,40	8,00	FALSO	min	3,11
OPMSPA110	Acab pe ms prb quad	9,00	300,00	3,00	4,00	FALSO	min	52,25
OPMSPA120	Acab pe ms prb quad	9,00	360,00	3,00	4,00	FALSO	min	56,66
OPMSPA210	Acab pe ms prb red	9,00	300,00	3,00	4,00	FALSO	min	79,06
OPMSPA220	Acab pe ms prb red	9,00	360,00	3,00	4,00	FALSO	min	84,50
OPMSPN110	Cort pe N/A ms prb quad	3,00	300,00	0,36	4,00	FALSO	min	22,65
OPMSPN120	Cort pe N/A ms prb quad	3,00	300,00	0,36	4,00	FALSO	min	16,32
OPMSPN210	Cort pe N/A ms prb red	3,00	300,00	0,36	4,00	FALSO	min	44,03
OPMSPN220	Cort pe N/A ms prb red	3,00	300,00	0,36	4,00	FALSO	min	31,96
OPMSSA110	mont ms S/A prb qd fin, f 5,00 prb qd, vnz		720,00	16,80	8,00	FALSO	min	3,20
OPMSSA120	mont ms S/A prb qd fin, f 5,00 prb red, vnz		720,00	16,80	8,00	FALSO	min	3,60

Código da operação:	Operação:	Set-Up (Item diverso):	Fila esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPMSSA210	mont ms S/A prb red fin, 5,00 pe prb qd, vnz		720,00	16,80	8,00	FALSO	min	2,87
OPMSSA220	mont ms S/A prb red fin, 5,00 pe prb red, vnz		720,00	16,80	8,00	FALSO	min	3,18
OPMSSA330	mont ms S/A pnh qd fin, 5,00 pe pnh qd, vnz		720,00	16,80	8,00	FALSO	min	3,08
OPMSSA340	mont ms S/A pnh qd fin, 5,00 pe pnh red, vnz		720,00	16,80	8,00	FALSO	min	3,62
OPMSSA430	mont ms S/A pnh red fin, 5,00 pe pnh qd, vnz		720,00	16,80	8,00	FALSO	min	2,97
OPMSSA440	mont ms S/A pnh red fin, 5,00 pe pnh red, vnz		720,00	16,80	8,00	FALSO	min	3,47
OPMSSA510	mont ms S/A prb qd esp, 5,00 pe prb qd, vnz		720,00	16,80	8,00	FALSO	min	2,67
OPMSSA520	mont ms S/A prb qd esp, 5,00 pe prb red, vnz		720,00	16,80	8,00	FALSO	min	2,95
OPMSSA610	mont ms S/A prb red esp 5,00 pe prb qd, vnz		720,00	16,80	8,00	FALSO	min	2,44
OPMSSA620	mont ms S/A prb red esp 5,00 pe prb red, vnz		720,00	16,80	8,00	FALSO	min	2,66
OPMSSA730	mont ms S/A pnh qd esp, 5,00 pe pnh qd, vnz		720,00	16,80	8,00	FALSO	min	2,76
OPMSSA740	mont ms S/A pnh qd esp, 5,00 pe pnh red, vnz		720,00	16,80	8,00	FALSO	min	3,19
OPMSSA830	mont ms S/A pnh red esp 5,00 pe pnh qd, vnz		720,00	16,80	8,00	FALSO	min	2,68
OPMSSA840	mont ms S/A pnh red esp 5,00 pe pnh red, vnz		720,00	16,80	8,00	FALSO	min	3,07
OPMSTA110	Acab tempo prb quad fin 9,00		300,00	6,00	4,00	FALSO	min	10,98
OPMSTA120	Acab tempo pnh quad fin 9,00		300,00	6,00	4,00	FALSO	min	13,77
OPMSTA210	Acab tempo prb red fin 9,00		300,00	6,00	4,00	FALSO	min	10,98

Código da Operação:	Operação:	Set-Up (Item diverso):	Fila esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPMSTA220	Acab tempo pnh red fin	9,00	300,00	6,00	4,00	FALSO	min	13,77
OPMSTA310	Acab tempo prb quad esp	9,00	300,00	6,00	4,00	FALSO	min	7,29
OPMSTA320	Acab tempo pnh quad esp	9,00	300,00	6,00	4,00	FALSO	min	9,97
OPMSTA410	Acab tempo prb red esp	9,00	300,00	6,00	4,00	FALSO	min	7,29
OPMSTA420	Acab tempo pnh red esp	9,00	300,00	6,00	4,00	FALSO	min	9,97
OPMSTN110	Cort tempo N/A prb quad	3,00	300,00	1,80	4,00	FALSO	min	8,28
OPMSTN120	Cort tempo N/A pnh quad	3,00	300,00	1,80	4,00	FALSO	min	13,33
OPMSTN210	Cort tempo N/A prb red fin	3,00	300,00	1,80	4,00	FALSO	min	8,28
OPMSTN220	Cort tempo N/A pnh red	3,00	300,00	1,80	4,00	FALSO	min	13,33
OPMSTN310	Cort tempo N/A prb quad	3,00	300,00	1,80	4,00	FALSO	min	4,26
OPMSTN320	Cort tempo N/A pnh quad	3,00	300,00	1,80	4,00	FALSO	min	6,97
OPMSTN410	Cort tempo N/A prb red	3,00	300,00	1,80	4,00	FALSO	min	4,26
OPMSTN420	Cort tempo N/A pnh red	3,00	300,00	1,80	4,00	FALSO	min	6,97
OPMSTS110	Montar tempo S/A prb	5,00	360,00	14,40	8,00	FALSO	min	9,43
OPMSTS120	Montar tempo S/A pnh	5,00	360,00	14,40	8,00	FALSO	min	12,28
OPMSTS210	Montar tempo S/A prb rei	5,00	360,00	14,40	8,00	FALSO	min	9,43
OPMSTS220	Montar tempo S/A pnh rei	5,00	360,00	14,40	8,00	FALSO	min	12,28
OPMSTS310	Montar tempo S/A prb	5,00	360,00	14,40	8,00	FALSO	min	5,97
OPMSTS320	Montar tempo S/A pnh	5,00	360,00	14,40	8,00	FALSO	min	8,45
OPMSTS410	Montar tempo S/A prb rei	5,00	360,00	14,40	8,00	FALSO	min	5,97
OPMSTS420	Montar tempo S/A pnh rei	5,00	360,00	14,40	8,00	FALSO	min	8,45
OPMSVA110	Acab trv ms prb quad crt	9,00	300,00	1,20	4,00	FALSO	min	126,06
OPMSVA120	Acab trv ms pnh quad crt	9,00	300,00	1,20	4,00	FALSO	min	150,72
OPMSVA210	Acab trv ms prb quad lng	9,00	300,00	1,80	4,00	FALSO	min	92,44

Código da operação:	Operação:	Set-Up (item diverso):	Fila esperada:	Tempo de processo:	Clean-Up:	Tempo constante:	Unidade de tempo:	EPP:
OPMSVA220	Acab trv ms pnh quad lng	9,00	300,00	1,80	4,00	FALSO	min	110,05
OPMSVA310	Acab trv ms prb red	9,00	360,00	1,20	4,00	FALSO	min	96,73
OPMSVA320	Acab trv ms pnh red	9,00	360,00	1,20	4,00	FALSO	min	154,76
OPMSVN110	Cort trv ms N/A prb quad crt	3,00	300,00	0,36	4,00	FALSO	min	59,83
OPMSVN120	Cort trv ms N/A pnh quad crt	3,00	300,00	0,36	4,00	FALSO	min	77,78
OPMSVN210	Cort trv ms N/A prb quad lng	3,00	300,00	0,36	4,00	FALSO	min	45,75
OPMSVN220	Cort trv ms N/A pnh quad lng	3,00	300,00	0,36	4,00	FALSO	min	59,83
OPMSVN310	Cort trv ms N/A prb red	3,00	300,00	0,36	4,00	FALSO	min	26,82
OPMSVN320	Cort trv ms N/A pnh red	3,00	300,00	0,36	4,00	FALSO	min	51,85

ANEXO D - Entradas de operações

Código da operação	Qtde Base	Ordem	Entrada	Qtde Entrada
OPCD1110	1,00	1	CDSA110	1,00
OPCD1110	1,00	2	XAC0100	0,20
OPCD1120	1,00	1	CDSA110	1,00
OPCD1120	1,00	2	XAC0200	0,20
OPCD1210	1,00	1	CDSA120	1,00
OPCD1210	1,00	2	XAC0100	0,20
OPCD1220	1,00	1	CDSA120	1,00
OPCD1220	1,00	2	XAC0100	0,20
OPCD1310	1,00	1	CDSA130	1,00
OPCD1310	1,00	2	XAC0100	0,20
OPCD1320	1,00	1	CDSA130	1,00
OPCD1320	1,00	2	XAC0100	0,20
OPCD1410	1,00	1	CDSA140	1,00
OPCD1410	1,00	2	XAC0100	0,20
OPCD1420	1,00	1	CDSA140	1,00
OPCD1420	1,00	2	XAC0100	0,20
OPCD2110	1,00	1	CDSA210	1,00
OPCD2110	1,00	2	XAC0100	0,20
OPCD2120	1,00	1	CDSA210	1,00
OPCD2120	1,00	2	XAC0200	0,20
OPCD2210	1,00	1	CDSA220	1,00
OPCD2210	1,00	2	XAC0100	0,20
OPCD2220	1,00	1	CDSA220	1,00
OPCD2220	1,00	2	XAC0200	0,20
OPCD2310	1,00	1	CDSA230	1,00
OPCD2310	1,00	2	XAC0100	0,20
OPCD2320	1,00	1	CDSA230	1,00
OPCD2320	1,00	2	XAC0200	0,20
OPCD2410	1,00	1	CDSA240	1,00
OPCD2410	1,00	2	XAC0100	0,20
OPCD2420	1,00	1	CDSA240	1,00
OPCD2420	1,00	2	XAC0200	0,20
OPCDAA100	1,00	1	CDAS100	1,00
OPCDAA110	1,00	1	CDAS110	1,00
OPCDAN100	5,00	1	XTB0100	3,00
OPCDAN110	5,00	1	XTB1100	3,00
OPCDAS100	1,00	1	CDAN100	1,00
OPCDAS110	1,00	1	CDAN110	1,00
OPCDEA100	1,00	1	CDES100	1,00
OPCDEA110	1,00	1	CDES110	1,00
OPCDEN100	1,00	1	XRP0200	0,50
OPCDEN110	1,00	1	XRP1200	0,50
OPCDES100	1,00	1	CDEN100	1,00
OPCDES110	1,00	1	CDEN110	1,00
OPCDPA100	4,00	1	CDPN100	4,00
OPCDPA110	4,00	1	CDPN110	4,00
OPCDPA200	4,00	1	CDPN100	4,00
OPCDPA210	4,00	1	CDPN110	4,00
OPCDPN100	4,00	1	XVG0300	2,00
OPCDPN110	4,00	1	XVG1300	2,00
OPCDSA110	1,00	1	CDAA100	1,00

ANEXO D - Entradas de operações

Código da operação	Qtde Base	Ordem	Entrada	Qtde Entrada
OPCDSA110	1,00	2	CDPA100	4,00
OPCDSA110	1,00	3	CDVA100	4,00
OPCDSA110	1,00	4	CDTA100	2,00
OPCDSA110	1,00	5	CDEA100	1,00
OPCDSA110	1,00	6	XFX0100	0,05
OPCDSA120	1,00	1	CDAA100	1,00
OPCDSA120	1,00	2	CDPA200	4,00
OPCDSA120	1,00	3	CDVA100	4,00
OPCDSA120	1,00	4	CDTA100	2,00
OPCDSA120	1,00	5	CDEA100	1,00
OPCDSA120	1,00	6	XFX0100	0,05
OPCDSA130	1,00	1	CDAA100	1,00
OPCDSA130	1,00	2	CDPA100	4,00
OPCDSA130	1,00	3	CDVA100	4,00
OPCDSA130	1,00	4	CDTA200	2,00
OPCDSA130	1,00	5	CDEA100	1,00
OPCDSA130	1,00	6	XFX0100	0,05
OPCDSA140	1,00	1	CDAA100	1,00
OPCDSA140	1,00	2	CDPA200	4,00
OPCDSA140	1,00	3	CDVA100	4,00
OPCDSA140	1,00	4	CDTA200	2,00
OPCDSA140	1,00	5	CDEA100	1,00
OPCDSA140	1,00	6	XFX0100	0,05
OPCDSA210	1,00	1	CDAA110	1,00
OPCDSA210	1,00	2	CDPA110	4,00
OPCDSA210	1,00	3	CDVA110	4,00
OPCDSA210	1,00	4	CDTA110	2,00
OPCDSA210	1,00	5	CDEA110	1,00
OPCDSA210	1,00	6	XFX0100	0,05
OPCDSA220	1,00	1	CDAA110	1,00
OPCDSA220	1,00	2	CDPA210	4,00
OPCDSA220	1,00	3	CDVA110	4,00
OPCDSA220	1,00	4	CDTA110	2,00
OPCDSA220	1,00	5	CDEA110	1,00
OPCDSA220	1,00	6	XFX0100	0,05
OPCDSA230	1,00	1	CDAA110	1,00
OPCDSA230	1,00	2	CDPA110	4,00
OPCDSA230	1,00	3	CDVA110	4,00
OPCDSA230	1,00	4	CDTA210	2,00
OPCDSA230	1,00	5	CDEA110	1,00
OPCDSA230	1,00	6	XFX0100	0,05
OPCDSA240	1,00	1	CDAA110	1,00
OPCDSA240	1,00	2	CDPA210	4,00
OPCDSA240	1,00	3	CDVA110	4,00
OPCDSA240	1,00	4	CDTA210	2,00
OPCDSA240	1,00	5	CDEA110	1,00
OPCDSA240	1,00	6	XFX0100	0,05
OPCDTA100	2,00	1	CDTN100	2,00
OPCDTA110	2,00	1	CDTN110	2,00
OPCDTA200	2,00	1	CDTN100	2,00
OPCDTA210	2,00	1	CDTN110	2,00

ANEXO D - Entradas de operações

Código da operação	Qtde Base	Ordem	Entrada	Qtde Entrada
OPCDTN100	2,00	1	XVG0300	1,50
OPCDTN110	2,00	1	XVG1300	1,50
OPCDVA100	4,00	1	CDVN100	4,00
OPCDVA110	4,00	1	CDVN110	4,00
OPCDVN100	4,00	1	XVG0100	1,50
OPCDVN110	4,00	1	XVG1100	1,50
OPMS1110	1,00	1	MSSA110	1,00
OPMS1110	1,00	2	XAC0100	0,75
OPMS1120	1,00	1	MSSA110	1,00
OPMS1120	1,00	2	XAC0200	0,75
OPMS1210	1,00	1	MSSA120	1,00
OPMS1210	1,00	2	XAC0100	0,75
OPMS1220	1,00	1	MSSA120	1,00
OPMS1220	1,00	2	XAC0200	0,75
OPMS2110	1,00	1	MSSA210	1,00
OPMS2110	1,00	2	XAC0100	0,75
OPMS2120	1,00	1	MSSA210	1,00
OPMS2120	1,00	2	XAC0200	0,75
OPMS2210	1,00	1	MSSA220	1,00
OPMS2210	1,00	2	XAC0100	0,75
OPMS2220	1,00	1	MSSA220	1,00
OPMS2220	1,00	2	XAC0200	0,75
OPMS3310	1,00	1	MSSA330	1,00
OPMS3310	1,00	2	XAC0100	0,75
OPMS3320	1,00	1	MSSA330	1,00
OPMS3320	1,00	2	XAC0200	0,75
OPMS3410	1,00	1	MSSA340	1,00
OPMS3410	1,00	2	XAC0100	0,75
OPMS3420	1,00	1	MSSA340	1,00
OPMS3420	1,00	2	XAC0200	0,75
OPMS4310	1,00	1	MSSA430	1,00
OPMS4310	1,00	2	XAC0100	0,75
OPMS4320	1,00	1	MSSA430	1,00
OPMS4320	1,00	2	XAC0200	0,75
OPMS4410	1,00	1	MSSA440	1,00
OPMS4410	1,00	2	XAC0100	0,75
OPMS4420	1,00	1	MSSA440	1,00
OPMS4420	1,00	2	XAC0200	0,75
OPMS5110	1,00	1	MSSA510	1,00
OPMS5110	1,00	2	XAC0100	0,75
OPMS5120	1,00	1	MSSA510	1,00
OPMS5120	1,00	2	XAC0200	0,75
OPMS5210	1,00	1	MSSA520	1,00
OPMS5210	1,00	2	XAC0100	0,75
OPMS5220	1,00	1	MSSA520	1,00
OPMS5220	1,00	2	XAC0200	0,75
OPMS6110	1,00	1	MSSA610	1,00
OPMS6110	1,00	2	XAC0100	0,75
OPMS6120	1,00	1	MSSA610	1,00
OPMS6120	1,00	2	XAC0200	0,75
OPMS6210	1,00	1	MSSA620	1,00

ANEXO D - Entradas de operações

Código da operação	Qtde Base	Ordem	Entrada	Qtde Entrada
OPMS6210	1,00	2	XAC0100	0,75
OPMS6220	1,00	1	MSSA620	1,00
OPMS6220	1,00	2	XAC0200	0,75
OPMS7310	1,00	1	MSSA730	1,00
OPMS7310	1,00	2	XAC0100	0,75
OPMS7320	1,00	1	MSSA730	1,00
OPMS7320	1,00	2	XAC0200	0,75
OPMS7410	1,00	1	MSSA740	1,00
OPMS7410	1,00	2	XAC0100	0,75
OPMS7420	1,00	1	MSSA740	1,00
OPMS7420	1,00	2	XAC0200	0,75
OPMS8310	1,00	1	MSSA830	1,00
OPMS8310	1,00	2	XAC0100	0,75
OPMS8320	1,00	1	MSSA830	1,00
OPMS8320	1,00	2	XAC0200	0,75
OPMS8410	1,00	1	MSSA840	1,00
OPMS8410	1,00	2	XAC0100	0,75
OPMS8420	1,00	1	MSSA840	1,00
OPMS8420	1,00	2	XAC0200	0,75
OPMSPA110	4,00	1	MSPN110	3,60
OPMSPA120	4,00	1	MSPN120	3,60
OPMSPA210	4,00	1	MSPN210	3,60
OPMSPA220	4,00	1	MSPN220	3,60
OPMSPN110	4,00	1	XVG0200	5,00
OPMSPN120	4,00	1	XVG0400	5,00
OPMSPN210	4,00	1	XVG1200	5,00
OPMSPN220	4,00	1	XVG1400	5,00
OPMSSA110	1,00	1	MSTA110	1,00
OPMSSA110	1,00	2	MSPA110	4,00
OPMSSA110	1,00	3	MSVA110	2,00
OPMSSA110	1,00	4	MSVA210	2,00
OPMSSA110	1,00	5	AFX0100	0,15
OPMSSA120	1,00	1	MSTA110	1,00
OPMSSA120	1,00	2	MSPA210	4,00
OPMSSA120	1,00	3	MSVA110	2,00
OPMSSA120	1,00	4	MSVA210	2,00
OPMSSA120	1,00	5	AFX0100	0,15
OPMSSA210	1,00	1	MSTA210	1,00
OPMSSA210	1,00	2	MSPA110	4,00
OPMSSA210	1,00	3	MSVA310	4,00
OPMSSA210	1,00	4	AFX0100	0,15
OPMSSA220	1,00	1	MSTA210	1,00
OPMSSA220	1,00	2	MSPA210	4,00
OPMSSA220	1,00	3	MSVA310	4,00
OPMSSA220	1,00	4	AFX0100	0,15
OPMSSA330	1,00	1	MSTA120	1,00
OPMSSA330	1,00	2	MSPA120	4,00
OPMSSA330	1,00	3	MSVA120	2,00
OPMSSA330	1,00	4	MSVA220	2,00
OPMSSA330	1,00	5	AFX0100	0,15
OPMSSA340	1,00	1	MSTA120	1,00

ANEXO D - Entradas de operações

Código da operação	Qtde Base	Ordem	Entrada	Qtde Entrada
OPMSSA340	1,00	2	MSPA220	4,00
OPMSSA340	1,00	3	MSVA120	2,00
OPMSSA340	1,00	4	MSVA220	2,00
OPMSSA340	1,00	5	XFX0100	0,15
OPMSSA430	1,00	1	MSTA220	1,00
OPMSSA430	1,00	2	MSPA120	4,00
OPMSSA430	1,00	3	MSVA320	4,00
OPMSSA430	1,00	4	XFX0100	0,15
OPMSSA430	1,00	1	MSTA220	1,00
OPMSSA440	1,00	2	MSPA220	4,00
OPMSSA440	1,00	3	MSVA320	4,00
OPMSSA440	1,00	4	XFX0100	0,15
OPMSSA440	1,00	1	MSTA310	1,00
OPMSSA510	1,00	2	MSPA110	4,00
OPMSSA510	1,00	3	MSVA110	2,00
OPMSSA510	1,00	4	MSVA210	2,00
OPMSSA510	1,00	5	XFX0100	0,15
OPMSSA510	1,00	1	MSTA310	1,00
OPMSSA520	1,00	2	MSPA210	4,00
OPMSSA520	1,00	3	MSVA110	2,00
OPMSSA520	1,00	4	MSVA210	2,00
OPMSSA520	1,00	5	XFX0100	0,15
OPMSSA520	1,00	1	MSTA410	1,00
OPMSSA610	1,00	2	MSPA110	4,00
OPMSSA610	1,00	3	MSVA310	4,00
OPMSSA610	1,00	4	XFX0100	0,15
OPMSSA610	1,00	1	MSTA410	1,00
OPMSSA620	1,00	2	MSPA210	4,00
OPMSSA620	1,00	3	MSVA310	4,00
OPMSSA620	1,00	4	XFX0100	0,15
OPMSSA620	1,00	1	MSTA320	1,00
OPMSSA730	1,00	2	MSPA120	4,00
OPMSSA730	1,00	3	MSVA120	2,00
OPMSSA730	1,00	4	MSVA220	2,00
OPMSSA730	1,00	5	XFX0100	0,15
OPMSSA730	1,00	1	MSTA320	1,00
OPMSSA740	1,00	2	MSPA220	4,00
OPMSSA740	1,00	3	MSVA120	2,00
OPMSSA740	1,00	4	MSVA220	2,00
OPMSSA740	1,00	5	XFX0100	0,15
OPMSSA740	1,00	1	MSTA420	1,00
OPMSSA830	1,00	2	MSPA120	4,00
OPMSSA830	1,00	3	MSVA320	4,00
OPMSSA830	1,00	4	XFX0100	0,15
OPMSSA830	1,00	1	MSTA420	1,00
OPMSSA840	1,00	2	MSPA220	4,00
OPMSSA840	1,00	3	MSVA320	4,00
OPMSSA840	1,00	4	XFX0100	0,15
OPMSSA840	1,00	1	MSTS110	1,00
OPMSTA110	1,00	1	MSTS120	1,00
OPMSTA120	1,00	1	MSTS210	1,00
OPMSTA210	1,00			

ANEXO D - Entradas de operações

Código da operação	Qtde Base	Ordem	Entrada	Qtde Entrada
OPMSTA220	1,00	1	MSTS220	1,00
OPMSTA310	1,00	1	MSTS310	1,00
OPMSTA320	1,00	1	MSTS320	1,00
OPMSTA410	1,00	1	MSTS410	1,00
OPMSTA420	1,00	1	MSTS420	1,00
OPMSTN110	1,00	1	XTB0200	2,00
OPMSTN120	1,00	1	XTB1200	2,00
OPMSTN210	1,00	1	XTB0200	2,00
OPMSTN220	1,00	1	XTB1200	2,00
OPMSTN310	1,00	1	XTB0400	2,00
OPMSTN320	1,00	1	XTB1400	2,00
OPMSTN410	1,00	1	XTB0400	2,00
OPMSTN420	1,00	1	XTB1400	2,00
OPMSTS110	1,00	1	MSTN110	1,00
OPMSTS110	1,00	2	XFX0100	0,30
OPMSTS120	1,00	1	MSTN120	1,00
OPMSTS120	1,00	2	XFX0100	0,30
OPMSTS210	1,00	1	MSTN210	1,00
OPMSTS210	1,00	2	XFX0100	0,30
OPMSTS220	1,00	1	MSTN220	1,00
OPMSTS220	1,00	2	XFX0100	0,30
OPMSTS310	1,00	1	MSTN310	1,00
OPMSTS310	1,00	2	XFX0100	0,30
OPMSTS320	1,00	1	MSTN320	1,00
OPMSTS320	1,00	2	XFX0100	0,30
OPMSTS410	1,00	1	MSTN410	1,00
OPMSTS410	1,00	2	XFX0100	0,30
OPMSTS420	1,00	1	MSTN420	1,00
OPMSTS420	1,00	2	XFX0100	0,30
OPMSVA110	4,00	1	MSVN110	4,00
OPMSVA120	4,00	1	MSVN120	4,00
OPMSVA210	4,00	1	MSVN210	4,00
OPMSVA220	4,00	1	MSVN220	4,00
OPMSVA310	4,00	1	MSVN310	4,00
OPMSVA320	4,00	1	MSVN320	4,00
OPMSVN110	4,00	1	XVG0100	3,60
OPMSVN120	4,00	1	XVG1100	3,60
OPMSVN210	4,00	1	XVG0100	4,80
OPMSVN220	4,00	1	XVG1100	4,80
OPMSVN310	4,00	1	XVG0300	4,20
OPMSVN320	4,00	1	XVG1300	4,20

ANEXO D - Recurso de Trabalho

Código:	Recurso	CT	Descrição:	# disponível	Regra
FRM001	Forma	MQPS	Prensa para dar forma "envergada" a encosto de cadeira	1	FLDN
LXD001	Lixadeira plana	LIXD	Lixadeira para tábuas ou peças planas (ou quase planas)	2	FLDN
LXD002	Lix cilíndrica	LIXD	Lixadeira para vigas e peças cilíndricas	1	FLDN
MSA001	Mesa op	BNCD	Mesa de operações	4	FLDN
SRA001	Serra de fita	SERRA	Serra de fita para corte de madeira pequena	1	FLDN
SRA002	Serra de mesa	SERRA	Serra de mesa para corte de madeira grande	1	FLDN
TRN001	Torno manual	TORNO	Torno manual para trabalhos detalhados	2	FLDN
TRN002	Torno automático	TORNO	Torno automático para operações sem detalhes, mas de grande volume que exigem alta produtividade	1	FLDN

ANEXO D - Disponibilidade

Dia	Início	Encerramento	24 h
1	00:00	00:00	FALSO
2	08:00	18:00	FALSO
3	08:00	18:00	FALSO
4	08:00	18:00	FALSO
5	08:00	18:00	FALSO
6	08:00	18:00	FALSO
7	08:00	12:00	FALSO

Prefixo *	Código	Nome	Descrição
CD	CADR	cadeiras	Produto acabado (pronto para comercialização) do tipo "cadeira"
CDA A	CDA A	asnto acbd	Assento para cadeira com acabamento (exceto pintura)
CDAN	CDAN	asnto nao-acbd	Assento para cadeira cortado nas dimensões próximas às do assento final, porém sem trabalho no formato
CDAS	CDAS	asnto smi-acbd	Assento para cadeira com formato adequado, porém sem acabamento
CDEA	CDEA	enc cadr acbd	Encosto para cadeira cortado com acabamento (exceto pintura)
CDEN	CDEN	enc cadr nao-acbd	Encosto cortado, sem curvatura anatômica adequada e sem acabamento
CDES	CDES	enc cadr smi-acbd	Encosto cortado e com curvatura anatômica adequada, porém sem acabamento
CDPA	CDPA	pe cadr acbd	Pé para cadeira com acabamento (exceto pintura)
CDPN	CDPN	pe cadr nao-acbd	Pé para cadeira cortado na dimensão adequada, porém sem acabamento
CDSA	CDSA	cadr smi acbd	Representa semi-acabados antes da última etapa para composição de um item da família "CADR", falta apenas o acabamento
CDTA	CDTA	estr enc cadr acbd	Estrutura para encosto de cadeira com acabamento (exceto pintura)
CDTN	CDTN	estr enc cadr n-acbd	Estrutura para encosto de cadeira sem acabamento
CDVA	CDVA	trav cadr acbd	Travessa para firmar os pés de cadeira com acabamento (exceto pintura)
CDVN	CDVN	trav cadr nao-acbd	Travessa para firmar os pés de cadeira sem acabamento
MS	MESA	mesas	Produto acabado (pronto para comercialização) do tipo "mesa"
XAC	MPAC	mat acbto	Material de acabamento
XFX	MPFX	mat fixacao	Ripas de madeira
XRP	MPRP	ripas	Ripas de madeira
XTB	MPTB	tabuas	Tábuas em madeira
XVG	MPVG	vigas	Vigas em madeira
MSPA	MSPA	pe mesa acbd	Pé de mesa com acabamento (exceto pintura)
MSPN	MSPN	pe mesa nao-acbd	Vigas cortadas na dimensão devida, porém sem acabamento
MSSA	MSSA	mesa sem acbd	Representa semi-acabados antes da última etapa para composição de um item da família "MESA", falta apenas o acabamento
MSTA	MSTA	tampo acbd	Tampo para mesa com acabamento (exceto pintura)
MSTN	MSTN	tampo não-montado	Tábuas cortadas nas dimensões devidas, porém sem "montagem" (justaposição)
MSTS	MSTS	tampo smi-acbd	Tampo montado (tábuas justapostas), mas sem acabamento
MSVA	MSVA	trav mesa acbd	Travessas - para firmar os pés da mesa - com acabamento (exceto pintura)
MSVN	MSVN	trav mesa nao-acbd	Vigas cortadas na dimensão devida, porém sem acabamento

* Utilizado na formação dos códigos de item alocados às famílias

Código	Centro de Trabalho	Descrição
BNCD	Bancada de montagem	Bancada de mesas de operações de montagem
LIXD	Lixadeiras	Lixadeiras automáticas para acabamento dos móveis antes da pintura
MQPS	Maq pesadas	Máquinas pesadas, como formas, prensas, etc.
SERRA	Serras	Setor que contém as serras de fita e de mesa
TORNO	Tornos	Seção que concentra os tornos

ANEXO D - Categoria

Código	Nome	Externo	Descrição
ME	Mat. Embalagem	VERDADEIRO	Todo tipo de material de embalagem: plástico, papelão
MP	Matéria-Prima	VERDADEIRO	Todo tipo de matéria-prima
PA	Produto Acabado	FALSO	Produtos acabado, ou seja, em seu estado final, pronto para comercialização
SA	Semi-Acabado	FALSO	Produto semi-acabado, ou em processo

Item	Qtde	Data
CD1420	60	08/12/2000 12:00
CD2110	150	08/12/2000 12:00
CD1110	120	08/12/2000 12:00
CD2310	90	08/12/2000 12:00
MS6220	15	08/12/2000 12:00
MS3410	25	08/12/2000 12:00
MS2210	30	08/12/2000 12:00
MS3310	15	08/12/2000 12:00
MSTA110	10	01/12/2000 12:00
MSTA120	10	01/12/2000 12:00
MSTA410	10	01/12/2000 12:00
CDPA200	320	01/12/2000 12:00
CDPA210	160	01/12/2000 12:00
CDPA100	400	01/12/2000 12:00
CDPA110	320	01/12/2000 12:00

Sigla:	Grandeza:	Expoente M:	Expoente L:	Expoente T:
A	Área	0	2	0
comp	Comprimento	0	1	0
m	Massa	1	0	0
t	Tempo	0	0	1
un	Unidade	0	0	0
vol	Volume	0	3	0

ANEXO D - Unidades

Sigla:	Unidade:	Grandeza:	Unidade Padrão:	Equivalência Padrão:
cm	centímetro	comp	FALSO	0,01000
dia	Dia	t	VERDADEIRO	1,00000
h	Hora	t	FALSO	0,04167
L	litro	vol	VERDADEIRO	1,00000
m	metro	comp	VERDADEIRO	1,00000
m2	mt quadrado	A	VERDADEIRO	1000,00000
mil	milheiro	un	FALSO	0,00069
min	Minuto	t	FALSO	7,00000
sem	Semana	t	FALSO	1,00000
un	Unidade	un	VERDADEIRO	1,00000

ANEXO D - Regras de sequenciação

Código:	Descrição:	Input para regra:	Max:	Input para regra:	Max:
DTLM	Menor data de conclusão	Fila DataMaxOP	FALSO	Fila TempoProcesso	FALSO
FIFO	First in - first out	Fila HoraChegada	FALSO	Fila TempoProcesso	FALSO
FLDN	Menor folga dinâmica	Fila FolgaDinamica	FALSO	Fila DataMaxOP	FALSO
MQTD	Menor quantidade	Fila Qtde	FALSO	Fila TempoProcesso	FALSO
MTPR	Menor tempo de processamento	Fila TempoProcesso	FALSO	Fila HoraChegada	FALSO

Item:	Prazo de entrega:
XAC0100	5
XAC0200	0
AFX0100	5
AFX0210	5
AFX0220	5
AFX0310	5
AFX0320	5
XRP0100	5
XRP0200	5
XRP1100	5
XRP1200	5
XTB0100	5
XTB0200	5
XTB0300	5
XTB0400	5
XTB1100	5
XTB1200	5
XTB1300	5
XTB1400	5
XVG0100	5
XVG0200	5
XVG0300	5
XVG0400	5
XVG1100	5
XVG1200	5
XVG1300	5
XVG1400	5

ANEXO E -ESTRUTURA DO BANCO DE DADOS

Nesta seção são registrados todas as tabelas/campos e consultas que compõem o banco de dados

Tabela		Campo		Legenda	Descrição
CamposFila		CmFl_NomeCampo		Nome do campo	Nome dos campos da tabela de Fila
CamposFila		CmFl_Legenda		Legenda	Legenda
CamposFila		CmFl_Desc		Descrição	Descrição do campo
Categoria		Catg_Codigo		Código	Código de categoria (2 caracteres)
Categoria		Catg_Nome		Nome	Nome da categoria (obrigatório com até 15 caracteres)
Categoria		Catg_Origem		Origem externa	Indicar se a categoria tem origem externa (ex.: matéria-prima, componentes comprados....)
Categoria		Catg_Desc		Descrição	Descrições e comentários da categoria (opcional)
CentroDeTrabalho		CT_Codigo		Código	Código do centro de trabalho (3 a 7 caracteres)
CentroDeTrabalho		CT_Nome		Centro de Trabalho	Nome do centro de trabalho (obrigatório com até 15 caracteres)
CentroDeTrabalho		CT_Desc		Descrição	Descrição do centro de trabalho (opcional)
Disponibilidade		Disp_DiaSemana		Dia da semana	Dia da semana, representado numericamente de 1 (domingo) a 7 (sábado)
Disponibilidade		Disp_Horainicio		Início	Hora de início das operações
Disponibilidade		Disp_HoraFim		Encerramento	Hora de encerramento das operações
Disponibilidade		Disp_DiaInteiro		24 h	Marcar se for turno de 24 h (independentemente dos horários marcados para início e encerramento)
Encomenda		Enc_Codigo		Código:	Código da encomenda (valor auto-incrementado pelo sistema)
Encomenda		Enc_ProdCod		Produto:	Código do produto
Encomenda		Enc_Qtde		Quantidade:	Quantidade que deve ser entregue
Encomenda		Enc_FimMax		Data de entrega:	Data e hora máxima para entrega
EncomendaInterna		Encln_Chave		Necessidade nº:	Código da necessidade gerada (valor auto-incrementado pelo sistema)
EncomendaInterna		Encln_OrdemRelacionada		Ordem Relacionada:	Código da ordem de produção relacionada
EncomendaInterna		Encln_Item		Item:	Código do item da necessidade
EncomendaInterna		Encln_Qtde		Quantidade:	Quantidade da necessidade gerada
EncomendaInterna		Encln_DataMax		Data máxima:	Data máxima para entrega (normalizada)

Tabela	Campo	Legenda	Descrição
EncomendaInterna	Encln_Ordenado	Ordenado:	Indicar se encomenda já foi transferida para Ordem de Produção
Familia	FamI_Codigo	Código	Código da família (de 2 a 4 caracteres)
Familia	FamI_Nome	Família	Nome da família (obrigatório com até 15 caracteres)
Familia	FamI_Desc	Descrição	Descrição da família (opcional)
Fila	Fila_CodRT	Recurso de Trabalho	Código do recurso de trabalho onde a ordem deve ser processada
Fila	Fila_IndiceRT	Índice	Índice do recurso de trabalho onde a ordem deve ser processada
Fila	Fila_CodOrdem	Ordem de Produção	Ordem de produção correspondente
Fila	Fila_CodOperacao	Operação	Código da operação que deve ser executada
Fila	Fila_DisponMat	Disponibilidade de materiais	Há disponibilidade de materiais para iniciar operação (elemento "vivo" na fila)
Fila	Fila_InicioOperacao	Início da operação	Data e hora (normalizados) de início da operação
Fila	Fila_ConclusaoOperacao	Conclusão da operação	Data e hora (normalizados) de conclusão da operação
Fila	Fila_Qtde	Quantidade	Quantidade que deve ser produzida
Fila	Fila_HoraChegada	Necessária Hora de chegada	Momento em que passou a haver disponibilidade de materiais (equivale a entrada do elemento na fila)
Fila	Fila_TempoSetUp	Set-up	Tempo de set-up
Fila	Fila_TempoProcesso	Tempo de processo	Tempo de processamento (não inclui set-up)
Fila	Fila_TempoCleanUp	Clean-up	Tempo de clean-up
Fila	Fila_DataMaxOP	Data limite para ordem	Data máxima para conclusão da ordem de produção correspondente
Fila	Fila_FolgaEstatica	Folga estática	Folga estática (data de entrega menos data de chegada no centro)
Fila	Fila_FolgaEstatDivTempoProcRestante	Folga estática/Tempo proc. restante	Folga estática dividida pelo tempo de processamento restante

Tabela	Campo	Legenda	Descrição
Fila	Fila_FolgaDinamica	Folga dinâmica	Folga dinâmica (período para entrega menos tempo de processamento restante)
Fila	Fila_FolgaDinDivTempoProcRestante	Folga dinâmica / Tempo proc. restante	Folga dinâmica dividida pelo tempo de processamento restante
Grandeza	Grnd_Codigo	Sigla:	Código da grandeza (1 a 4 caracteres)
Grandeza	Grnd_Nome	Grandeza:	Nome da grandeza (obrigatório com até 15 caracteres)
Grandeza	Grnd_ExpM	Expoente M:	Expoente de M na base MLT
Grandeza	Grnd_ExpL	Expoente L:	Expoente de L na base MLT
Grandeza	Grnd_ExpT	Expoente T:	Expoente de T na base MLT
Inventario_Dinamico	Inv_ItemCod	Item	Código do item (3 a 7 caracteres)
Inventario_Dinamico	Inv_Qtde	Quantidade disponível	Quantidade em estoque
Inventario_Dinamico	Inv_OrdemAlocada	Ordem alocada	Ordem de produção para a qual o estoque está alocado
Inventario_Dinamico	Inv_MomentoEntrada	Entrada em	Indica o momento de entrada no estoque (conclusão da operação que gerou)
Inventario_Dinamico	Inv_MomentoSaída	Saída em	Indica o momento de saída do estoque (início da ordem que consumirá o estoque)
Inventario_Inicial	Inv_ItemCod	Item:	Código do item (3 a 7 caracteres)
Inventario_Inicial	Inv_Serie	Série/Lote:	Série ou lote de produção (opcional com até 12 caracteres)
Inventario_Inicial	Inv_Qtde	Quantidade Disponível:	Quantidade inicialmente em estoque
Inventario_Inicial	Inv_Bloqueado	Bloqueado:	Indica se o lote está ou não bloqueado
Item	Item_Codigo	Código:	Código que identifica univocamente o item (de 3 a 7 caracteres)
Item	Item_Nome	Nome:	Nome completo do item (até 40 caracteres)
Item	Item_Desc	Descrição:	Descrição detalhada do item
Item	Item_Categoria	Categoria:	Indica se é matéria-prima, material de embalagem, semi-acabado, acabado,...
Item	Item_Familia	Família:	Família do item
Item	Item_Grandeza	Grandeza:	Grandeza de medida do item
Item	Item_Unid	Unidade de medida:	Unidade de medida

Tabela		Campo	Legenda		Descrição
Item	Item_CustoUnitInv		Custo unitário de inventário:	Custo MENSAL de estoque de uma unidade do item (incluindo custo de armazenagem, custo de capital, ...)	
LeadTime_MP	LT_CodItem		Item:	Código do item (somente matéria-prima)	
LeadTime_MP	LT_Prazo		Prazo de entrega:	Prazo de entrega da matéria-prima	
LowLevelCoding	LLC_CodItem		Item:	Código do item (de 3 a 7 caracteres)	
LowLevelCoding	LLC_Nivel		Nível:	Nível do item na classificação Low Level Coding	
Operacao	Opr_Codigo		Código da operação:	Código da operação (de 4 a 10 caracteres)	
Operacao	Opr_Nome		Operação:	Nome da operação (obrigatório com até 25 caracteres)	
Operacao	Opr_ProdSaida		Produto final:	Produto final da operação	
Operacao	Opr_Desc		Descrição:	Descrição da operação	
Operacao	Opr_RTcod		Recurso de Trabalho:	Código do recurso de trabalho utilizado na operação	
Operacao	Opr_MinLote		Lote mínimo:	Quantidade mínima que pode ser processada (lote mínimo)	
Operacao	Opr_MaxLote		Lote máximo:	Quantidade máxima que pode ser processada (lote máximo). Deixar valor 0 se não houver restrição.	
Operacao	Opr_QtdMultipla		Quantidade múltipla:	Valor cuja quantidade obrigatoriamente deve ser múltiplo. Deixar valor 0 se não houver restrição.	
Operacao	Opr_SetUpItem		Set-Up (mesmo item):	Tempo de set-up quando a operação precedente produziu o mesmo item	
Operacao	Opr_SetUpFamilia		Set-Up (mesma família):	Tempo de set-up quando a operação precedente produziu outro item, mas da mesma família	
Operacao	Opr_SetUpDiverso		Set-Up (item diverso):	Tempo de set-up quando a operação precedente produziu outro item de família diferente	
Operacao	Opr_TempoFila		Fila esperada:	Tempo médio estimado para espera em fila no recurso de trabalho (na mesma unidade dos demais tempos)	
Operacao	Opr_TempoUnitario		Tempo de processo:	Tempo unitário de processamento ou único se tempo de processo não variar com quantidade	
Operacao	Opr_CleanUp		Clean-Up:	Tempo de clean-up (após o processo)	
Operacao	Opr_TempoConst		Tempo constante:	Verdadeiro se tempo de processamento não depender da quantidade	

ANEXO E - Tabelas e campos

Tabela		Campo		Legenda		Descrição	
Operacao	Opr_TempoUnit	Unidade de tempo:	Unidade de medida dos tempos				
Operacao	Opr_EPP	EPP:	"Economic Part-Period". Quantidade x Custo Unitário de Inventário (MENSAL) que equivale a 1 set-up				
Operacao	OpEn_OpCod	Operação:	Código da operação				
OperacaoEntrada	OpEn_QtdeBase	Quantidade Base:	Quantidade base (quantidade do produto final) para as quantidades das entradas				
OperacaoEntrada	OpEn_OrdemEntrada	Ordem de entrada:	Ordem do produto de entrada (uma ordem por entrada para cada produto final)				
OperacaoEntrada	OpEn_EntradaCod	Produto de entrada:	Código do produto de entrada				
OperacaoEntrada	OpEn_QtdeEntrada	Entrada:	Quantidade necessária do produto de entrada para se obter a quantidade base				
OperacaoEntrada	Ord_Chave	Chave:	Campo chave para as ordens de produção				
OrdemDeProducao	Ord_Item	Item:	Código do item				
OrdemDeProducao	Ord_Qtde	Quantidade:	Quantidade a ser produzida				
OrdemDeProducao	Ord_ConclusaoMax	Conclusão máxima:	Data máxima (normalizada) para conclusão da ordem de produção				
OrdemDeProducao	Prog_CodRT	Recurso de Trabalho:	Código do recurso de trabalho (de 3 a 7 caracteres)				
Programa	Prog_IndiceRT	Índice:	Índice do recurso de trabalho (caso haja mais do que 1)				
Programa	Prog_OrdemProducao	Ordem de Produção:	Código da ordem de produção				
Programa	Prog_Inicio	Início:	Momento inicial da ordem de produção				
Programa	Prog_Conclusao	Conclusão:	Momento de conclusão da ordem de produção				
Programa_DataHora	PgDH_OrdemProducao	Ordem de Produção:	Código da ordem de produção				
Programa_DataHora	PgDH_Inicio	Início:	Data e hora de início da ordem de produção				
Programa_DataHora	PgDH_Conclusao	Concluído	Data e hora de conclusão da ordem de produção				
RecursoDeTrabalho	RT_Codigo	Código:	Código do recurso de trabalho (de 3 a 7 caracteres)				
RecursoDeTrabalho	RT_Nome	Recurso de Trabalho:	Nome da máquina (obrigatório com até 15 caracteres)				

ANEXO E - Tabelas e campos

Tabela		Campo		Legenda		Descrição	
RecursoDeTrabalho	RT_CT			Centro de Trabalho:		Código do centro de trabalho correspondente (opcional)	
RecursoDeTrabalho	RT_Desc			Descrição:		Descrição detalhada do recurso de trabalho	
RecursoDeTrabalho	RT_Qtde			Máquinas disponíveis:		Quantidade disponível de recursos de trabalho equivalentes	
RecursoDeTrabalho	RT_RegraSeq			Regra de Sequencição:		Regra de priorização de sequencição	
RegraSeq	Seq_Codigo			Código:		Código da regra de sequencição (2 a 4 caracteres)	
RegraSeq	Seq_Desc			Descrição:		Descrição da regra de sequencição (obrigatório com até 30 caracteres)	
RegraSeq	Seq_Input_1			Input para regra:		Indica a propriedade que deve ser maximizada ou minimizada (1º critério para seleção)	
RegraSeq	Seq_MinMax_1			Maximizar:		Verdadeiro se a prioridade for para maiores valores do input. Falso se for para menores valores.	
RegraSeq	Seq_Input_2			Input para regra:		Indica a propriedade que deve ser maximizada ou minimizada (2º critério para seleção)	
RegraSeq	Seq_MinMax_2			Maximizar:		Verdadeiro se a prioridade for para maiores valores do input. Falso se for para menores valores.	
RelacionamentoDeOrdens	Rel_OrdemCriada			Ordem gerada:		Ordem de Produção cuja produção será alocada	
RelacionamentoDeOrdens	Rel_OrdemDeReferencia			Ordem de Referência:		Ordem de Produção de destino para as saídas da ordem principal	
RelacionamentoDeOrdens	Rel_QtdeAlocada			Quantidade:		Quantidade alocada da ordem principal para a ordem de referência	
Unidade	Unid_Sigla			Sigla:		Sigla da unidade de medida. Este campo também serve como chave, portanto não pode se repetir.	
Unidade	Unid_Nome			Unidade:		Nome da unidade	
Unidade	Unid_GrндCod			Grandeza:		Grandeza relativa à unidade	
Unidade	Unid_Padrao			Unidade Padrão:		Indicar se é a unidade padrão da grandeza	
Unidade	Unid_EquivPadrao			Equivalência Padrão:		Relação com a unidade padrão	
Verificacao	Vrf_Tabela			Tabela:		Nome da tabela onde deve ser feita a correção/atualização	

ANEXO E - Tabelas e campos

Tabela		Legenda		Descrição
	Campo			
Verificacao	Vrf_CampoChave	Campo chave:	Nome do campo-chave da tabela onde deve ser feita a correção/atualização ou referência para a atualização	
Verificacao	Vrf_RegistroChave	Registro:	Chave do registro onde foi encontrado erro	
Verificacao	Vrf_RegistroDesc	Complemento:	Complemento do registro onde foi encontrado erro	
Verificacao	Vrf_Erro	Descrição do erro:	Descrição do erro detectado	

Consulta	Instrução SQL
qry_ApagarProv_TempoSetUp	DROP TABLE prov_TempoSetUp
qry_AtualizarDisponibilidadeDeMateriais	UPDATE Fila SET Fila_Fila_DisponMat = True, Fila_Fila_HoraChegada = [Momento] WHERE (((Fila_Fila_DisponMat)=False) AND ((Fila_Fila_InicioOperacao) Is Null) AND ((Fila_Fila_CodOrdem) In (SELECT Ord_Chave FROM qry_VerificarDisponibilidadeDeMateriais_3 UPDATE LowLevelCoding SET LLC_Nivel = [Nivel] + 1 WHERE LLC_CodItem IN (SELECT DISTINCT OperacaoEntrada.OpEn_EntradaCod FROM (LowLevelCoding INNER JOIN Operacao ON LowLevelCoding.LLC_CodItem = Operacao.Opr_ProdSaida) INNER JOIN Operaca UPDATE Fila SET Fila_Fila_FolgaEstatica = [Fila_DataMaxOP]-[Fila_HoraChegada], Fila_Fila_FolgaDinamica = [Fila_DataMaxOP]-[Fila_TempoSetUp]-[Fila_TempoProcesso]- [Fila_TempoCleanUp]-[Momento], Fila_Fila_FolgaEstatDivTempoProcRestante = ([Fila_DataMaxOP]-[F SELECT Max(OrdemDeProducao.Ord_Chave) AS UltimaOrdem FROM OrdemDeProducao; UPDATE Inventario_Dinamico SET Inventario_Dinamico.Inv_MomentoSaida = [Momento] WHERE (((Inventario_Dinamico.Inv_OrdemAlocada)=[Ordem de Producao])); SELECT Count(*) FROM Fila WHERE Fila_InicioOperacao = Null; SELECT Count(*) AS NumeroDeOrdens FROM Fila; UPDATE Fila SET Fila_TempoSetUp = [Tempo Corrigido] WHERE Fila_CodOrdem = [Ordem de Producao]; SELECT Encln_Chave, Encln_DataMax FROM EncomendaInterna; SELECT Fila_Fila_CodOrdem, Programa.Prog_CodRT AS RTNecessario, Programa.Prog_IndiceRT AS IndiceDoRTNecessario, Max(Programa.Prog_Conclusao) AS Liberado FROM (Programa RIGHT JOIN Fila ON Programa.Prog_CodRT = Fila_Fila_CodRT) LEFT JOIN OrdemDeProducao ON
qry_AtualizarLLC	
qry_AtualizarPropriedadesDinamicas	
qry_CodigoUltimaOrdem	
qry_ConsumirInventario	
qry_ContarOrdensNaoRealizadas	
qry_ContarTotalDeOrdens	
qry_CorrigerTempoSetUp	
qry_DataEncomendaInterna	
qry_DefinirSetUpDinamico_1	

Consulta	Instrução SQL
qry_DefinirSetUpDinamico_2	SELECT qry_DefinirSetUpDinamico_1.Fila_CodOrdem, qry_DefinirSetUpDinamico_1.RTNecessario, qry_DefinirSetUpDinamico_1.IndiceDoRTNecessario, Programa.Prog_OrdemProducao, OrdemDeProducao.Ord_Item AS UltimoItemProduzido, Item.Item_Familia AS FamiliaUltimoItem SELECT Fila.Fila_CodOrdem, Min(qry_DefinirSetUpDinamico_2.SetUp) AS SetUpMinimo INTO prov_TempoSetUp FROM Fila LEFT JOIN qry_DefinirSetUpDinamico_2 ON Fila.Fila_CodOrdem = qry_DefinirSetUpDinamico_2.Fila_CodOrdem GROUP BY Fila.Fila_CodOrdem; UPDATE Fila INNER JOIN prov_TempoSetUp ON Fila.Fila_CodOrdem = prov_TempoSetUp.Fila_CodOrdem SET Fila.Fila_TempoSetUp = [SetUpMinimo]; UPDATE EncomendaInterna SET EncomendaInterna.Encln_Qtde = [Encln_Qtde]-[Desconto] WHERE (((EncomendaInterna.Encln_Chave)=[Encomenda])); SELECT Disponibilidade.Disp_Horainicio, Disponibilidade.Disp_HoraFim, Disponibilidade.Disp_DiaInteiro, Disponibilidade.Disp_DiaSemana FROM Disponibilidade ORDER BY Disponibilidade.Disp_DiaSemana; SELECT DISTINCT (SELECT Disp_Horainicio FROM Disponibilidade WHERE Disp_DiaSemana = 1) AS DomIni, (SELECT Disp_Horainicio FROM Disponibilidade WHERE Disp_DiaSemana = 2) AS SegIni, (SELECT Disp_Horainicio FROM Disponibilidade WHERE Disp_DiaSemana = 3) AS T SELECT OrdemDeProducao.Ord_Chave, OrdemDeProducao.Ord_Item, OrdemDeProducao.Ord_Qtde, OrdemDeProducao.Ord_ConclusaoMax, ConverterTempoParaHora([Opr_TempoFila]+[Opr_SetUpDiverso]+[Opr_TempoUnitario])*If([Opr_TempoConst],1,[Ord_Qtde])+[Opr_CleanUp],[Opr_Tem DELETE EncomendaInterna.Encln_Chave FROM EncomendaInterna WHERE (((EncomendaInterna.Encln_Chave)=[Encomenda])); SELECT 7/30/(24*Sum(If([Disp_DiaInteiro]=True,1,[Disp_HoraFim]-[Disp_Horainicio]))) AS Fator FROM Disponibilidade; INSERT INTO EncomendaInterna (Encln_OrdemRelacionada, Encln_Item, Encln_Qtde, Encln_DataMax, Encln_Ordenado) SELECT -1*[Enc_Codigo] AS OrdemRelacionada, Encomenda.Enc_ProdCod, Encomenda.Enc_Qtde, Encomenda.Enc_FimMax, False AS EncomendaOrdenada FROM E
qry_DefinirSetUpDinamico_3	
qry_DefinirSetUpDinamico_4	
qry_DescontarInventariolInicial	
qry_DisponibilidadeDiaria	
qry_DisponibilidadeDiaria_2	
qry_EncomendaInterna	
qry_ExcluirEncomendaPorInventario	
qry_FatorDeConversaoMensal	
qry_GerarEncomendaInternaDireta	

Consulta	Instrução SQL
qry_GerarEncomendaInternaIndireta	<pre> INSERT INTO EncomendaInterna (Encln_OrdemRelacionada, Encln_Item, Encln_Qtde, Encln_DataMax) SELECT qry_NecessidadePorNivelLLC.Ord_Chave, qry_NecessidadePorNivelLLC.LLC_CodItem, qry_NecessidadePorNivelLLC.Quantidade, qry_NecessidadePorNivelLLC.[Conclus qry_NecessidadePorNivelLLC.DataHora (PgDH_OrdemProducao, PgDH_Inicio, PgDH_Conclusao) INSERT INTO Programa_DataHora (Prog_OrdemProducao, SELECT qry_RelatPrograma_Auxiliar.Prog_OrdemProducao, CDate((InicioSemana))+((SemanasNoInicio)*7+IIf((RestoInicio)=0,[MomentoInicialUtili],IIf((RestoInicio)<= [SegAcum],[DomI INSERT INTO LowLevelCoding (LLC_CodItem, LLC_Nivel) SELECT Item_Codigo, 0 FROM Item; SELECT Count(*) AS [nº dias] FROM Disponibilidade WHERE (((Disponibilidade.Disp_HoraInicio)<=[Disp_HoraFim])) OR (((Disponibilidade.Disp_DiaInteiro)=True)); SELECT Count(*) AS [nº encomendas] FROM Encomenda; SELECT OperacaoEntrada.OpEn_OpCod, Operacao.Opr_Nome, [Opr_ProdSaida] & " - " & [Item_Nome] AS Item FROM (Item LEFT JOIN Operacao ON Item.Item_Codigo = Operacao.Opr_ProdSaida) LEFT JOIN OperacaoEntrada ON Operacao.Opr_Codigo = OperacaoEntrada.OpEn_OpCod SELECT Operacao.Opr_Codigo, Operacao.Opr_Nome, Operacao.Opr_MinLote, Operacao.Opr_MaxLote, Operacao.Opr_QtdeMultiplo FROM Operacao WHERE (((Operacao.Opr_MinLote)>[Opr_MaxLote]) AND ((Operacao.Opr_MaxLote)<=0)) OR (((Operacao.Opr_MaxLote)<[Opr_QtdeMultip SELECT Item.Item_Codigo, Item.Item_Nome, Item.Item_Desc FROM Categoria LEFT JOIN Item ON Categoria.Catg_Codigo = Item.Item_Categoria WHERE (((Item.Item_Codigo) Not In (SELECT Opr_ProdSaida FROM Operacao)) AND ((Categoria.Catg_Origem)=False)); </pre>
qry_GerarProgramaDataHora	
qry_IniciarLLC_Nivel_0	
qry_InicVerif_ContarDisponibilidade	
qry_InicVerif_ContarEncomendas	
qry_InicVerif_ListarEntradaIgualSaida	
qry_InicVerif_ListarIncoerenciaLote	
qry_InicVerif_ListarItemSemOperacao	

Consulta	Instrução SQL
qry_InicVerif_ListarOperacaoSemEntrada	SELECT Operacao.Opr_Codigo, Operacao.Opr_Nome, Operacao.Opr_ProdSaida, Item.Item_Nome, Operacao.Opr_Desc FROM Categoria LEFT JOIN (Item LEFT JOIN (Operacao LEFT JOIN OperacaoEntrada ON Operacao.Opr_Codigo = OperacaoEntrada.OpEn_OpCod) ON Item.Item_Codigo SELECT qry_InicVerif_ListarRepeticaoOrdemEntrada_Pre.OpEn_OpCod, Operacao.Opr_Nome, qry_InicVerif_ListarRepeticaoOrdemEntrada_Pre.OpEn_OrdemEntrada, qry_InicVerif_ListarRepeticaoOrdemEntrada_Pre.NumEntradas FROM qry_InicVerif_ListarRepeticaoOrdemEntrada_Pre SELECT OperacaoEntrada.OpEn_OpCod, OperacaoEntrada.OpEn_OrdemEntrada, Count(*) AS NumEntradas FROM OperacaoEntrada GROUP BY OperacaoEntrada.OpEn_OpCod, OperacaoEntrada.OpEn_OrdemEntrada HAVING (((Count(*))>1)) ORDER BY OperacaoEntrada.OpEn_OpCod;
qry_InserirInventariolnicial	INSERT INTO Inventario_Dinamico (Inv_ItemCod, Inv_Qtde, Inv_OrdemAlocada, Inv_MomentoEntrada) SELECT [Item] AS [Item inserido], [Quantidade] AS Qtde, [Ordem Alocada] AS Ordem, [Momento] AS
qry_InserirMensagemDeErro	INSERT INTO Verificacao (Vrf_Tabela, Vrf_CampoChave, Vrf_RegistroChave, Vrf_RegistroDesc, Vrf_Erro) SELECT [Tabela], [CampoChave], [RegistroChave], [DescricaoDoRegistro], [MensagemDeErro];
qry_InserirMP_Prazos	INSERT INTO LeadTime_MP (LT_CodItem, LT_Prazo) SELECT Item.Item_Codigo, 0 AS Prazo FROM Item INNER JOIN Categoria ON Item.Item_Categoria = Categoria.Catg_Codigo WHERE (((Item.Item_Codigo) Not In (SELECT LT_CodItem FROM LeadTime_MP)) AND ((Categoria C
qry_InserirOrdemDeProducao	INSERT INTO OrdemDeProducao (Ord_Item, Ord_Qtde, Ord_ConclusaoMax) SELECT [Item] AS ItemInserido, [Quantidade] AS Qtde, [Conclusao] AS DataMax;
qry_InserirProducaoInventario	INSERT INTO Inventario_Dinamico (Inv_ItemCod, Inv_MomentoEntrada, Inv_OrdemAlocada, Inv_Qtde) SELECT Operacao.Opr_ProdSaida, Fila.Fila_ConclusaoOperacao, RelacionamentoDeOrdens.Rel_QtdeAlocada

Consulta	Instrução SQL
qry_InserirPrograma	<pre> INSERT INTO Programa (Prog_CodRT, Prog_IndiceRT, Prog_OrdemProducao, Prog_Inicio, Prog_Conclusao) SELECT Fila.Fila_CodRT, Fila.Fila_IndiceRT, Fila.Fila_CodOrdem, Fila.Fila_InicioOperacao, Fila.Fila_ConclusaoOperacao FROM Fila WHERE (((Fila.Fila_CodOr INSERT INTO RelacionamentoDeOrdens (Rel_OrdemCriada, Rel_OrdemDeReferencia, Rel_QtdeAlocada) SELECT [Ordem criada] AS Ordem, [Ordem de referencia] AS Referencia, [Quantidade] AS QuantidadeAlocada; SELECT EncomendaInterna.Encln_Chave, EncomendaInterna.Encln_OrdemRelacionada, EncomendaInterna.Encln_Item, EncomendaInterna.Encln_DataMax, EncomendaInterna.Encln_Qtde, Sum(Inventario_Inicial.Inv_Qtde) AS [Inventario Inicial] FROM EncomendaInterna LEFT JO SELECT EncomendaInterna.Encln_Chave, EncomendaInterna.Encln_Qtde, IIf([Opr_QtdeMultiplo]=0,[Opr_MaxLote],IIf(Cint([Opr_MaxLote]/[Opr_QtdeMultiplo])>([Opr_MaxLote]/[O IIf([Opr_QtdeMultiplo]*[Opr_QtdeMultiplo]>Cint([Opr_MaxLote]/[Opr_QtdeMultiplo]-1),[Opr_QtdeM pr_QtdeMultiplo]),[Opr_QtdeMultiplo]) AS Encln_Item, Encln_Qtde, INSERT INTO EncomendaInterna (Encln_OrdemRelacionada, Encln_Item, Encln_Qtde, Encln_DataMax, Encln_Ordenado) SELECT EncomendaInterna.Encln_OrdemRelacionada, EncomendaInterna.Encln_D qry_LimitarEncomendaInterna_1 qry_LimitarEncomendaInterna_2 qry_LimitarEncomendaInterna_3 DELETE * FROM Disponibilidade; DELETE * FROM EncomendaInterna; DELETE * FROM Fila; DELETE * FROM Inventario_Dinamico; </pre>
qry_InserirRelacionamentoDeOrdem	
qry_InventarioInicial	
qry_LimitarEncomendaInterna_1	
qry_LimitarEncomendaInterna_2	
qry_LimitarEncomendaInterna_3	
qry_LimparDisponibilidade	
qry_LimparEncomendaInterna	
qry_LimparFila	
qry_LimparInventarioDinamico	

ANEXO E - Consultas

	Instrução SQL
Consulta	DELETE *
qry_LimparLLC	FROM LowLevelCoding;
qry_LimparOrdemDeProducao	DELETE * FROM OrdemDeProducao;
qry_LimparPrograma	DELETE * FROM Programa;
qry_LimparProgramaDataHora	DELETE * FROM Programa_DataHora;
qry_LimparRelacionamentoDeOrdens	DELETE * FROM RelacionamentoDeOrdens;
qry_LimparVerificacao	DELETE * FROM Verificacao;
qry_MarcarEncomendaOrdenada	UPDATE EncomendaInterna INNER JOIN LowLevelCoding ON EncomendaInterna.Encln_Item = LowLevelCoding.LLC_CodItem SET EncomendaInterna.Encln_Ordenado = True (LowLevelCoding.LLC_CodItem SET EncomendaInterna.Encln_Ordenado)=[False] AND ((LowLevelCoding.LLC_Nivel)=[Nivel])); WHERE (((EncomendaInterna.Encln_Ord_Chave, LowLevelCoding.LLC_CodItem, SELECT qry_EncomendaInterna.[OpEn_QtdeEntrada] AS Quantidade, [Ord_Qtde]/[OpEn_QtdeBase]*[OpEn_QtdeEntrada]) AS Quantidade, ConvertirTempoParaHora([Operacao_1].[Opr_TempoFila],[Operacao_1].[Opr_TempoUnid]) AS [Tempo de fila], ConvertirTempoParaHora([O de fila], ConvertirTempoParaHora([O SELECT EncomendaInterna.Encln_Chave, EncomendaInterna.Encln_OrdemRelacionada, EncomendaInterna.Encln_Item, EncomendaInterna.Encln_Qtde, EncomendaInterna.Encln_DataMax, EncomendaInterna.Encln_MinLote, Operacao.Opr_MaxLote, O Operacao.Opr_EPP, Item.Item_CustoUnitInv, Operacao.Opr_MinLote, Operacao.Opr_MaxLote, O UPDATE Fila SET Fila.Fila_InicioOperacao = [Momento], Fila.Fila_IndexeRT = [Indice do RT], Fila.Fila_ConclusaoOperacao = CSng([Momento]+[Fila_TempoSetUp]+[Fila_TempoProcesso]+[Fila_TempoCleanUp]) WHERE (((Fila.Fila_CodOrdem)=[Ordem de Producao])); SELECT qry_RecursosComFila.Fila_Fila_CodRT, qry_RecursosComFila.RT_Qtde, qry_RecursoOcupado.QuantidadeOcupada, If([QuantidadeOcupada] Is Not Null,[RT_Qtde]- [QuantidadeOcupada],[RT_Qtde]) AS RecursoDisponivel, qry_RecursosComFila.CampoDeSequenciacao_1, qry_Rec
qry_NecessidadePorNivelLLC	
qry_NecessidadesAbertas	
qry_RealizarOperacao	
qry_RecursoDisponivel	

Consulta	Instrução SQL
qry_RecursoOcupado	<pre>SELECT Fila.Fila_CodRT, Count(*) AS QuantidadeOcupada FROM Fila WHERE (((Fila.Fila_InicioOperacao)<=[Momento]) AND (Fila.Fila_ConclusaoOperacao)>[Momento])) GROUP BY Fila.Fila_CodRT;</pre>
qry_RecursoComFila	<pre>SELECT Fila.Fila_CodRT, RecursoDeTrabalho.RT_Qtde, RegraSeq.Seq_Input_1 AS CampoDeSequenciacao_1, If([Seq_MinMax_1]=True,"DESC","ASC") AS OrdemDeSequenciacao_1, RegraSeq.Seq_Input_2 AS CampoDeSequenciacao_2, If([Seq_MinMax_2]=True,"DESC","ASC") AS Ordem</pre>
qry_RelatDatasFinais	<pre>SELECT Encomenda.Enc_Codigo, Programa_DataHora.PgDH_OrdemProducao, [EncIn_Item] & " - " & [Item_Nome] AS Item, EncomendaInterna.EncIn_Qtde, Encomenda.Enc_FimMax, Programa_DataHora.PgDH_Conclusao, If([PgDH_Conclusao]>[Enc_FimMax],[PgDH_Conclusao]- [Enc_Fim</pre>
qry_RelatNecessidadesMP	<pre>SELECT [OpEn_EntradaCod] & " - " & [Item_Nome] AS Item, DateValue([PgDH_Inicio]) AS Necessidade, LeadTime_MP.LT_Prazo, DateValue([PgDH_Inicio])-[LT_Prazo] AS Compra, Sum([Ord_Qtde]*[OpEn_QtdeEntrada])/([OpEn_QtdeBase]) AS Quantidade</pre>
qry_RelatOcupacao	<pre>FROM ([Programa_Dat SELECT [RT_CT] & " - " & [CT_Nome] AS CT, [Prog_CodRT] & " - " & [RT_Nome] & " (" & [Prog_IndiceRT] & ")") AS RT, Min([Programa.Prog_Inicio] AS InicioLocal, Max([Programa.Prog_Conclusao] AS ConclusaoLocal, Sum([Prog_Conclusao]-[Prog_Inicio]) AS TempoEmUso, [T</pre>
qry_RelatPrograma_Auxiliar	<pre>SELECT Programa.Prog_OrdemProducao, Programa.Prog_Inicio, Programa.Prog_Conclusao, If([Prog_Inicio/SemanaCompleta]<1,0,If(Cint([Prog_Inicio/SemanaCompleta])<=(Prog_Inicio/SemanaCo mpleta),Cint([Prog_Inicio/SemanaCompleta],Cint([Prog_Inicio/SemanaCompleta])-1)</pre>
qry_RelatPrograma_DataInicio	<pre>SELECT DateValue([PgDH_Inicio]) AS Data, ([Prog_CodRT] & " - " & [RT_Nome] & " (" & [Prog_IndiceRT] & ")") AS [Recurso de Trabalho], Programa_DataHora.PgDH_OrdemProducao, ([Ord_Item] & " - " & [Item_Nome]) AS Item, OrdemDeProducao.Ord_Qtde, [Item_Nome]) AS</pre>
qry_RelatPrograma_Item	<pre>SELECT ([Item_Familia] & " - " & [FamI_Nome]) AS Familia, ([Ord_Item] & " - " & [Item_Nome]) AS Item, Programa_DataHora.PgDH_OrdemProducao, OrdemDeProducao.Ord_Qtde, Programa_DataHora.PgDH_Inicio, Programa_DataHora.PgDH_Conclusao</pre>
qry_RelatPrograma_RT	<pre>FROM Programa_DataHora I SELECT ([RT_CT] & " - " & [CT_Nome]) AS [Centro de Trabalho], ([Prog_CodRT] & " - " & [RT_Nome] & " (" & [Prog_IndiceRT] & ")") AS [Recurso de Trabalho], Programa_DataHora.PgDH_OrdemProducao, OrdemDeProducao.Ord_Item, Item.Item_Nome,</pre>

ANEXO E - Consultas

Consulta	Instrução SQL
qry_RelatTempoFluxo	<pre>SELECT [CT_Codigo] & " - " & [CT_Nome] AS CT, [Fila_CodRT] & " - " & [RT_Nome] & " (" & [Fila_IndiceRT] & " " AS RT, Avg((([Fila_InicioOperacao]-[Fila_HoraChegada])/1000) AS Espera, Avg(((Fila_ConclusaoOperacao)-[Fila_InicioOperacao])/1000) AS TempoOperac INSERT INTO fila (Fila_CodRT, Fila_CodOrdem, Fila_CodOperacao, Fila_Qtde, Fila_DisponMat, Fila_HoraChegada, Fila_TempoProcesso, Fila_TempoCleanUp, Fila_DataMaxOP) SELECT Operacao.Opr_RTCod, OrdemDeProducao.Ord_Chave, Operacao.Opr_Codigo, OrdemDeProducao SELECT OrdemDeProducao.Ord_Chave, OrdemDeProducao.Ord_Item, OperacaoEntrada.OpEn_EntradaCod, [Ord_Qtde]/[OpEn_QtdeBase]*[OpEn_QtdeEntrada] AS Necessidade, Categoria.Catg_Origem FROM (Categoria INNER JOIN Item ON Categoria.Catg_Codigo = Item.Item_Categori SELECT qry_VerificarDisponibilidadeDeMateriais_1.Ord_Chave, qry_VerificarDisponibilidadeDeMateriais_1.OpEn_EntradaCod, qry_VerificarDisponibilidadeDeMateriais_1.Necessidade, qry_VerificarDisponibilidadeDeMateriais_1.Catg_Origem, Sum(Inventario_Dinamico.In qry_VerificarDisponibilidadeDeMateriais_2.Ord_Chave, Not SELECT qry_VerificarDisponibilidadeDeMateriais_2 AS OrdemDisponivel (CBool(Sum((MaterialDisponivel)))) AS OrdemDisponivel FROM qry_VerificarDisponibilidadeDeMateriais_2 GROUP BY qry_VerificarDisponibilidadeDeMateriais_2.Ord_Chave, SELECT Programa.Prog_CodRT, Programa.Prog_IndiceRT, Max(Programa.Prog_Inicio) AS MaxDeProg_Inicio, Max(Programa.Prog_Conclusao) AS MaxDeProg_Conclusao FROM (RecursoDeTrabalho INNER JOIN (OrdemDeProducao AS OrdemDeProducao_1 INNER JOIN Operacao AS Operaca SELECT Programa.Prog_CodRT, Programa.Prog_IndiceRT, Max(Programa.Prog_Inicio), MaxDeProg_Inicio, Max(Programa.Prog_Conclusao) AS MaxDeProg_Conclusao, OrdemDeProducao_1.Ord_Item, Min(IIf([Item].[Item_Codigo]=[Item_1].[Item_Codigo],[Operacao_1].[Opr_SetUp SELECT Min([Fila_ConclusaoOperacao]) AS ProximoMomento FROM Fila WHERE ((([Fila.Fila_ConclusaoOperacao]>[Ultimo Momento])));</pre>
qry_TransferirOrdemParaFila	
qry_VerificarDisponibilidadeDeMateriais_1	
qry_VerificarDisponibilidadeDeMateriais_2	
qry_VerificarDisponibilidadeDeMateriais_3	
qry_VerificarIndiceRT_EmUso	
qry_VerificarIndiceRT_Usados	
qry_VerificarProximoMomento	

Consulta	Instrução SQL
qry_VerificarSetUpReal	SELECT qry_DefinirSetUpDinamico_2.Fila_CodOrdem, qry_DefinirSetUpDinamico_2.IndiceDoRTNecessario, If([IndiceDoRTNecessario] = [Indice do Recurso] Or [IndiceDoRTNecessario] Is Null, [SetUp], ConverterTempoParaHora([Opr_SetUpDiverso], [Opr_TempoUnid])) AS SetUp