

ESCOLA POLITÉCNICA DA U.S.P.

ENGENHARIA MECÂNICA

PMC 581

PROGRAMAÇÃO DO MÉTODO DE CRANK-NICHOLSON

Serviço de Bibliotecas
Biblioteca de Engenharia Mecânica

CURITIBA

DEZEMBRO DE 1991

JOÃO FERNANDES ALVES NETO
PROGRAMAÇÃO DO MÉTODO DE CRANK-NICHOLSON

Trabalho final da disciplina
PMC 581, do curso de Engenharia
Mecânica (03040), sob orientação do
prof. Dr. Marcos de Mattos Pimenta
para obtenção de grau de
Engenheiro.

São Paulo
~~CURITIBA~~

DEZEMBRO DE 1991



Agradecimentos

Ao meu orientador

Prof. Doutor Marcos de Mattos Pimenta

e à todos de quem pude contar um sincero apoio

Sumário

NOTAÇÃO UTILIZADA, 1

1. Proposta do Trabalho, 2

2. Introdução à Transferência de Calor, 4

2.1. Histórico, 4

2.2 Mecanismos, 5

2.2.1 Condução, 5

2.2.2. Convecção, 6

2.2.3. Radiação, 7

2.3. Complemento, 8

2.3.1. Dedução da Equação da Condução de Calor

Bidimensional, 10

2.3.2. Descrição do Método de Crank-Nicholson, 11

3. Programação, 13

3.1. TELA1, 14

3.2. TELA2, 16

3.3. CA1, 20

3.3.1. Módulo Normalização das Arestas da Figura, 21

3.3.2. Módulo Retas da Figura, 23

3.3.3. Módulo Pontos de Cruzamento, 24

3.3.4. Módulo Descarte dos pts de cruzamento falsos, 26

3.3.5. Módulo Ordenamento dos Pontos de Cruzamento & Módulo Descarte dos Pontos de cruzamento Repetidos, 28

3.3.6. Módulo Pontos Internos, 32

3.3.7. Módulo Internos, 40

3.3.8. Módulo Arestas,	42
3.3.9. Módulo Periféricos,	43
3.3.10. Módulo Externos,	45
3.3.11. Módulo Estrutura na tela,	47
3.3.12. Módulo Subrotina "Coloca Pontos",	49
3.3.13. Módulo Subrotina "É linha de contornoΣ",	50
3.4. CA2,	52
3.4.1. Módulo Definições Iniciais,	53
3.4.2. Módulo Entrada de dados,	54
3.4.3. Módulo Impressão dos dados do mód. anterior,	57
3.4.4. Módulo H e T por face,	58
3.4.5. Módulo Correção da variável TIPO\$(I,J),	61
3.4.6. Módulo Construção da tabela de caminhos,	65
3.4.7. Núcleo do Método de Crank-Nicholson,	67
3.4.8. Anexos do módulo anterior,	76
4. Como utilizar o programa,	78
5. Conclusão Final,	86
6. Bibliografia,	88

NOTAÇÃO UTILIZADA

∂	derivada parcial
A	Area, m^2
α	indica proporcionalidade
CE	Calor específico, $kJ/kg^\circ C$
Δt	intervalo de tempo entre duas iterações consecutivas
H	Coeficiente de transferência de calor, $W/m^2^\circ C$
K	Condutividade térmica, $W/m^\circ C$
q	Fluxo de calor, KJ por unidade de tempo
R_o	Densidade, kg/m^3
Σ	Somatória
T	Temperatura, $^\circ C$
t	Tempo, s
X,Y	coordenadas do sistema cartesiano bidimensional
x,y	coordenadas do sistema cartesiano bidimensional

INDICES

SUPERFICIE	Avaliado nas condições de parede
ω	Avaliado nas condições de escoamento livre

1. Proposta do Trabalho

Ao término deste trabalho, teremos em mãos um programa de computador baseado no método de Crank-Nicholson.

Com ele poderemos solucionar problemas de transferência de calor em regime transitório, para figuras bidimensionais, aonde ocorram condução de calor com ou sem geração interna de calor, tendo como condições de contorno fronteiras à temperaturas constantes ou submetidas ao processo de convecção de calor.

O Método de Crank-Nicholson nos fornecerá como saída uma matriz que será resolvida pelo Método de Gauss-Siedel; ainda, para implementá-lo, discretizaremos a figura de nosso interesse na forma de uma malha de pontos.

No desenvolvimento deste trabalho, algumas limitações operacionais foram impostas para que o mesmo fosse factível. São elas:

1. A figura proposta deverá consistir de retas verticais e horizontais.
2. As coordenadas das arestas da figura deverão coincidir com os pontos da malha definida pelo número de divisões em x e y .
3. As coordenadas das arestas deverão ser positivas diferentes de zero.

4. As coordenadas das arestas deverão ser informadas ao programa em uma sequência que siga o sentido horário.

5. As Figuras propostas não poderão ter duas ou mais faces sem nenhum ponto interno a separá-las, ou seja, na prática, não são permitidas figuras nas quais, em uma das direções, tenhamos definido o número de divisões igual a um.

2. Introdução à Transferência de Calor

2.1. Histórico

Talvez desde os primórdios de nossa história o homem se pergunte o que é o fogo, porque ele queima, como ele aquece.

Desta curiosidade de nossos ancestrais aos dias de hoje, personagens famosos tem debatido este tema.

Desde Newton, que em 1701 afirmou ser a velocidade de resfriamento de um corpo proporcional a diferença de temperatura entre o corpo e os arredores, passando por Fahrenheit, o primeiro a descrever um termômetro de mercúrio em 1724, e que deu seu nome a escala de temperatura Fahrenheit; Franklin, autor de uma série de observações a respeito da condutibilidade térmica de materiais diversos, metais, cerâmicas, madeira, em particular aplicados a bules de chá; Scheele (1724-1786), que identificou um dos três processos de transferência de calor, a radiação, ao notar que o calor de uma lareira é capaz de atravessar o ar, praticamente sem aquecê-lo, mas sendo absorvido por um objeto de vidro, vimos a evolução de uma série de princípios e leis cujo conjunto resultou na atual disciplina Transferência de Calor.

Atualmente definimos o transporte de calor como sendo o transporte de energia que ocorre devido ao desequilíbrio de temperaturas.

2.2 Mecanismos

São identificados três mecanismos pelos quais ocorre a transferência de calor:

2.2.1 Condução

Aonde o calor é conduzido pelo transporte de energia de movimento e choques entre moléculas adjacentes, este movimento varia desde a vibração dos átomos em uma estrutura cristalina até o movimento randômico das moléculas ou átomos de um gás, ou o transporte de energia por elétrons livres em sólidos metálicos.

Em sólidos o transporte por elétrons é muito mais efetivo do que a vibração de átomos em uma estrutura cristalina sendo este o motivo pelo qual bons condutores elétricos são geralmente bons condutores de calor, e isolantes elétricos bons isolantes térmicos.

Podemos quantificar a condução de calor através da Lei de Fourier da condução de calor:

" A taxa de transferência de calor, por unidade de área, em um corpo, é proporcional ao gradiente de temperatura na direção normal à área".

Ou seja: $\frac{q}{A} \propto \frac{\partial T}{\partial x} \quad (1)$

Acrescentando uma constante de proporcionalidade (K) positiva, e lembrando que a condução de calor ocorre das temperaturas altas para as baixas ($\partial T / \partial x$ é negativo), podemos escrever :

$$q = - K A \frac{\partial T}{\partial x} \quad (2)$$

, onde q : fluxo de calor na direção x (W).

K : condutividade térmica do material (W/mK).

A : área transversal a x (m²).

T : temperatura (K).

2.2.2. Convecção

Aqui o calor é transportado junto a porções macroscópicas de elementos de líquidos e gás, portanto a convecção de calor é regida pela Mecânica dos Fluidos.

Podemos caracterizá-la em convecção natural, se é induzida por diferenças de densidades devido a diferenças de temperaturas, e em convecção forçada, quando o movimento do fluido é resultante da aplicação de forças externas.

De uma forma bastante simples , Newton propos que a taxa de transferência de calor por unidade de área fosse relacionada à diferença entre a temperatura da superfície e a temperatura do fluido, então:

$$\frac{q}{A} \propto (T_{\text{SUPERFICIE}} - T_{\infty}) \quad (3)$$

Adotando um coeficiente de transferência de calor por convecção H , podemos reescrever:

$$q = H A (T_{\text{SUPERFICIE}} - T_{\infty}) \quad (4)$$

onde q : fluxo de calor por convecção (W).

H : coeficiente de transferência de calor por convecção ($\text{W/m}^2\text{K}$).

A : área superficial (m^2).

$T_{\text{SUPERFICIE}}$: temperatura da superfície em contato com o fluido (K).

T_{∞} : temperatura do fluido (K).

2.2.3. Radiação

O calor é transportado através de ondas eletromagnéticas. Embora não seja considerada no trabalho aqui apresentado, podemos dizer que é a forma primordial de transporte de calor quando lembramos que é o meio pelo qual o Sol transfere uma ínfima parcela de sua energia ao nosso planeta (admite-se uma taxa de transporte de 1395 W/m^2 na superfície da terra).

2.3. Complemento

Um aspecto interessante de ser aqui mencionado é o que diz respeito à complementaridade entre o estudo da Termodinâmica Clássica e o estudo da Transferência de Calor.

Embora as duas envolvam o estudo de fenômenos de calor, a Termodinâmica está interessada no aspecto quantitativo do problema, enquanto que a Transferência de Calor nos fornece principalmente as taxas com que estes fenômenos ocorrem, e portanto os tempos .

Para resolver os diversos problemas possíveis na área de condução de calor, foram desenvolvidas diversas abordagens teóricas e numéricas .

Para o caso específico de condução em regime transitório, podemos citar:

(métodos analíticos)

- dos Sistemas Concentrados
- Cartas de Heissler-

Sólidos Multidimensionais Simples

(métodos numéricos)

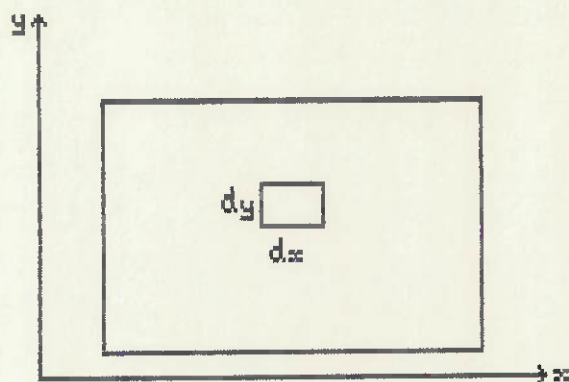
- de Euler Explícito
- de Euler Implícito
- de Crank-Nicholson

Entre todos, nos interessa o último, Método de Crank-Nicholson.

Para compreendermos a estrutura deste método, mostraremos a seguir a dedução da Equação da Condução de Calor Bidimensional, para em seguida passarmos ao método propriamente dito.

2.3.1. Dedução da Equação da Condução de Calor Bidimensional

FIGURA 1



Tomando um elemento de área dA (FIGURA 1), podemos escrever o seguinte balanço de energia:

A ENERGIA QUE ENTRA MAIS A ENERGIA GERADA INTERNAMENTE É IGUAL A ENERGIA QUE SAI MAIS A VARIAÇÃO DA ENERGIA INTERNA

onde

$$\text{ENERGIA QUE ENTRA} = q_x + q_y = -K \, dy \, \frac{\partial T}{\partial x} - K \, dx \, \frac{\partial T}{\partial y}$$

$$\text{ENERGIA GERADA INTERNAMENTE} = q_{\text{GER}} \, dx \, dy$$

$$\text{ENERGIA QUE SAI} = q_{x+dx} + q_{y+dy} =$$

$$= \left[K \, \frac{\partial T}{\partial x} + \frac{\partial}{\partial x} \left(K \, \frac{\partial T}{\partial x} \right) dx \right] dy +$$

$$- \left[K \, \frac{\partial T}{\partial y} + \frac{\partial}{\partial y} \left(K \, \frac{\partial T}{\partial y} \right) dy \right] dx$$

$$\text{VARIAÇÃO DA ENERGIA INTERNA} = \frac{dE}{dt} = \rho \, c \, dx \, dy \, \frac{\partial T}{\partial t}$$

então

$$\frac{\partial}{\partial x} \left(K \, \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(K \, \frac{\partial T}{\partial y} \right) + q_{\text{GER}} = \rho \, c \, \frac{\partial T}{\partial t}$$

para K constante

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{q_{\text{GER}}}{K} = \frac{1}{\alpha} \, \frac{\partial T}{\partial t}$$

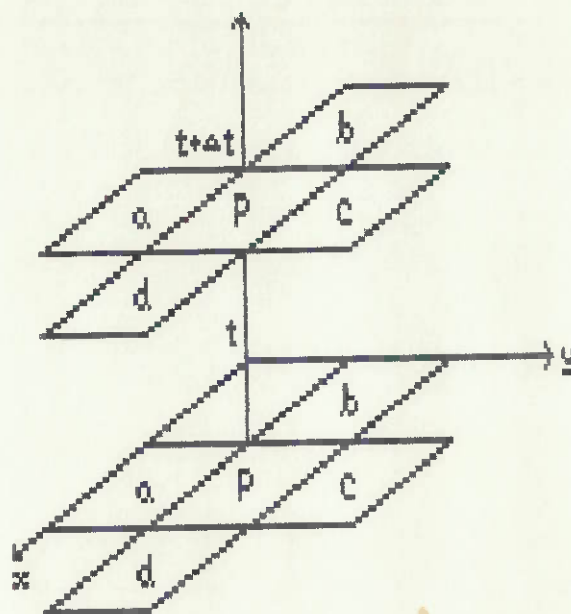
2.3.2. Descrição do Método de Crank-Nicholson

De forma semelhante à equação deduzida acima, podemos definir este método como sendo um balanço de energia na seguinte forma:

VARIAÇÃO DA ENERGIA INTERNA ENTRE OS TEMPOS t E $t+\Delta t$ É IGUAL A MÉDIA ARITMÉTICA DAS SOMATORIAS DOS FLUXOS DE CALOR MAIS OS CALORES GERADOS INTERNAMENTE, NOS TEMPOS t E $t+\Delta t$.

ou seja, para o elemento genérico (FIGURA 2)

FIGURA 2



temos

$$\frac{T_P^{t+\Delta t} - T_P^t}{\alpha \Delta t} = \frac{1}{2} \{ (\sum q_{PI}^{t+\Delta t} + q_{GER}^{t+\Delta t}) + (\sum q_{PI}^t + q_{GER}^t) \}$$

onde

- os termos em **negrito** são conhecidos.

- $\alpha = K / (R_0 \times CE)$

- $\sum q = q_{PA} + q_{PB} + q_{PC} + q_{PD}$

- $q_{PI}^{t+\Delta t}$ é o fluxo de calor entre os elementos P e I no instante $t+\Delta t$. Estes fluxos são formulados conforme as equações (2) e (4) descritas em 2.2.1. e 2.2.2.

Desta forma conseguimos obter, para cada elemento de área de uma figura proposta, uma equação das temperaturas no instante $t+\Delta t$.

Como conhecemos uma temperatura para cada elemento e condições de contorno conhecidas, temos em mãos um conjunto de N equações a N incógnitas, passível de ser resolvido.

3. PROGRAMAÇÃO

O programa foi elaborado em quatro etapas:

TELA1 - faz a tela de apresentação do programa. Define a forma de saída do programa (tela ou tela e impressora).

TELA2 - faz a definição da figura com a qual queremos trabalhar.

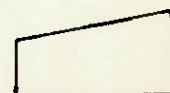
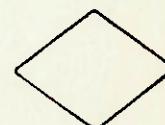
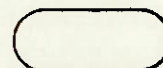
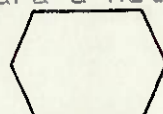
CA1 - desenvolve a malha de pontos sobre a figura proposta.

CA2 - cerne do programa. Realiza todos os cálculos termodinâmicos.

A forma de apresentação destes programas seguirá o seguinte esquema: Comentário, Variáveis, Diagrama de Blocos e Listagem. Os programas CA1 e CA2, por serem extensos, serão divididos em seus componentes, aqui chamados de Módulos.

O Diagrama de Blocos utilizará a notação:

FOR.....
IF.....
INICIO/FIM.....
INPUT.....
INSTRUÇÃO A EXECUTAR.....
LIGAÇÃO (USADA AOS PARES).....
NEXT.....
RETURN.....
SAIDA EM IMPRESSORA.....
SAIDA EM TELA.....



3.1. TELA1

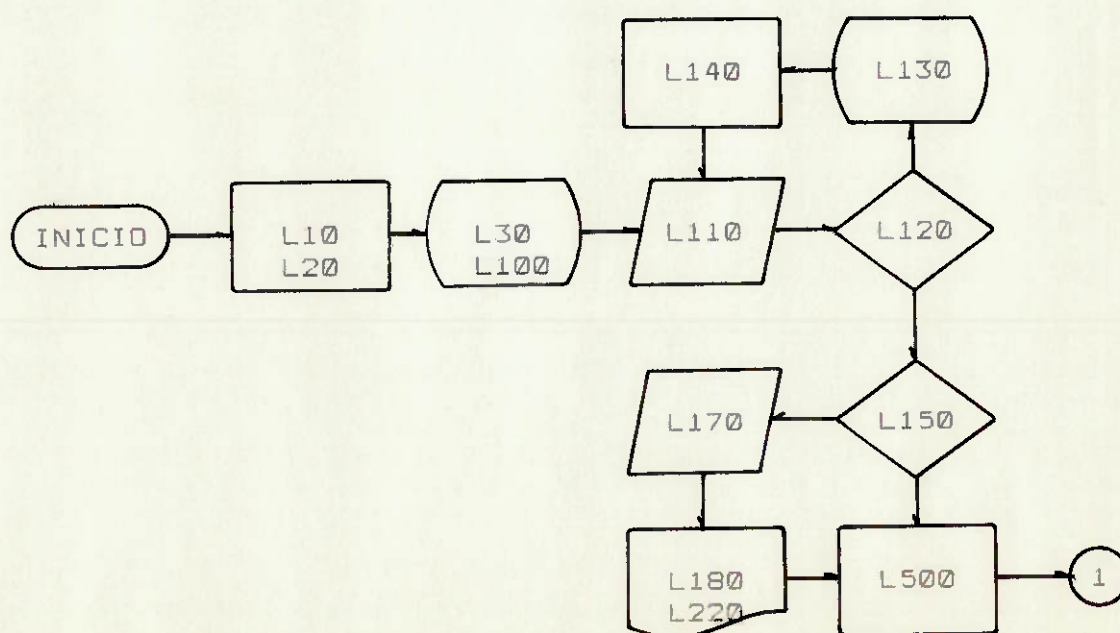
Comentário: este programa tem por finalidade preparar o microcomputador, limpando todas as suas variáveis e a tela com as instruções da linha 10, em seguida montar uma tela de apresentação (linhas 20 a 100), e por fim pedir ao usuário do programa que defina o tipo de saída desejada (somente tela ou tela e impressora), e um nome para o exemplo a ser rodado, no caso de saída com impressora (linhas 110 a 220). Em seguida é carregado o programa TELA2 (linha 500).

Variáveis:

SAIDA\$ - contém o tipo de saída .

NOME\$ - contém o nome do exemplo.

Diagrama:



Listagem:

```
10 REM TELA 1
20 CLEAR : CLS : KEY OFF
30 LOCATE 5,5 : PRINT "Universidade de Sao Paulo"
40 LOCATE 6,5 : PRINT "Escola Politecnica"
50 LOCATE 7,5 : PRINT "Departamento de Engenharia
Mecanica"
60 LOCATE 8,5 : PRINT "Materia : PMC-580"
70 LOCATE 9,5 : PRINT "Professor Orientador :
Marcos de Mattos Pimenta"
80 LOCATE 12,10 : PRINT "PROGRAMACAO DO METODO DE
CRANK-NICHOLSON"
90 LOCATE 14,5 : PRINT "Aluno : Joao Fernandes
Alves Neto"
100 LOCATE 15,5 : PRINT "N";CHR$(167);" U.S.P. :
2233234"
110 LOCATE 21,5 : INPUT "ESCOLHA A SAIDA
(Tela/Impressora) ";SAIDA$
120 IF SAIDA$="T" OR SAIDA$="t" OR SAIDA$="I" OR
SAIDA$="i" THEN 150
130 LOCATE 21,41 : PRINT " "
140 GOTO 110
150 IF SAIDA$="T" OR SAIDA$="t" THEN 500
170 INPUT "Entre o nome do exemplo ";NOME$
180 LPRINT "Programacao do Metodo de Crank-
Nicholson"
190 LPRINT
200 LPRINT NOME$
210 LPRINT
220 LPRINT DATE$;" ";TIME$
500 CHAIN "TELA2",,ALL
```

3.2. TELA2

Comentário: neste programa dimensionamos algumas variáveis que serão usadas em CA1 (linha 30). Lembramos, ao usuário, algumas restrições do programa (linhas 100 a 140). Pedimos ao usuário para definir a figura com a qual iremos trabalhar fornecendo o número de divisões em x e y (linha 150), o número de arestas da figura (linha 160) e as suas coordenadas em metros (linhas 220 a 290). Tendo estes dados, obtemos os passos DX e DY (linhas 340 e 350), e os valores já normalizados para as maiores abscissa e afastamento da figura (C e D nas linhas 360 e 370). Em seguida reapresentamos os dados digitados pelo usuário (linhas 400 a 460), e pedimos confirmação (linha 470 a 490). Na sequência, caso o usuário tenha definido a saída em impressora, são impressos os dados (linhas 500 a 580). Por último passamos para o programa CA1 (linha 1000).

Variáveis:

XD - número de divisões da figura em x.

YD - idem em y.

A - número de arestas da figura.

XL - menor abscissa da figura.

XH - maior abscissa da figura.

YL - menor afastamento da figura.

YH - maior afastamento da figura.

DX - passo na direção x.

DY - passo na direção y.

C - número máximo de nós na direção x.

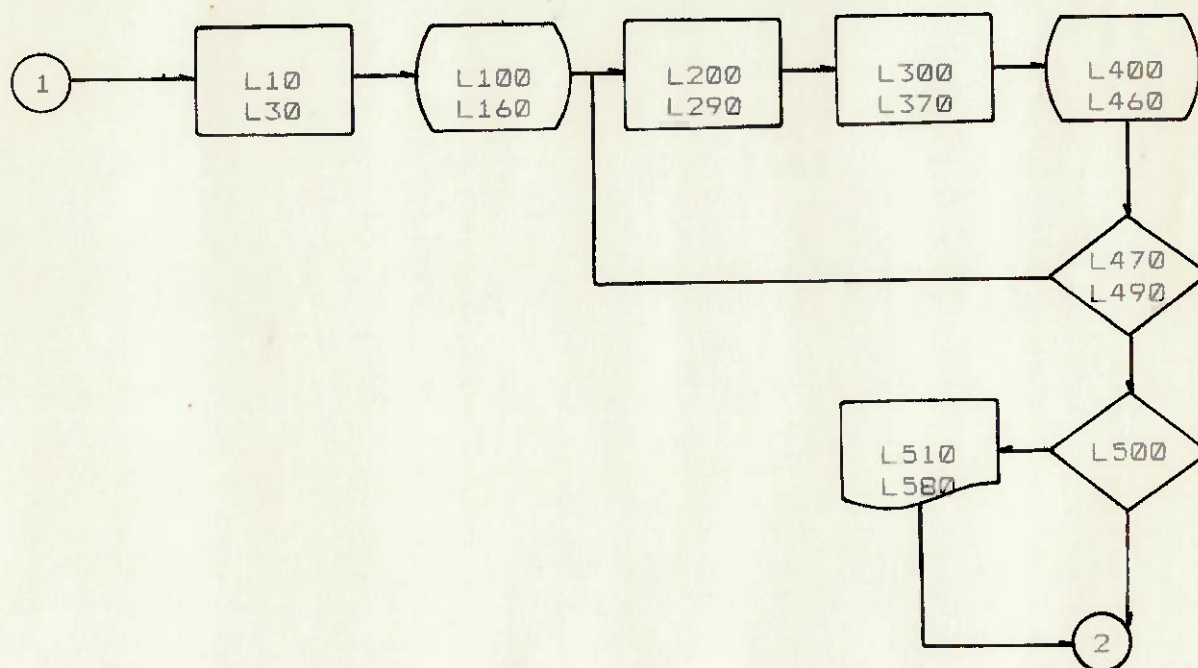
D - número máximo de nós na direção y.

P(I,1) - abcissas das arestas da figura.

P(I,2) - afastamentos das arestas da
figura.

onde $1 < I < A$

Diagrama:



Listagem:

```

10 REM Tela 2
20 CLS
30 DIM P(100,7) , ID(30,30) , LA(50,8) , PI(30,30)
  , NP(100,2)
100 LOCATE 1,5 : PRINT "ENTRE AS COORDENADAS DAS
ARESTAS ."
110 LOCATE 3,5 : PRINT "AS COORDENADAS DEVERAO
COINCIDIR COM A MALHA PROPOSTA "
120 LOCATE 4,5 : PRINT "PELO NUMERO DE DIVISOES
ADIANTE ."
130 LOCATE 6,5 : PRINT "NAO ALINHE 3 OU MAIS
PONTOS ."
140 LOCATE 8,5 : PRINT "A SEQUENCIA DE PONTOS
DEVERA OBEDECER O SENTIDO HORARIO ."
150 LOCATE 10,5 : INPUT "NUMERO DE DIVISOES (X,Y)
";XD,YD
160 LOCATE 12,5: INPUT"NUMERO DE PONTOS ARESTA ";A
200 XH=0 : YH=0
210 XL=1E+20 : YL=1E+20
220 FOR I=1 TO A
230     LOCATE 14,10 : PRINT "PONTO :";I
240         LOCATE 16,10 : INPUT " (X,Y)
";P(I,1),P(I,2)
250     IF XH<P(I,1) THEN XH=P(I,1)
260     IF YH<P(I,2) THEN YH=P(I,2)
270     IF XL>P(I,1) THEN XL=P(I,1)
280     IF YL>P(I,2) THEN YL=P(I,2)
290 NEXT I
300 P(0,1)=P(A,1) : P(0,2)=P(A,2)
310 P(A+1,1)=P(1,1) : P(A+1,2)=P(1,2)
320 P(A+2,1)=P(2,1) : P(A+2,2)=P(2,2)
340 DX=(XH-XL)/XD
350 DY=(YH-YL)/YD
360 C=XD+1
370 D=YD+1
400 LOCATE 18,2 : PRINT "( X )"
410 LOCATE 19,2 : PRINT "( Y )"
420 FOR I=1 TO A
430     LOCATE 17,8+5*I : PRINT I
440     LOCATE 18,8+5*I : PRINT P(I,1)
450     LOCATE 19,8+5*I : PRINT P(I,2)
460 NEXT I
470 INPUT " AS COORDENADAS ESTAO CORRETAS (S/N)
";RESP$
480 IF RESP$<>"S" AND RESP$<>"s" AND RESP$<>"N"
AND RESP$<>"n" THEN 470
490 IF RESP$="N" THEN 200
500 IF SAIDA$="T" OR SAIDA$="t" THEN 1000
510 LPRINT
520 LPRINT "PONTO    X        Y    "
530 FOR I=1 TO A

```

```
540 LPRINT SPC(2);I;  
550 LPRINT SPC(6-(LPOS(0)-3));P(I,1);  
560 LPRINT SPC(6-(LPOS(0)-9));P(I,2)  
570 LPRINT  
580 NEXT I  
1000 CHAIN "CA1",,ALL
```

3.3. CA1

O programa CA1 foi elaborado para atender a seguinte necessidade:

Tendo uma figura e uma malha propostas pelo usuário em TELA2, dentro das características do programa, precisamos identificar cada ponto da intersecção entre a figura e a malha, identificar quais pontos são seus vizinhos, e identificar o tipo do ponto, ou seja, se é interno, periférico, aresta, etc.

O núcleo básico deste programa é o princípio da alternância que será descrito no módulo "Pontos Internos". O programa foi construído de maneira modular, e daqui para frente iremos descrever cada um deles.

3.3.1. Módulo Normalização das Arestas da Figura

Comentário: a necessidade de montar este bloco surgiu devido a uma característica básica da programação, qual seja, todas as variáveis indexadas (matrizes) aceitam como índice exclusivamente valores inteiros. Então, por exemplo, $A(1.9321, 3.1415)$ não é aceito pelo computador ou, quando aceito, tem seus valores truncados ou arredondados para números inteiros, no exemplo, arredondando, $A(2, 3)$. Como o programa exige uma variável $PI(I, J)$ que contenha o "mapa" dos pontos, fomos obrigados a normalizar as coordenadas das arestas, e no decorrer do programa, dos outros pontos da figura. Entende-se por "normalizar" o ato de substituir as coordenadas reais dos pontos, que na maioria dos casos não são números inteiros, por coordenadas inteiras, de forma a não alterar a figura com a qual estamos trabalhando. Isto é possível, desde que guardemos, em alguma variável, a distância real entre os pontos, aqui utilizamos as variáveis DX e DY , e colocando como coordenada dos pontos da figura a sua posição relativa dentro da malha, por exemplo, o sexto ponto, da esquerda para a direita, e quarto, de baixo para cima, de uma figura hipotética, passa a ter as coordenadas seis e quatro.

Variáveis:

$NP(I,1)$ - guarda as coordenadas normalizadas x de todos os pontos da figura.

onde $1 < I < \text{número de arestas} + 2$.

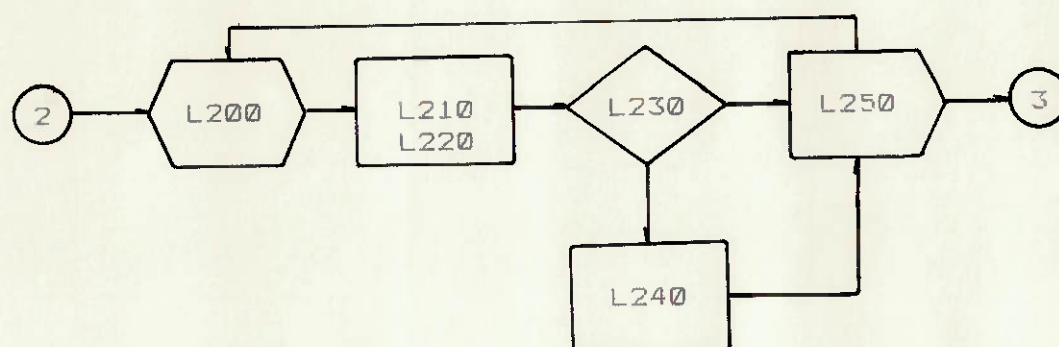
$NP(I,2)$ - idem para as coordenadas y.

onde $1 < I < \text{número de arestas} + 2$.

$PI(I,J)$ - "mapa" normalizado dos pontos da figura.

onde $1 < I < C$ e $1 < J < D$.

Diagrama:



Listagem:

```

200 FOR I=0 TO A+2
210   NP(I,1)=CINT((P(I,1)-XL)/DX)+1
220   NP(I,2)=CINT((P(I,2)-YL)/DY)+1
230   IF I>A THEN 250
240   PI(NP(I,1),NP(I,2))=I
250 NEXT I
  
```

3.3.2. Módulo Retas da Figura

Comentários: utilizando as coordenadas das arestas da figura, definidas pelo usuário em TELA2, vamos obter os coeficientes a , b , c das retas $ax+by+c=0$, que compõe a figura.

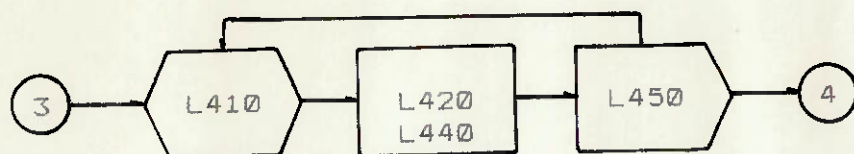
Variáveis: $R(I,1)$ - corresponde a a .

$R(I,2)$ - " b .

$R(I,3)$ - " c .

onde $0 < I < \text{numero de arestas} + 1$

Diagrama :



Listagem:

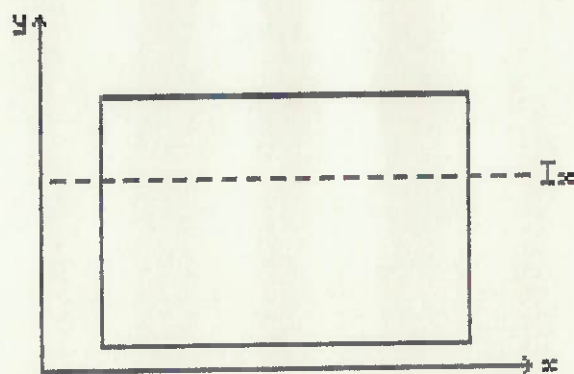
```

400 REM RETAS DA FIGURA
410 FOR I=0 TO A+1
420   R(I,1)=NP(I,2)-NP(I+1,2)
430   R(I,2)=NP(I+1,1)-NP(I,1)
440   R(I,3)=NP(I,1)*NP(I+1,2)-NP(I+1,1)*NP(I,2)
450 NEXT I
  
```

3.3.3. Módulo Pontos de Cruzamento

Comentário: este bloco tem como objetivo calcular as coordenadas x e y dos pontos de cruzamento, ou intersecção (linha 590 e 600), de todas as retas que compõem a figura com cada reta paralela ao eixo x (FIGURA 3) pertencente à malha. Quando ocorrer casos de retas paralelas não superpostas (sem ponto de intersecção), teremos como saída em x e y o valor -1 (linha 550), já retas paralelas superpostas terão como saída as coordenadas da aresta que seja a primeira no sentido horário (linhas 560 e 570).

FIGURA 3



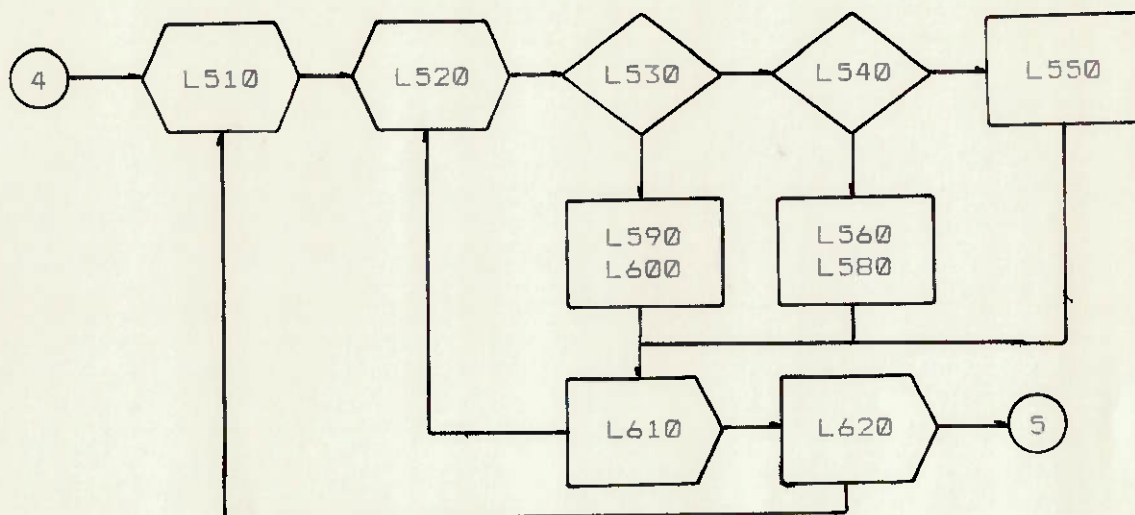
Variáveis: $PC(I,J,1)$ armazena a coordenada x do ponto de cruzamento da reta I da malha com a reta J da figura.

$PC(I,J,2)$ idem para y .

onde $1 < I < D$

$1 < J < \text{numero de arestas (ou retas)} + 1$.

Diagrama :



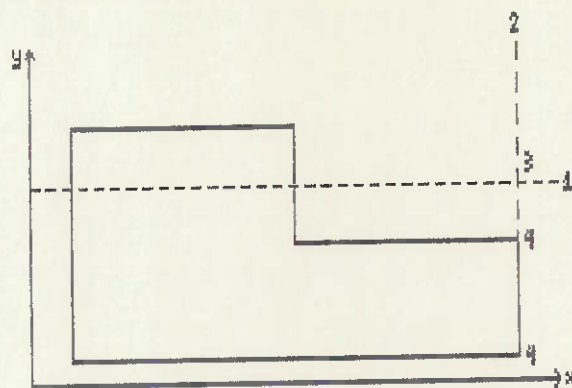
Listagem:

```

500 REM PONTOS DE CRUZAMENTO
510 FOR I=1 TO D
520   FOR J=1 TO A+1
530     IF R(J,1)<>0 THEN 590
540     IF I=NP(J,2) THEN 560
550     PC(I,J,1)=-1 : PC(I,J,2)=-1 : GOTO 610
560     PC(I,J,1)=NP(J,1)
570     PC(I,J,2)=NP(J,2)
580     GOTO 610
590     PC(I,J,1)=-(R(J,2)*I+R(J,3))/R(J,1)
600     PC(I,J,2)=I
610   NEXT J
620 NEXT I
  
```

3.3.4. Módulo Descarte dos pontos de cruzamento falsos.

FIGURA 4



onde

1 - reta paralela ao eixo x ($1x$)
pertencente à malha.

2 - reta da figura.

3 - ponto de cruzamento falso.

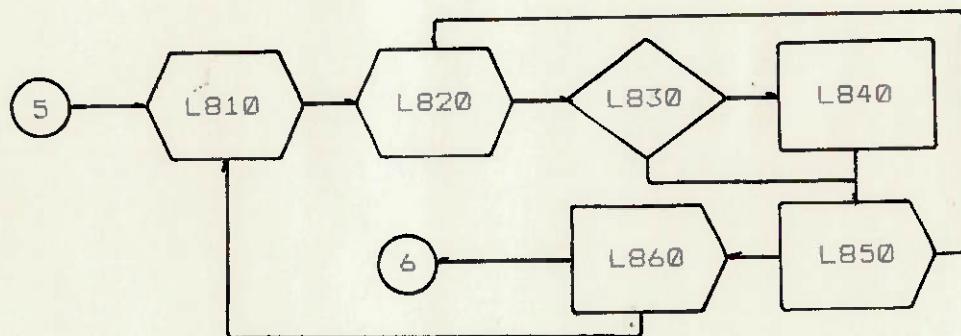
4 - arestas da reta da figura.

Comentário: como podemos observar na FIGURA 4, para a reta da malha analisada, temos um ponto que não corresponde a um cruzamento entre esta reta e a figura proposta. Para identificarmos este tipo de ponto, verificamos as arestas das retas da figura, comparando suas coordenadas y com as coordenadas y dos pontos de cruzamento obtidos (linha 830), caso o afastamento do ponto de cruzamento não seja um valor entre ou igual ao das arestas, ele será um ponto de cruzamento falso.

Para identificá-lo colocaremos nas variáveis $PC(I,J,1)$ e $PC(I,J,2)$ o valor -1 (linha 840).

Variáveis: nenhuma nova variável é implementada neste módulo.

Diagrama:



Listagem:

```

800 REM DESCARTE DE PONTOS DE CRUZAMENTO FALSOS
810 FOR I=1 TO D
820   FOR J=1 TO A
830     IF (PC(I,J,2)<=NP(J+1,2)
AND PC(I,J,2)>=NP(J,2)) OR (PC(I,J,2)>=NP(J+1
,2) AND PC(I,J,2)<=NP(J,2)) THEN 850
840     PC (I,J,1)=-1 : PC(I,J,2)=-1
850   NEXT J
860 NEXT I
  
```

3.3.5. Módulo Ordenamento dos Pontos de Cruzamento & Módulo Descarte dos Pontos de cruzamento Repetidos

Comentário: para cada reta horizontal da malha obtivemos um certo número de pontos de cruzamento, com as retas que formam a figura. Por exemplo, caso nossa figura seja um retângulo as retas horizontais da malha irão cruzar a figura em dois pontos. Estes pontos não estão necessariamente dispostos em ordem crescente na variável $PC(I,J,1)$. O Módulo Ordenamento dos Pontos de Cruzamento faz o reordenamento desta variável (linhas 910 a 1000).

As vezes o mesmo ponto de cruzamento aparece duas ou mais vezes nas variáveis $PC(I,J,1)$ e $PC(I,J,2)$, é o caso de Arestas. Através do Módulo Descarte dos Pontos de Cruzamento repetidos colocamos as coordenadas $(-1,-1)$ em uma das repetições (linhas 1070 e 1080), e sinalizamos, com a variável FLAG (=1) (linha 1090), que um novo ordenamento é necessário (linha 1120).

Variáveis:

K - variável índice auxiliar.

AX - armazena o valor de $PC(I,J,1)$.

AY - armazena o valor de $PC(I,J,2)$.

FLAG - variável de sinalização.

Diagrama 1 :

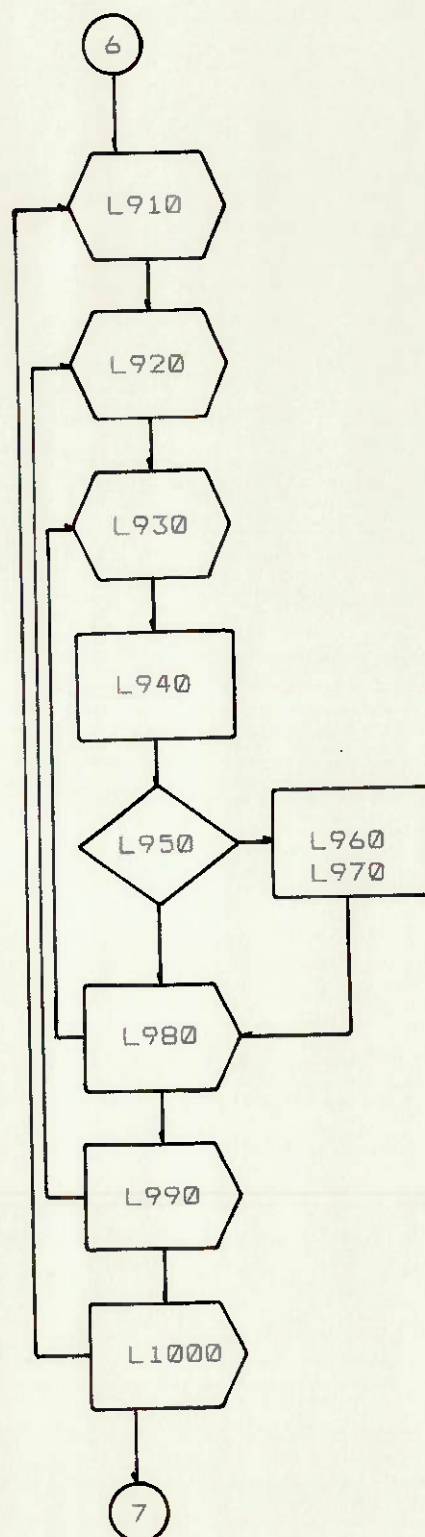
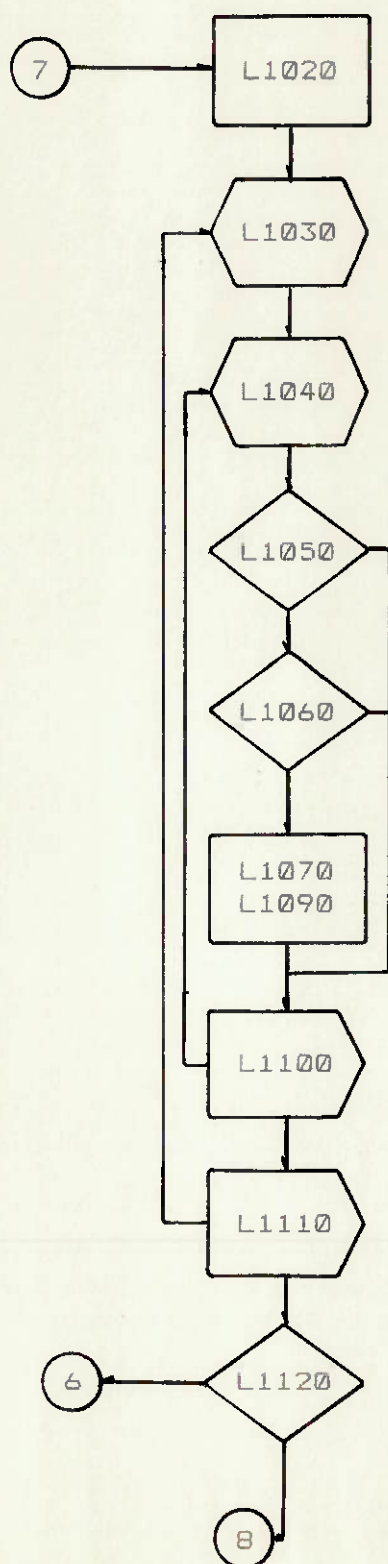


Diagrama 2 :



Listagem:

```
900 REM ORDENAMENTO DOS PONTOS DE CRUZAMENTO
910 FOR I=1 TO D
920   FOR J=1 TO A-1
930     FOR K=J+1 TO A
940       AX=PC(I,J,1) : AY=PC(I,J,2)
950       IF PC(I,J,1)<PC(I,K,1) THEN 980
960       PC(I,J,1)=PC(I,K,1) :
PC(I,J,2)=PC(I,K,2)
970       PC(I,K,1)=AX : PC(I,K,2)=AY
980     NEXT K
990   NEXT J
1000 NEXT I
1010 REM DESCARTE DE PC REPETIDO
1020 FLAG=0
1030 FOR I=1 TO D
1040   FOR J=1 TO A-1
1050     IF PC(I,J,2)=-1 THEN 1100
1060     IF PC(I,J,1)< $\Delta$ PC(I,J+1,1) OR
PC(I,J,2)< $\Delta$ PC(I,J+1,2) THEN 1100
1070     PC(I,J,1)=-1
1080     PC(I,J,2)=-1
1090     FLAG=1
1100   NEXT J
1110 NEXT I
1120 IF FLAG=1 THEN 900
```

3.3.6. Módulo Pontos Internos

Comentário: este módulo foi desenvolvido juntamente com as subrotinas "Coloca Pontos" e "Intervalo coincide com alguma reta da figura", para nos fornecer como saída a variável $PI(I,J)$, que guarda as coordenadas normalizadas de todos os pontos da malha que pertencem à figura. A seguir apresentamos a estrutura deste módulo.

A variável $SOMA$ será usada para efetuarmos a numeração dos pontos internos. Inicialmente terá o valor do número de arestas mais um (linha 1210). Iniciamos uma instrução FOR-NEXT de forma a podermos estudar todas as retas horizontais pertencentes à malha (linha 1220).

Verificamos quantos pontos de cruzamento existem para cada uma destas retas (linhas 1250 a 1300).

Colocamos na variável J o valor aonde começam os pontos de cruzamento das variáveis $PC(I,J,1)$ e $PC(I,J,2)$ (linhas 1310 a 1340).

Observação: estas variáveis foram ordenadas no módulo anterior, e, possivelmente, nos primeiros valores de J ($J=1,2,3,\dots$), está alocado o valor -1 (ver seção 3.3.5. e 3.3.2.).

Condiçionamos a continuidade da programação ao número de pontos de cruzamento (linha 1360).

Se tivermos apenas dois pontos de cruzamento, podemos afirmar que temos um intervalo Interno, e seguir diretamente para a subrotina "Coloca Pontos" (linha 1370), e passarmos para a próxima reta horizontal da malha (linha 1380).

Se tivermos mais do que dois pontos, o primeiro intervalo sempre terá pontos internos (linha 1400), e poderá ser Interno ou Intervalo coincidente, ou seja, os dois primeiros pontos de cruzamento coincidem com duas arestas da figura. Esta condição é verificada (linhas 1410 e 1422).

Se for Interno, então passamos para o próximo intervalo (linha 1500).

Se for Intervalo coincidente, verificamos as arestas da figura imediatamente anterior e posterior à Linha de Contorno. Se as duas tiverem as suas variáveis $PC(I,J,2)$ acima e abaixo das variáveis $PC(I,J,2)$ da Intervalo coincidente, então o próximo intervalo será Interno ($TR0=0$ na linha 1450), caso contrário será Externo ($TR0=1$ na linha 1430).

Passamos ao próximo intervalo (linha 1460). Se o intervalo é Interno, colocamos os seus pontos, e passamos para o próximo intervalo (linhas 1630 e 1640).

Se o intervalo é Externo, passamos para o próximo intervalo (linha 1610).

Atualizamos a variável J (linha 1500), e verificamos se chegamos ao último ponto de cruzamento (linha 1510). das variáveis PC(I,J,2) da intervalo coincidente, então o próximo intervalo será Interno (TRD=0 na linha 1450), caso contrário será Externo (TRD=1 na linha 1430).

Passamos ao próximo intervalo (linha 1460).

Se o intervalo é Interno, colocamos os seus pontos, e passamos para o próximo intervalo (linhas 1630 e 1640).

Se o intervalo é Externo, passamos para o próximo intervalo (linha 1610).

Atualizamos a variável J (linha 1500), e verificamos se chegamos ao último ponto de cruzamento (linha 1510).

Em caso afirmativo, passamos para a próxima reta horizontal da malha (próximo I).

Em caso negativo, verificamos se o intervalo é intervalo coincidente.

Em caso afirmativo, colocamos os pontos internos (linha 1670), e verificamos as arestas imediatamente anterior e posterior à intervalo coincidente (linha 1680).

Se as duas tiverem as suas variáveis PC(I,J,2) acima ou abaixo das variáveis PC(I,J,2)

da Intervalo coincidente, então o próximo Intervalo é idêntico ao anterior à Intervalo coincidente, e passamos diretamente para ele (linha 1730).

Se as duas tiverem as suas variáveis $PC(1,j,2)$ acima e abaixo da Intervalo coincidente, então o próximo Intervalo é o contrário ao anterior à Intervalo coincidente.

Se o Intervalo anterior à Intervalo coincidente era Interno, o Intervalo seguinte à Intervalo coincidente será Externo, e vice-versa (linhas 1690 a 1720).

Passamos para o próximo Intervalo (linha 1500).

Esta sequência irá se repetir até que cheguemos ao último ponto de cruzamento da última reta horizontal da malha.

Variáveis:

SX(I) - esta variável guarda o número de pontos de cruzamento para cada reta horizontal da malha.

I - indica a reta horizontal da malha,

com a qual estamos trabalhando no momento.

J - indica o ponto de cruzamento, com o qual estamos trabalhando no momento.

qual estamos trabalhando no momento. O intervalo, então, será o indicado por J e J+1.

TRD - indicador de estado. Igual a um,

se as arestas imediatamente anterior e posterior a

uma intervalo coincidente tiverem as suas

variáveis $PC(I,J,2)$ acima e abaixo das variáveis

$PC(I,J,2)$ da intervalo coincidente.

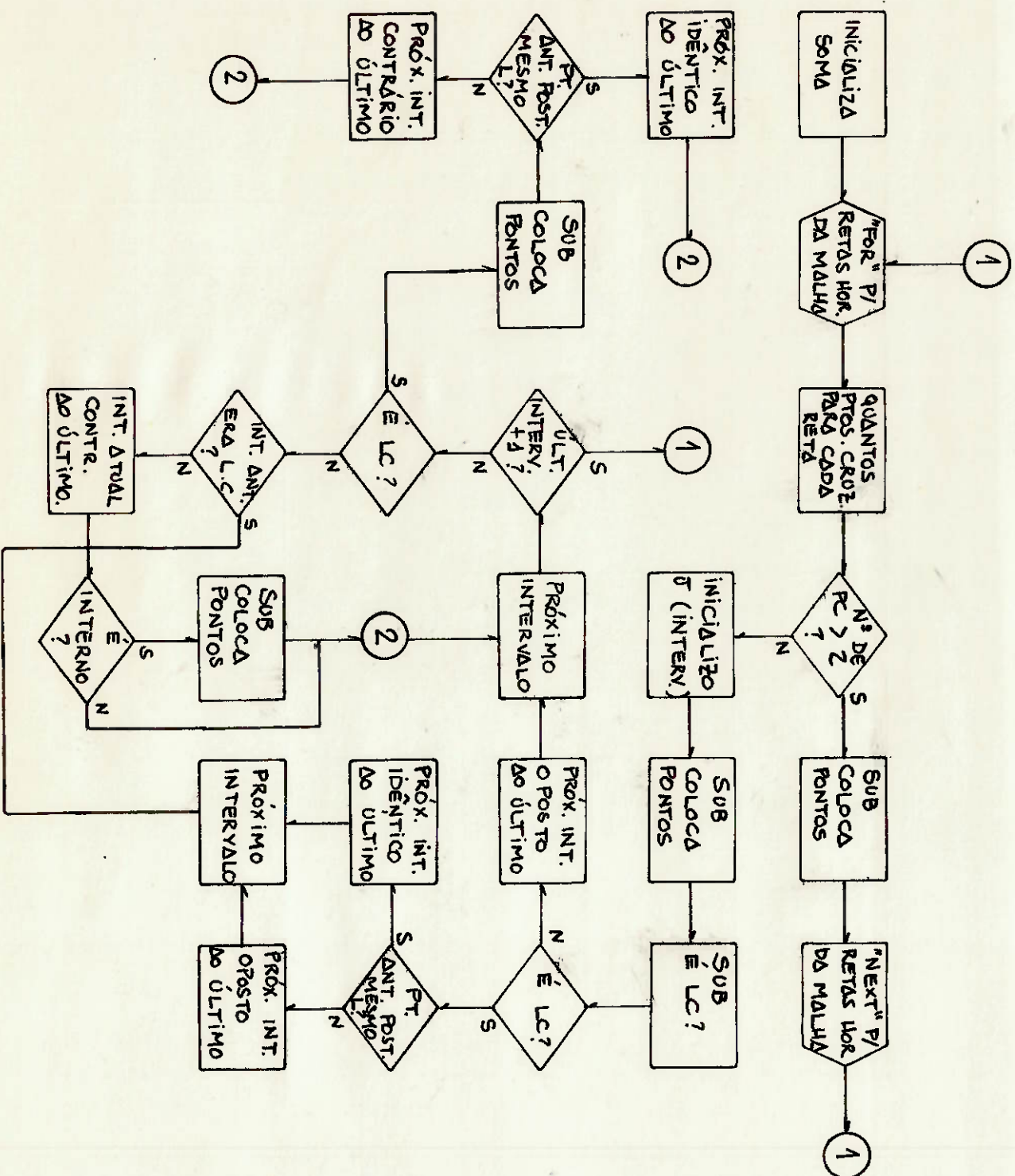
FLAG - indicador de estado. Igual a um

se o intervalo analisado for intervalo

coincidente.

GALF - indicador de estado. Igual a um

se o intervalo anterior é intervalo coincidente.



```

1200 REM OBTENCAO DOS PONTOS INTERNOS
1210 SOMA=A+1
1220 FOR I=1 TO D
1230 GALT=0
1240 SX(I)=0
1250 FOR J=1 TO A
1260 IF PC(I,J,2)=-1 THEN 1300
1270 IF J=A THEN 1290
1280 IF PC(I,J,1)=PC(I,J+1,1) AND
PC(I,J,2)=PC(I,J+1,2) THEN 1300
1290 SX(I)=SX(I)+1
1300 NEXT J
1310 J=1 : REM J DEFINI PC(I,J,1 OU 2)
1320 IF PC(I,J,2)<-1 THEN 1350
1325 IF J=A-1 THEN 1350 : REM PENULTIMO J
1330 J=J+1
1340 GOTO 1320
1350 IF SX(I)=1 THEN PRINT"LINHA 1310"
1360 IF SX(I)>2 THEN 1400
1370 GOSUB 5000 : REM SUB COLOCA PONTOS
1380 GOTO 1740
1400 GOSUB 5000 : REM SUB COLOCA PONTOS
1410 GOSUB 6000 : REM SUB E LC 3 X
1422 IF FLAG=0 THEN 1480
1425 IF SGN(NP(PA,2)-1)<ASGN(NP(PB,2)-1) THEN
1450
1450 TR0=1
1440 GOTO 1460
1450 TR0=0
1460 J=J+1
1470 GOTO 1590
1480 TR0=0
1490 REM VERIFICO PROXIMO INTERVALO
1500 J=J+1
1510 IF J=A GOTO 1740
1520 GOSUB 6000 : REM SUB E LC 3 X
1530 IF FLAG=1 THEN 1650
1540 IF GALT=1 THEN 1590
1550 IF TR0=1 THEN 1580
1560 TR0=1
1570 GOTO 1590
1580 TR0=0
1590 GALT=0
1610 IF TR0=1 THEN 1500
1630 GOSUB 5000 : REM SUB COLOCA PONTOS
1640 GOTO 1500
1650 GALT=FLAG
1670 GOSUB 5000 : REM SUB COLOCA PONTOS
1680 IF SGN(NP(PA,2)-1)=SGN(NP(PB,2)-1) THEN
1730
1690 IF TR0=0 THEN 1720

```

Listaagem:

1700 TR0=0
1710 GOTO 1730
1720 TR0=1
1730 GOTO 1500
1740 NEXT I

3.3.7. Módulo Internos

Comentário: a ideia utilizada é a de analisar os quatro pontos ao redor do ponto analisado. Se para nenhum destes pontos, $PI(I \pm 1, J \pm 1)$ é igual a zero (linha 3130), então o ponto é interno (linha 3140). Para não sobrepormos nenhuma aresta da figura ($1 \leq PI(I, J) \leq A$) fazemos a verificação da linha 3135.

Variáveis:

TIPO\$(I, J) - esta variável irá guardar a classificação de todos os pontos da figura com a qual estamos trabalhando, seguindo esta codificação:

IN - ponto interno.

F2 e F3 - arestas.

F1 - pontos periféricos excluindo as

arestas.

F4 - pontos externos em contato com

algum ponto da malha.

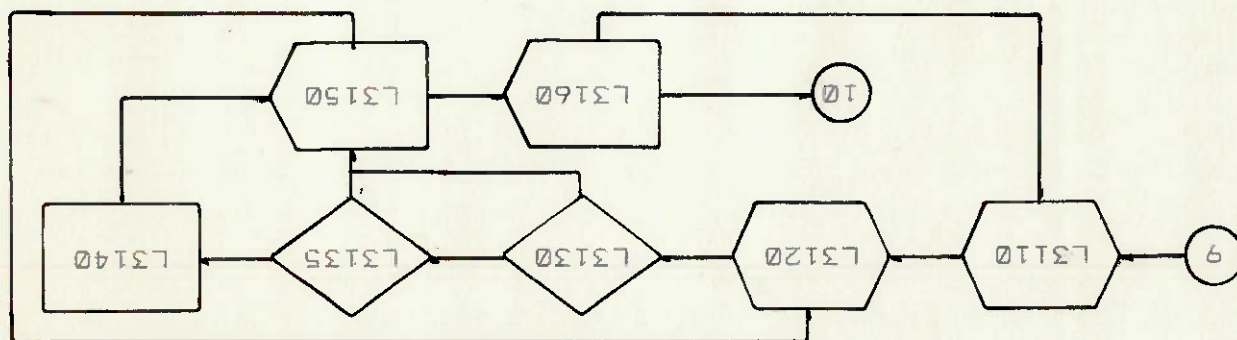
" " - pontos externos sem contato com a

figura.

onde $1 < I < C$

$1 < J < D$

Diagrama:



Listagem:

```

3100 REM IN
3110 FOR I=1 TO C
3120   FOR J=1 TO D
3130     IF P1(I-1,J)=0 OR P1(I+1,J)=0 OR
3135     P1(I,J-1)=0 OR P1(I,J+1)=0 THEN 3150
3140     TIPO$(I,J)="IN"
3150   NEXT J
3160 NEXT I

```

3.3.8. Módulo Arestas

Comentário: as arestas da figura são definidas em TELAZ pelo usuário, portanto, para identificar estes pontos na variável TIPO\$(I,J) como "F2", basta tomarmos as coordenadas das variáveis NP(I,1) e NP(I,2), para os "A" primeiros pontos, que são as próprias arestas.

Variáveis: nenhuma nova variável é implementada neste módulo.

Diagrama:



Listagem:

```

3200 REM ARESTAS
3210 FOR I=1 TO A
3220   TIPO$(NP(I,1)*NP(I,2))="F2"
3230 NEXT I
  
```

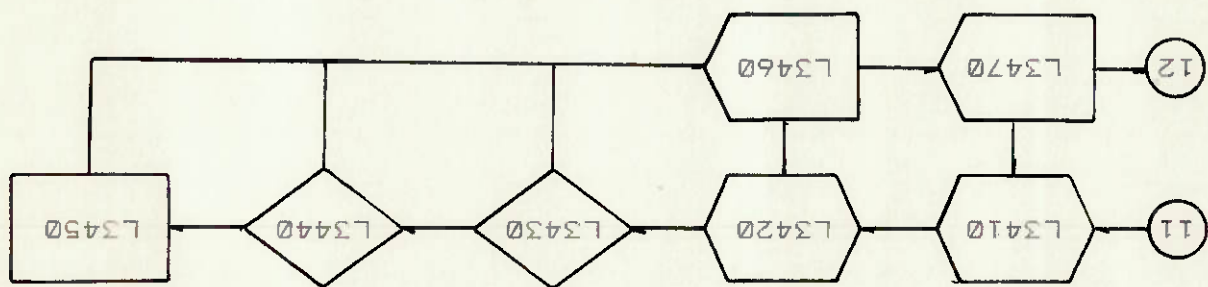
3.3.9. Módulo Periféricos

Comentário: chamamos de pontos periféricos os pontos que estão em contato com algum ponto externo à figura, mas que pertencem à mesma, e não são arestas. Neste módulo analisamos todos os pontos da malha com a qual estamos trabalhando através das duas instruções FOR NEXT concatenadas).

Os pontos periféricos, que agora queremos identificar, pertencem à figura então, por exclusão, os pontos onde temos a condição $PI(I,J)=0$ podem ser excluídos por serem externos à figura (linha 3430). Por outro lado os únicos pontos ainda não identificados da figura ($TIPO$(I,J)=""$) são os periféricos, então podemos eliminar aqueles para os quais $(TIPO$(I,J) < \Delta "")$ (linha 3440). Os pontos remanescentes serão periféricos (linha 3450).

Variáveis: nenhuma nova variável é implementada neste módulo.

Diagrama:



Listagem:

```

3400 REM PERIFERICOS
3410 FOR I=1 TO C
3420   FOR J=1 TO D
3430     IF PI(I,J)=0 THEN 3460
3440     IF TIPO$(I,J)<>" " THEN 3460
3450     TIPO$(I,J)="F1"
3460   NEXT J
3470 NEXT I

```

3.3.10. Módulo Externos

Comentário: neste módulo, analisamos todos os pontos da malha com a qual estamos trabalhando (através das duas instruções FOR NEXT concatenadas). Para cada ponto que está em contato com algum ponto externo (condição: $TIP0(I,J) = "F2"$ ou $TIP0(I,J) = "F3"$, linha 3530), verificamos, nos quatro pontos que estão à sua volta, a variável $PI(I\pm 1, J\pm 1)$ (linhas 3540, 3560, 3580 e 3600). Caso esta variável seja igual a zero, localizamos um ponto externo na posição $(I\pm 1, J\pm 1)$ (linhas 3550, 3570, 3590 e 3610).

Variáveis:

ADICAO - utilizada para numerarmos os pontos externos em contato com a figura.
Incrementada de um, sempre que numeramos um ponto.

```

3490 ADICAO=SOMA
3500 REM Externos
3510 FOR I=1 TO C
3520 FOR J=1 TO D
3530 IF TIPO$(I,J)<>"F2" AND TIPO$(I,J)<>"F3"
THEN 3620
3540 IF PI(I+1,J)<>0 THEN 3560
TIPO$(I+1,J)="F4" : PI(I+1,J)=ADICAO :
ADICAO=ADICAO+1
3560 IF PI(I-1,J)<>0 THEN 3580
TIPO$(I-1,J)="F4" : PI(I-1,J)=ADICAO :
ADICAO=ADICAO+1
3580 IF PI(I,J+1)<>0 THEN 3600
TIPO$(I,J+1)="F4" : PI(I,J+1)=ADICAO :
ADICAO=ADICAO+1
3600 IF PI(I,J-1)<>0 THEN 3620
TIPO$(I,J-1)="F4" : PI(I,J-1)=ADICAO :
ADICAO=ADICAO+1
3620 NEXT J
3630 NEXT I

```

Listagem:

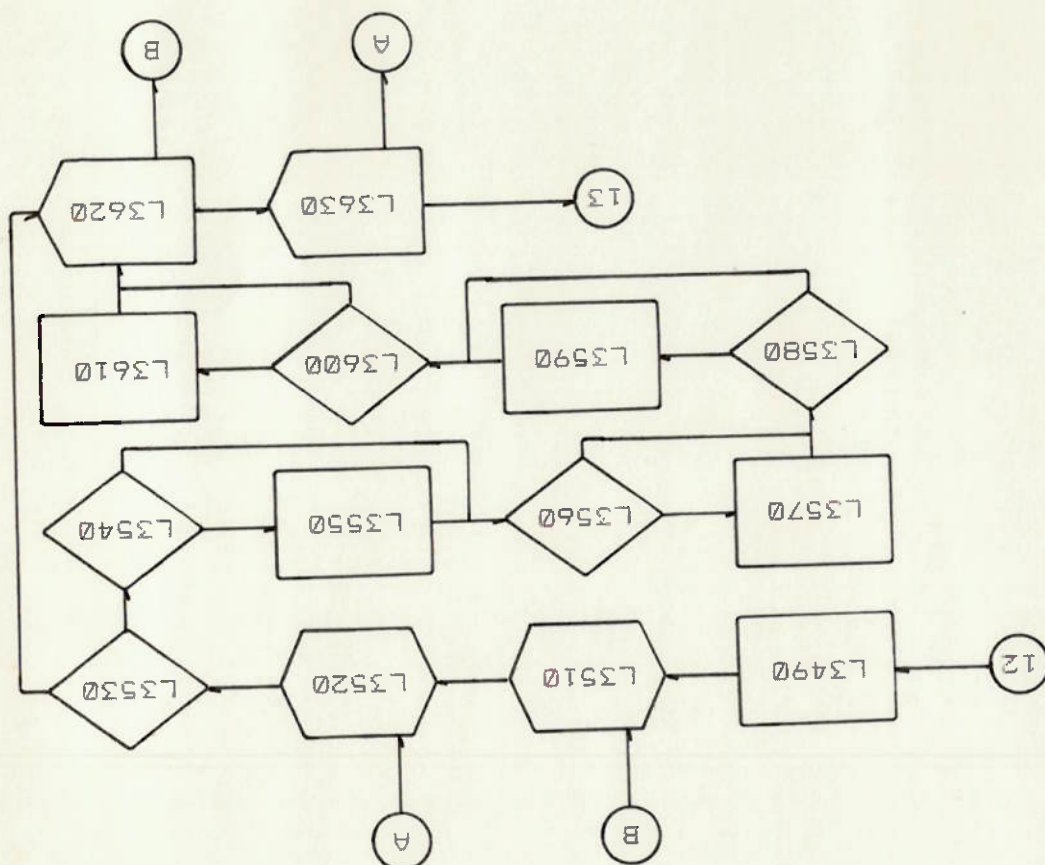


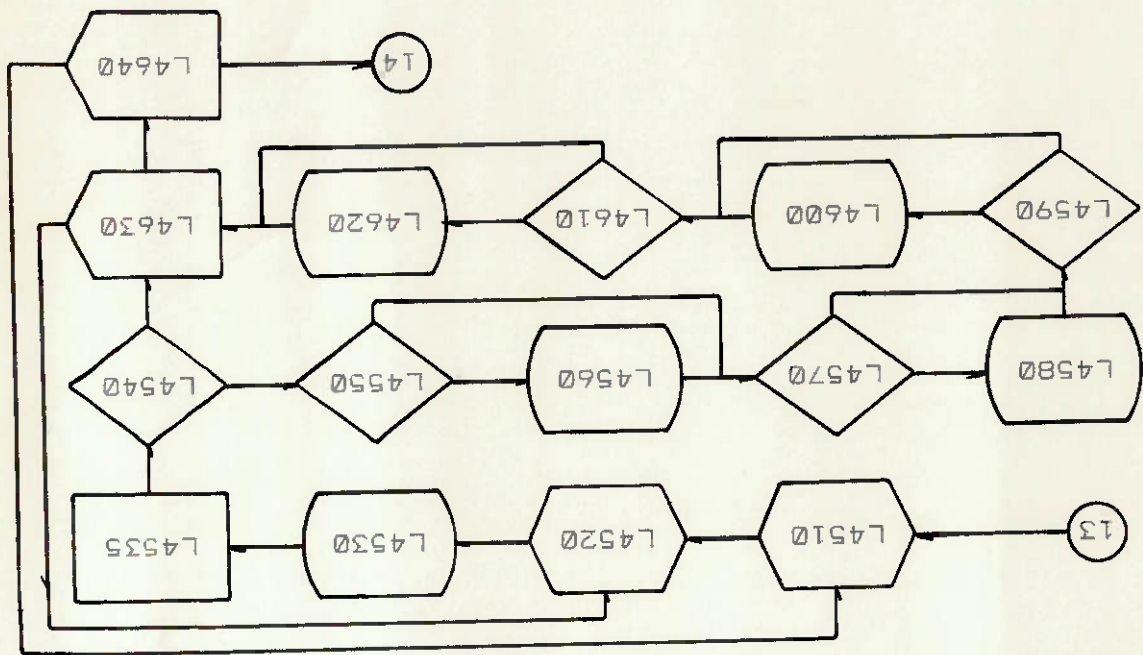
Diagrama:

3.3.11. Módulo Estrutura na tela

Comentário: monta na tela do micro a estrutura com a qual estamos trabalhando, esta eixo x direcionado de cima para baixo, e o eixo y direcionado para a direita. Ainda, este módulo aproveitado para completarmos as variáveis NP e NP(I,2) (linha 4535).

Variáveis: nenhuma nova variável é implementada neste módulo.

Diagrama:



Listagem:

```

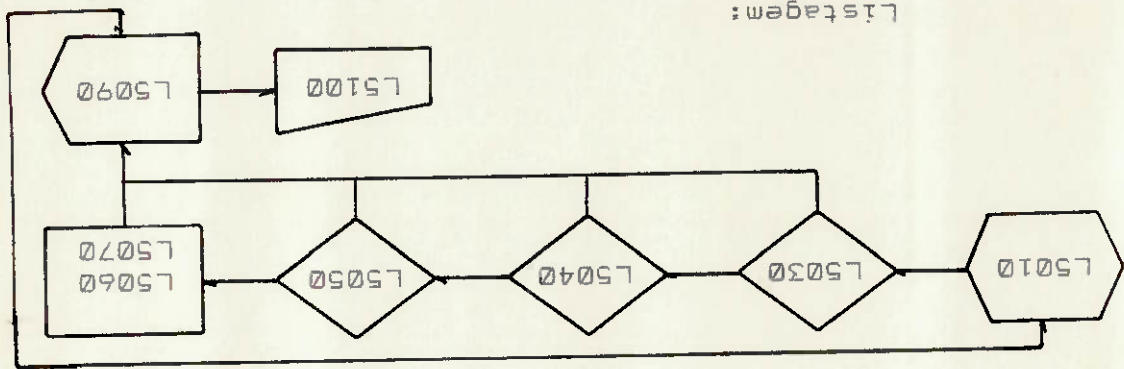
4500 REM ESTRUTURA E NUMERACAO DOS PONTOS
4510 FOR I=0 TO C+1
4520   FOR J=0 TO D+1
4530     LOCATE I*2+1,J*5+1 : PRINT PI(I,J)
4535     NP(PI(I,J),1)=I : NP(PI(I,J),2)=J
4540     IF TIPO$(I,J)="" OR TIPO$(I,J)="F4" THEN
4550       IF TIPO$(I-1,J)="F4" THEN 4570
4560       LOCATE I*2,J*5+2 : PRINT CHR$(179)
4570       IF TIPO$(I,J+1)="F4" THEN 4590
4580       LOCATE I*2+1,J*5+5 : PRINT "-"
4590       IF TIPO$(I+1,J)="F4" THEN 4610
4600       LOCATE I*2+2,J*5+2 : PRINT CHR$(179)
4610       IF TIPO$(I,J-1)="F4" THEN 4630
4620       LOCATE I*2+1,J*5 : PRINT "-"
4630     NEXT J
4640   PRINT
4650 NEXT I

```

3.3.12. Módulo Subrotina Coloca Pontos

Comentário: dado um intervalo compreendido entre PC(I,J,1) e PC(I,J+1,1) inclusive (linhas 5040 e 5050), e após verificar que PI(K,I) não é um ponto conhecido (PI(I,J)=0) (linha 5030), colocamos em PI(I,J) o valor de soma que é "o número de pontos já identificados mais um", e em seguida incrementamos soma de um (linhas 5060 e 5070).
 Variáveis: nenhuma nova variável é implementada neste módulo.

Diagrama:



Listagem:

```

5000 REM SUBROTINA COLOCA PONTOS
5010 FOR K=1 TO C
5030 IF PI(K,I)<>0 THEN 5090
5040 IF K<PC(I,J,1) THEN 5090
5050 IF K>PC(I,J+1,1) THEN 5090
5060 PI(K,I)=SOMA
5070 SOMA=SOMA+1
5090 NEXT K
5100 RETURN
    
```

3.3.13. Módulo Subrotina "Intervalo coincide com alguma reta da figura?"

Comentário: como mostramos no Módulo Pontos

Internos, é fundamental sabermos se o intervalo (

$PC(I,J,1)$, $PC(I,J+1,1)$), que estamos analisando,

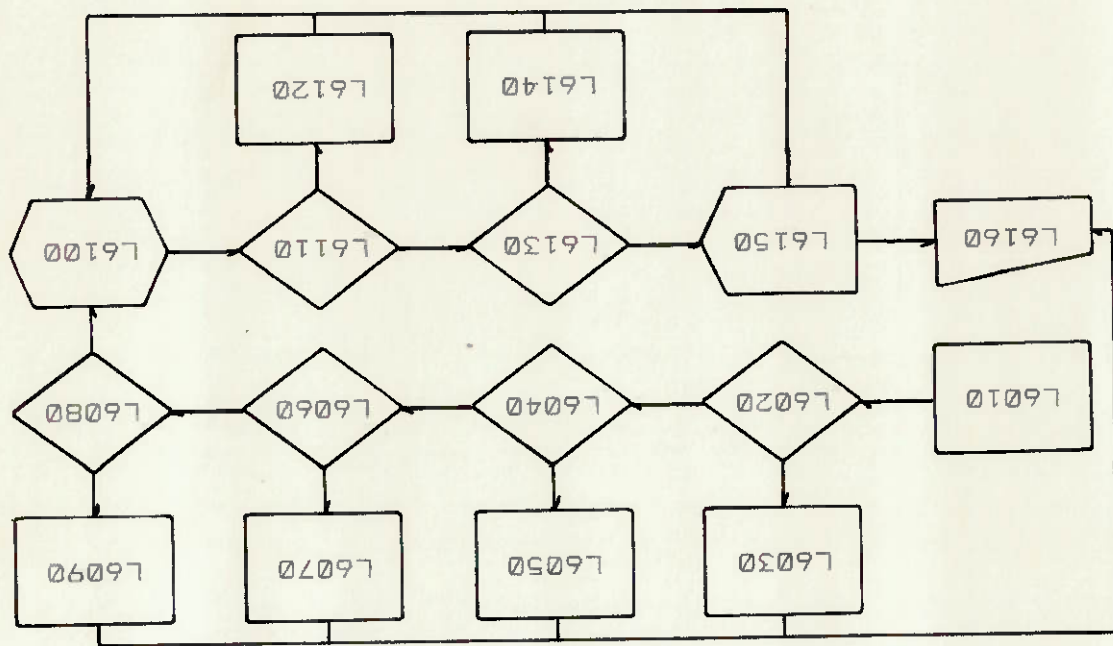
é ou não coincidente com alguma das retas da

figura . Este módulo faz esta verificação.

Variáveis: nenhuma nova variável é implementada

neste módulo.

Diagrama:



```

6000 REM INTERVALO COINCIDENTE COM ALGUMA RETA DA
FIGURAZ
6010 FLAG=0
6020 IF PC(I,J,1)<>ΔNP(1,1) OR PC(I,J,2)<>ΔNP(1,2)
OR PC(I,J+1,1)<>ΔNP(2,1) OR
PC(I,J+1,2)<>ΔNP(2,2) THEN 6040
6030 PA=1 : PB=2 : FLAG=1 : GOTO 6160
6040 IF PC(I,J,1)<>ΔNP(1,1) OR PC(I,J,2)<>ΔNP(1,2)
OR PC(I,J+1,1)<>ΔNP(A,1) OR
PC(I,J+1,2)<>ΔNP(A,2) THEN 6060
6050 PA=1 : PB=A : FLAG=1 : GOTO 6160
6060 IF PC(I,J,1)<>ΔNP(A,1) OR PC(I,J,2)<>ΔNP(A,2)
OR PC(I,J+1,1)<>ΔNP(A-1,1) OR
PC(I,J+1,2)<>ΔNP(A-1,2) THEN 6080
6070 PA=A : PB=A-1 : FLAG=1 : GOTO 6160
6080 IF PC(I,J,1)<>ΔNP(A,1) OR PC(I,J,2)<>ΔNP(A,2)
OR PC(I,J+1,1)<>ΔNP(1,1) OR
PC(I,J+1,2)<>ΔNP(1,2) THEN 6100
6090 PA=A : PB=1 : FLAG=1 : GOTO 6160
6100 FOR K=2 TO A-1
6110 IF PC(I,J,1)<>ΔNP(K,1) OR PC(I,J,2)<>ΔNP(K,2)
OR PC(I,J+1,1)<>ΔNP(K-1,1) OR
PC(I,J+1,2)<>ΔNP(K-1,2) THEN 6130
6120 PA=K : PB=K-1 : FLAG=1 : GOTO 6160
6130 IF PC(I,J,1)<>ΔNP(K,1) OR PC(I,J,2)<>ΔNP(K,2)
OR PC(I,J+1,1)<>ΔNP(K+1,1) OR
PC(I,J+1,2)<>ΔNP(K+1,2) THEN 6150
6140 PA=K : PB=K+1 : FLAG=1
6150 NEXT K
6160 RETURN

```

Listagem:

Segundo o que foi escrito na proposta deste trabalho, e utilizando algumas variáveis obtidas em CA1, vamos agora elaborar um programa capaz de fazer todos os cálculos termodinâmicos necessários, ou seja, o cálculo das temperaturas de todos os pontos da intersecção malha-figura, para cada iteração no tempo, e o cálculo dos fluxos de calor instantâneos, também para cada iteração no tempo.

Tanto quanto possível, este programa será elaborado de maneira modular como CA1.

3.4.1.1. Módulo Definições Iniciais.

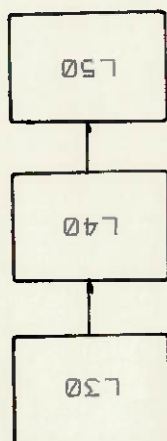
Comentário: estas três linhas iniciais tem como objetivo dimensionar e definir algumas variáveis (linhas 30 e 50), além de permitir o uso do zero (0) como índice, em variáveis matriciais (linha 40).

Variáveis:

N - redefine o número de pontos da malha.

DD - assume o lugar da variável D, para que esta possa ser utilizada de outra forma.

Diagrama:



Listagem:

```

30 N=SOMA-1 : DD=D
40 OPTION BASE 0
50 DIM CAMIN$(ADICAO,4), CTE(ADICAO,ADICAO),
M(ADICAO+1,ADICAO), L(ADICAO,4), T(A
DICAO),NT(ADICAO),AUX(ADICAO)
  
```

3.4.2. Módulo Entrada de dados.

Comentário: todos os dados, necessários aos cálculos termodinâmicos, que serão efetuados neste programa, são pedidos neste módulo (11 linhas a 140 a 600). São reapresentadas três variáveis (11 linhas a 130). Lembramos que o programa exige que todas as propriedades termodinâmicas tenham suas unidades expressas no Sistema Internacional. É pedido confirmação dos valores digitados (11 linhas a 870).

Variáveis:

K - condutividade térmica (W/mK).

CE - calor específico kJ/kgK).

RO - densidade (kg/m³).

TAMB - nas faces aonde ocorre convecção,

é a temperatura do fluido (K).

T - tempo total durante a qual serão

calculadas as iterações. Normalmente entraremos um

valor muito alto, para que haja tempo dos cálculos

realizados convergirem (s).

DT - é o intervalo de tempo entre as

iteraões (s).

QSL - calor gerado internamente (W/m²).

W - temperatura inicial de todos os

pontos da figura (K).

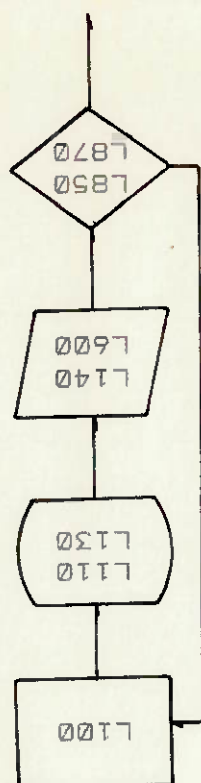
T(I) - este vetor irá guardar a temperatura de cada ponto da figura durante os cálculos (K).

CC\$(I) - para cada face da figura guardamos a condição de contorno. "CO" representa a face submetida a Convecção, e "TC" representa a Temperatura Constante.

TC(I) - para as faces cuja condição de contorno é Temperatura Constante, guardamos a temperatura em que está cada face (K).

AGA(I) - para as faces cuja condição de contorno é Convecção, guardamos o valor do Coeficiente de Convecção (W/m^2K).

Diagrama:



```

100 CLS : REM ENTRADA DE DADOS
110 PRINT "NUMERO DE PONTOS DA MALHA " ; N
120 PRINT "PASSO EM X [DX - M] " ; DX
130 PRINT "PASSO EM Y [DY - M] " ; DY
140 INPUT "CONDUTIVIDADE TERMICA [K - W/MC] " ; K
150 INPUT "CALOR ESPECIFICO [CE - J/KGC] " ; CE
160 INPUT "DENSIDADE [RO - KG/M3] " ; RO
170 INPUT "TEMPERATURA AMBIENTE [C] " ; TAMB
180 INPUT "INTERVALO TOTAL DE TEMPO [T - S] " ; T
190 INPUT "INTERVALO PARCIAL DE TEMPO [DT - S] " ; DT
200 INPUT "CALOR GERADO [W] " ; QZL
210 INPUT "CALCULAR FLUXOS DE CALOR PARA CADA
ITERACAO (S/N) " ; RESP2$
220 IF RESP2$ <> "S" AND RESP2$ <> "N"
AND RESP2$ <> "n" THEN 210
300 REM TEMPERATURA INICIAL - IGUAL PARA TODOS OS
PONTOS
310 INPUT "TEMPERATURA INICIAL - IGUAL PARA TODOS
OS PONTOS " ; W
320 FOR I=1 TO N
330 T(I)=W
340 NEXT I
500 REM CONDICAOES DE CONTORNO
510 FOR I=1 TO A
520 J=I+1
530 IF I=A THEN J=1
540 PRINT "Condicao de contorno entre arestas
"; I ; " E " ; J
550 INPUT "Convecacao - CO      ,      Temp. Const. -
TC " ; CC$(I)
560 IF CC$(I) <> "TC" AND CC$(I) <> "tc" THEN 580
570 INPUT "temperatura " ; TC(I)
580 IF CC$(I) <> "CO" AND CC$(I) <> "co" THEN 600
590 INPUT "Coeficiente de Convecacao [H -
W/M2C] " ; AGA(I)
600 NEXT I
850 INPUT "ESTES DADOS ESTAO CORRETOS (S/N)
" ; RESP$
860 IF RESP$ <> "S" AND RESP$ <> "s" AND RESP$ <> "N"
AND RESP$ <> "n" THEN 850
870 IF RESP$ = "N" OR RESP$ = "n" THEN 100

```

Listagem:

```

910 IF SAIDA$="T" OR SAIDA$="f" THEN 1200
920 LPRINT "NUMERO DE PONTOS DA MALHA " ; N
930 LPRINT "CONDUTIVIDADE TERMICA (K) " ; K
940 LPRINT "CALOR ESPECIFICO (CE) " ; CE
950 LPRINT "DENSIDADE (RO) " ; RO
960 LPRINT "TEMPERATURA AMBIENTE " ; TAMB
970 LPRINT "INTERVALO TOTAL DE TEMPO (T) " ; T
980 LPRINT "INTERVALO PARCIAL DE TEMPO (DT) " ; DT
990 LPRINT "PASSO EM X (DX) " ; DX
1000 LPRINT "PASSO EM Y (DY) " ; DY
1010 LPRINT "CALOR GERADO " ; Q3L
1020 FOR I=1 TO A
1030 J=I+1
1040 IF I=A THEN J=1
1050 LPRINT "Condicao de contorno entre arestas
"; I ; " E " ; J
1060 IF CC$(I)="CO" OR CC$(I)="co" THEN LPRINT
"Conveccao . H=" ; AGA(I)
1070 IF CC$(I)="TC" OR CC$(I)="tc" THEN LPRINT
"Temperatura Constante . TC=" ; TC(I)
1080 NEXT I
1090 LPRINT "TEMPERATURA INICIAL - IGUAL PARA
TODOS OS PONTOS " ; W

```

Listagem:

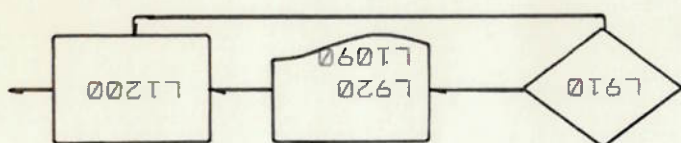


Diagrama:

neste módulo.

Comentário: este módulo foi desenvolvido para que o usuário tenha um relatório completo do exemplo por ele proposto. Todas as entradas pedidas no módulo anterior são impressas (linhas 920 a 1090), deste que em TELAI o usuário define o tipo de saída para impressora (verificação na linha 910).

Variáveis: nenhuma nova variável é implementada

3.4.3. Módulo Impressão dos dados do módulo anterior.

3.4.4. Módulo H e T por face.

Comentário: este módulo tem por objetivo permitir ao programa a utilização de diferentes valores do Coeficiente de Convecção e de Temperatura Constante, de acordo com as condições de contorno definidas anteriormente.

Variáveis:

$H(I,J)$ - melhorando a definição da variável $CC(I)$, esta variável guarda o valor do Coeficiente de Convecção para cada ponto que compoem a face em questão (W/m^2K).

$CTE(I,J)$ - idem para as faces a Temperatura Constante (K).

onde $1 < I < C$

e $1 < J < DD$

Listagem:

```

1200 REM H e T POR FACE
1205 FOR I=1 TO A
1210 J=I+1
1220 IF I=A THEN J=1
1230 IF NP(I,1)<ΔNP(J,1) THEN 1460
1240 IF NP(J,2)<NP(I,2) THEN 1350
1250 AUX=NP(I,2)-1
1260 AUX=AUX+1
1270 IF AUX=NP(J,2)+1 THEN 1700
1280 IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 1320
1290 IF TIPO$(NP(I,1),AUX)="F3"
THEN 1310
1300 TIPO$(NP(I,1)-1,AUX)="CO" :
H(NP(I,1)-1,AUX)=AGA(I)
GOTO 1260
1310 IF TIPO$(NP(I,1),AUX)="F3"
THEN 1320
1320 IF TIPO$(NP(I,1),AUX)="F3"
THEN 1340
1330 TIPO$(NP(I,1)-1,AUX)="TC" :
CTE(NP(I,1)-1,AUX)=TC(I)
GOTO 1260
1340 REM NP(J,2)<NP(I,2)
AUX=NP(J,2)-1
1360 AUX=AUX+1
1370 IF AUX=NP(I,2)+1 THEN 1700
1380 IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 1430
1400 IF TIPO$(NP(I,1),AUX)="F3" THEN
1420
1410 TIPO$(NP(I,1)+1,AUX)="CO" :
H(NP(I,1)+1,AUX)=AGA(I)
GOTO 1370
1430 IF TIPO$(NP(I,1),AUX)="F3" THEN
1450
1440 TIPO$(NP(I,1)+1,AUX)="TC" :
CTE(NP(I,1)+1,AUX)=TC(I)
GOTO 1370
1450 IF NP(I,2)<ΔNP(J,2) THEN PRINT "#1460
ERROR"
1470 IF NP(J,1)<NP(I,1) THEN 1580
1480 AUX=NP(I,1)-1
1490 AUX=AUX+1
1500 IF AUX=NP(J,1)+1 THEN 1700
1510 IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 1550
1520 IF TIPO$(AUX,NP(I,2))="F3"
THEN 1540
1530 TIPO$(AUX,NP(I,2)+1)="CO" :
H(AUX,NP(I,2)+1)=AGA(I)
GOTO 1490
1540

```

```

1550      IF TIPO$(AUX,NP(I,2))="F3"
1560      TIPO$(AUX,NP(I,2)+1)="TC" :
1570      THEN
1580      REM NP(J,1)<NP(I,1)
1590      AUX=NP(J,1)-1
1600      AUX=AUX+1
1610      IF AUX=NP(I,1)+1 THEN 1700
1620      IF CC$(I)<>"CO" AND CC$(I)<>"CO"
1630      THEN 1660
1640      IF TIPO$(AUX,NP(I,2))="F3" THEN
1650      TIPO$(AUX,NP(I,2)-1)="CO" :
1660      H(AUX,NP(I,2)-1)=AGA(I)
1670      GOTD 1600
1680      IF TIPO$(AUX,NP(I,2))="F3" THEN
1690      TIPO$(AUX,NP(I,2)-1)="TC" :
1700      CTE(AUX,NP(I,2)-1)=TC(I)
1710      GOTD 1600
1720      NEXT I

```

3.4.5. Módulo Correção da variável TIPO\$(I,J)

Comentário: a variável TIPO\$(I,J) é inicialmente desenvolvida no programa Cal. Agora, no programa CA2, tendo sido definidas as condições de contorno para cada face, surge a necessidade de corrigirmos esta variável, de forma que ela represente corretamente a realidade física, ou seja, por exemplo, para a seguinte figura proposta, um quadrado com duas divisões na direção x e duas na direção y, em que um dos lados tenha sido definido a temperatura constante, e os outros lados como submetidos à convecção. Para o lado à temperatura constante temos inicialmente em TIPO\$(I,J), a saída { F2, F1, F2 }. Lembramos que a variável TIPO\$(I,J) contém o tipo de cada ponto da malha.

Estes valores seriam alterados, neste módulo para { F1, IN, F1 }. Por decorrência as duas faces adjacentes submetidas à convecção teriam suas saídas alteradas de { F2, F1, F2 } para { F1, F1, F2 }. Observamos que para pontos tipo "F3", temos que ter ambas as faces definidas com a mesma condição de contorno.

Variáveis:

NTIPO\$(I,J) - guarda as alterações obtidas para TIPO\$(I,J) enquanto estas ainda não estão completas.

TOTAL - indica o tipo de configuração física proposta pelo usuário. É usada para decidir qual alteração precisa ser feita nos pontos onde TIPO\$(I,J) é igual a "F1" ou "F2".

FLAG - variável indicadora de estado.

Quando igual a um, TIPO\$(I,J) passa de "F3" para "IN".

Listaagem:

```

5000 REM CORRECAO DA VARIAVEL TIPO$(I,J)
5005 CLS
5010 FOR I=1 TO C
5020 FOR J=1 TO D
5022 TIPO$(I,J)=TIPO$(I,J)
5025 IF TIPO$(I,J)="TC" OR TIPO$(I,J)="CO"
OR TIPO$(I,J)="" THEN 5440
5030 TOTAL=CVI(TIPO$(I-
1,J))+CVI(TIPO$(I+1,J))+CVI(TIPO
$(I,J-1))
5040 IF TIPO$(I,J)<>"F2" THEN 5130
5050 IF TOTAL=65810 THEN NTIPO$(I,J)="F2"
5060 IF TOTAL=66834 THEN NTIPO$(I,J)="F2"
5070 IF TOTAL=66322 THEN NTIPO$(I,J)="F2"
5080 IF TOTAL=62755 THEN NTIPO$(I,J)="F1"
5090 IF TOTAL=63267 THEN NTIPO$(I,J)="F1"
5100 IF TOTAL=59700 THEN NTIPO$(I,J)="IN"
5130 IF TIPO$(I,J)<>"F1" THEN 5230
5170 IF TOTAL=65816 THEN NTIPO$(I,J)="F1"
5180 IF TOTAL=65560 THEN NTIPO$(I,J)="F1"
5190 IF TOTAL=66072 THEN NTIPO$(I,J)="F1"
5200 IF TOTAL=62505 THEN NTIPO$(I,J)="IN"
5210 IF TOTAL=62761 THEN NTIPO$(I,J)="IN"
5220 IF TOTAL=63017 THEN NTIPO$(I,J)="IN"
5225 IF TOTAL=63273 THEN NTIPO$(I,J)="IN"
5230 IF TIPO$(I,J)<>"F3" THEN 5440
5240 FLAG=0
5250 IF TIPO$(I-1,J-1)<>"TC" THEN 5270
5260 FLAG=1
5270 IF TIPO$(I-1,J+1)<>"TC" THEN 5290
5280 FLAG=1
5290 IF TIPO$(I+1,J-1)<>"TC" THEN 5310
5300 FLAG=1
5310 IF TIPO$(I+1,J+1)<>"TC" THEN 5330
5320 FLAG=1
5330 IF FLAG<>1 THEN 5360
5340 NTIPO$(I,J)="IN"
5350 GOTO 5440
5360 IF TIPO$(I-1,J-1)<>"CO" THEN 5380
5370 H(I-1,J)=H(I-1,J-1) : H(I,J-1)=H(I-1,J-
1)
5380 IF TIPO$(I-1,J+1)<>"CO" THEN 5400
5400 IF TIPO$(I+1,J-1)<>"CO" THEN 5420
5410 H(I+1,J)=H(I+1,J-1) : H(I,J-1)=H(I+1,J-
1)
5420 IF TIPO$(I+1,J+1)<>"CO" THEN 5440
5430 H(I+1,J)=H(I+1,J+1) :
5440 NEXT J
5450 NEXT I

```

```
5550 FOR I=1 TO C
    FOR J=1 TO D
        TIPO$(I,J)=NTIPO$(I,J)
    NEXT J
NEXT I
5590 NEXT I
5600 RETURN
```

3.4.6. Módulo Construção da tabela de caminhos

Comentário: cada ponto da nossa malha está ligado a quatro outros pontos. Chamamos de caminho à soma da variável $TIP0\$(I,J)$, no ponto que estamos analisando, com o valor da mesma variável, no ponto para onde vamos. São possíveis todas as combinações com "IN", "F1", "F2", "F3", "TC" e "CO", exceto aquelas começadas por "TC" e "CO". A variável $CAMIN\$(I,J)$ irá armazenar estas combinações (linhas 1780 a 1810), sendo que I se refere ao número do ponto e J a cada uma das quatro ligações, ou seja, "1 < I < SOMA" e "1 < J < 4". Paralelamente montamos a variável $L(I,J)$ (linhas 1820 a 1850), onde I e J tem os mesmos parâmetros descritos acima, que irá armazenar o número dos pontos que estão ligados a um determinado ponto.

Variáveis:

$CAMIN\$(I,J)$ - é a soma de $TIP0\$(I,J)$ do ponto que estamos analisando com $TIP0\$(I,J)$ dos pontos com os quais está ligado.

$L(I,J)$ - armazena o número dos pontos que estão ligados a cada ponto.

Diagrama:



Listagem:

```

1750 REM CONSTRUCAO DA TABELA DE 'CAMINHOS'
1760 FOR I=1 TO C
1770   FOR J=1 TO D
1780     CAMINH$(PI(I,J),1)=TIPO$(I,J)+TIPO$(I-1,J)
1790     CAMINH$(PI(I,J),2)=TIPO$(I,J)+TIPO$(I,J+1)
1800     CAMINH$(PI(I,J),3)=TIPO$(I,J)+TIPO$(I+1,J)
1810     CAMINH$(PI(I,J),4)=TIPO$(I,J)+TIPO$(I,J-1)
1820     L(PI(I,J),1)=PI(I-1,J)
1830     L(PI(I,J),2)=PI(I,J+1)
1840     L(PI(I,J),3)=PI(I+1,J)
1850     L(PI(I,J),4)=PI(I,J-1)
1860   NEXT J
1870 NEXT I

```

3.4.7. Núcleo do Método de Crank-Nicholson.

Comentário: esta estrutura, que inicia na linha 1900 e termina na linha 3830, é o ponto principal de nosso trabalho. É formada basicamente por duas instruções FOR-NEXT, uma inserida dentro da outra.

A mais externa representa a variação no tempo, e a mais interna a pesquisa sobre todos os pontos da malha. A sua estrutura é formada por:

Looping da variação no tempo

 Looping da variação de posição

 Montagem da matriz $M(I,J)$

 Resolução da matriz $M(I,J)$

 Cálculo dos fluxos instantâneos de calor

 Impressão em tela e impressora

 Verificação do nível de convergência

O looping de variação de posição nos fornece como saída a matriz $M(I,J)$, que é o resultado das equações de balanço utilizadas.

Neste looping percorremos todos os pontos da figura em estudo. Para cada um, analisamos as quatro ligações, que cada ponto possui na figura, através da variável $CAMIN\$(I,J)$, e para cada ligação verificamos o seu tipo.

Então passamos para o conjunto de equações que representam o tipo da ligação, através da variável CAMIN. Temos ao todo seis conjuntos de equações. Cada um destes conjuntos acrescenta à matriz $M(I,J)$ alguns valores. Ao término do looping de variação de posição, temos a matriz $M(I,J)$ completa (linhas 2100 a 2930). Podemos passar para a sua resolução, onde nos utilizamos do método de Gauss-Siedel, obtendo as temperaturas de todos os pontos da malha (linhas 3000 a 3300). Como estamos dentro do looping de variação do tempo, estas temperaturas se referem a um determinado tempo, dado pela variável TIME.

Tendo estas temperaturas, se determinado pelo usuário (linha 3500), podemos calcular e imprimir os fluxos instantâneos de calor nas faces da figura (linhas 6000 a 6990 do programa). Em seguida, imprimimos as temperaturas obtidas (linhas 3510 a 3740), e verificamos se foi atingido o nível de convergência exigido, ou seja, se a variação entre as temperaturas nos dois últimos instantes calculados é menor ou igual ao nível de convergência. Em caso negativo, passamos para a próxima iteração do looping de variação do tempo. Do contrário, terminamos a execução do programa com a mensagem de nível de convergência atingido (linhas 3750 a 3780).

Variáveis:

$M(I,J)$ - é a matriz dos coeficientes das equações termodinâmicas, onde $1 < I < N+1$ e $1 < J < N$.

CAMIN - utilizada como variável de distribuição pela intrusão ON CAMIN GOTO ...

D - relação entre área e distância entre pontos nodais, nos caso de condução, e área superficial, nos casos de convecção.

ITERA - número de iterações realizadas na resolução de $M(I,J)$.

EPSILON - nível de convergência exigido na resolução de $M(I,J)$.

AUX(I) - guarda os valores das temperaturas calculadas na iteração anterior (K).

NT(I) - guarda os valores das temperaturas calculadas na iteração atual (K).

VARIA - variação porcentual entre as temperaturas recém calculadas e as anteriores.

VAMAX - a maior variação de VARIA.

Listagem:

```

1900 CLS : REM INICIO DO LOOPING DE VARIACAO DO
TEMPO
1910 FLIG=1 : FLOG=0 : FLUG=0 : AUXDT=DT
1920 FOR TIME=DT TO T+20 STEP DT
1930 IF FLOG=1 THEN DT=AUXDT
1940 IF FLOG=1 THEN 1970
1950 DT=10 : TIME=FLIG*DT
1960 FLIG=FLIG+1
1970 FLUG=FLUG+1
2000 REM ZERANDO A MATRIZ M
2010   FOR I=1 TO N+1
2020     FOR J=1 TO N
2030       M(I,J)=0
2040     NEXT J
2050   NEXT I
2100 REM INICIO DO LOOPING DE VARIACAO DE
POSICAO
2110   FOR I=1 TO N
2150     FOR J=1 TO 4
2170     CAMIN= ((CAMIN$(I,J)="F1F3")*10
+(CAMIN$(I,J)="F1TC")*11 +(CAMIN$(I,J)
)="F1CO")*12+(CAMIN$(I,J)="F2IN")*13+(CAMIN$(I,J)=
"F2F1")*14+(CAMIN$(I,J)="F2F2"
)*15+(CAMIN$(I,J)="F2F3")*16+(CAMIN$(I,J)="F2TC")*
17+(CAMIN$(I,J)="F2CO")*18)*(-
1)
2175       IF CAMIN<=0 THEN 2190
2180     CAMIN= ((CAMIN$(I,J)="F3IN")*19+(CAMIN$(I,J)="F3F1"
)*20+(CAMIN$(I,J)
)="F3F2")*21+(CAMIN$(I,J)="F3F3")*22+(CAMIN$(I,J)=
"F3TC")*23+(CAMIN$(I,J)="F3CO"
)*24)*(-1)
2190     ON CAMIN GOTO
2200,2200,2300,2200,2400,2300,2200,2500,2500,2500,
26
00,2700,2300,2500,2300,2500,2300,2800,2200,2500,25
00,2300,2300,2300
2200     REM
2210     D=DY/DX : IF J=2 OR J=4 THEN D=DX/DY
2220     M(I,I)=M(I,I)+K*D
2230     M(L(I,J),I)=M(L(I,J),I)-K*D
2240     M(N+1,I)=M(N+1,I)-K*D*(T(I)-
T(L(I,J)))
2250     GOTO 2900
2300     LOCATE 20,10 : PRINT "#2300" : REM
IMPOSSIVEL FISICAMENTE
2400     REM
2410     D=DY/DX : IF J=2 OR J=4 THEN D=DX/DY
2420     M(I,I)=M(I,I)+K*D

```

```

2430      REM M(L(I,J),I)=M(L(I,J),I) -Δ AQUI
L(I,J)=-1
2440      M(N+1,I)=M(N+1,I)-K*D*(T(I)-
CTE(NP(L(I,J),1),NP(L(I,J),2)))+K*D*CT
E(NP(L(I,J),1),NP(L(I,J),2))
2450      GOTO 2900
2500      REM
2510      D=DY/(DX*2) : IF J=2 OR J=4 THEN
D=DX/(DY*2)
2520      M(I,I)=M(I,I)+K*D
2530      M(L(I,J),I)=M(L(I,J),I)-K*D
2540      M(N+1,I)=M(N+1,I)-K*D*(T(I)-
T(L(I,J)))
2550      IF TIPO$(NP(I,1),NP(I,2))<Δ"F3" THEN
2900
2560      M(I,I)=M(I,I)+H(NP(L(I,J),1),NP(L(I,J),2))*D
2570      REM M(L(I,J),I)=M(L(I,J),I)
2580      M(N+1,I)=M(N+1,I)-
H(NP(L(I,J),1),NP(L(I,J),2))*D*(T(I)-TAMB)+H(NP(
L(I,J),1),NP(L(I,J),2))*D*TAMB
2590      GOTO 2900
2600      REM
2610      D=DY/(DX*2) : IF J=2 OR J=4 THEN
D=DX/(DY*2)
2620      M(I,I)=M(I,I)+K*D
2630      REM M(L(I,J),I)=M(L(I,J),I) -Δ AQUI
L(I,J)=-1
2640      M(N+1,I)=M(N+1,I)-K*D*(T(I)-
CTE(NP(L(I,J),1),NP(L(I,J),2)))+K*D*CT
E(NP(L(I,J),1),NP(L(I,J),2))
2650      GOTO 2900
2700      REM
2710      D=DY : IF J=2 OR J=4 THEN D=DX
2720      M(I,I)=M(I,I)+H(NP(L(I,J),1),NP(L(I,J),2))*D
2730      M(L(I,J),I)=M(L(I,J),I)
2740      M(N+1,I)=M(N+1,I)-
H(NP(L(I,J),1),NP(L(I,J),2))*D*(T(I)-TAMB)+H(NP(
L(I,J),1),NP(L(I,J),2))*D*TAMB
2750      GOTO 2900
2800      REM
2810      D=DY/2 : IF J=2 OR J=4 THEN D=DX/2
2820      M(I,I)=M(I,I)+H(NP(L(I,J),1),NP(L(I,J),2))*D
2830      M(L(I,J),I)=M(L(I,J),I)
2840      M(N+1,I)=M(N+1,I)-
H(NP(L(I,J),1),NP(L(I,J),2))*D*(T(I)-TAMB)+H(NP(
L(I,J),1),NP(L(I,J),2))*D*TAMB
2850      GOTO 2900
2900      NEXT J
2910      M(I,I)=M(I,I)+(2*RO*CE)/(DT*K)
2920      M(N+1,I)=M(N+1,I)+(2*RO*CE*T(I))/(DT*K)+2*Q3L

```

```

2930     NEXT I
3000     REM Resolucao da matriz M(I,J)
3010     ITERA=1
3020     EPSILON =.0001
3030     VAMAX=-10
3040     FOR I=1 TO N
3050         AUX(I)=T(I)
3060     NEXT I
3100     REM Gauss-Siedel
3110     FOR I=1 TO N
3120         NT(I)=M(N+1,I)
3130         FOR J=1 TO N
3140             IF J=I THEN 3160
3150             NT(I)=NT(I)-M(J,I)*T(J)
3160         NEXT J
3170         NT(I)=NT(I)/M(I,I)
3180     NEXT I
3190     FOR I=1 TO N
3200         T(I)=NT(I)
3210     NEXT I
3220     FLAG=0
3230     FOR I=1 TO N
3240         VARIA=(100*ABS(NT(I)-AUX(I)))/AUX(I)
3245         IF VAMAX<VARIA THEN 3250
3246         VAMAX=VARIA
3250         IF VARIA<EPSILON THEN 3270
3260         FLAG=1
3270     NEXT I
3280     ITERA=ITERA+1
3290     IF ITERA<15 THEN 3500
3300     IF FLAG=0 THEN
3500         IF RESP2$="S" OR RESP2$="s" THEN GOSUB6000
3510         LOCATE 1,1 : PRINT "
"
3520         LOCATE 1,1 : PRINT"TIME :";TIME;"
VARIACAO MAXIMA :";VAMAX
3530     FOR I=1 TO C
3540         FOR J=1 TO DD
3550             LOCATE I*2,J*9+1 : PRINT PI(I,J);
3560             LOCATE I*2+1,J*9+1 : PRINT USING
"###.####";T(PI(I,J))
3570         NEXT J
3580     NEXT I
3590     IF FLAG<10 THEN 3750
3600     FLAG=0
3610     IF SAIDA$="T" OR SAIDA$="t" THEN 3750
3620     LPRINT"VARIACAO MAXIMA :";VAMAX
3630     FOR I=1 TO C
3640         FOR J=1 TO DD
3650             LPRINT USING "###";PI(I,J);
3660             LPRINT "
";
3670         NEXT J
3680         LPRINT"*"
3690         FOR J=1 TO DD

```

```

3700          LPRINT USING "###.###";T(PI(I,J));
3710          LPRINT " ";
3720      NEXT J
3730      LPRINT "**" : LPRINT
3740  NEXT I
3745      LPRINT CHR$(11)
3750      IF VAMAXΔ.01 THEN 3790
3750      IF VAMAXΔ.01 THEN 3790
3760      TIME=T+20 : LOCATE 24,5 : PRINT "NIVEL DE
CONVERGENCIA ATINGIDO"
3770      IF SAIDA$="T" OR SAIDA$="t" THEN 3790
3780      LPRINT "NIVEL DE CONVERGENCIA ATINGIDO"
3785      END
3790      IF FLOG=1 THEN 3820
3800      IF FLIGΔ20 THEN FLOG=1
3810      IF FLIG≤20 THEN 1950
3820  NEXT TIME
3830  END

6000  REM CALCULO DOS FLUXOS INSTANTANEOS DE CALOR
POR FACE
6010  FOR I=1 TO A
6020  Q(I)=0
6030  NEXT I
6040  FOR I=1 TO A
6050      J=I+1
6060      IF I=A THEN J=1
6070      IF NP(I,1)<ΔNP(J,1) THEN 6340
6080      IF NP(J,2)<NP(I,2) THEN 6210
6090      AUX=NP(I,2)-1
6100      AUX=AUX+1
6110      IF AUX=NP(J,2)+1 THEN
6120          IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 6160
6130          IF TIPO$(NP(I,1),AUX)="F3"
THEN 6100
6135          D=DY : IF
TIPO$(NP(I,1),AUX)="F2" THEN D=D/2
6140          Q(I)=Q(I)+H(NP(I,1)-
1,AUX)*D*(NT(PI(NP(I,1),AUX))-TAMB)
6150          GOTO 6100
6160          IF TIPO$(NP(I,1),AUX)="F3"
THEN 6100
6170          D=DY/DX
6180          IF TIPO$(NP(I,1),AUX)="F1"
THEN D=D/2
6190          Q(I)=Q(I)+K*D*(NT(PI(NP(I,1),AUX))-CTE(NP(I,1)-
1,AUX))
6200          GOTO 6100
6210      REM NP(J,2)<NP(I,2)
6220      AUX=NP(J,2)-1
6230      AUX=AUX+1
6240      IF AUX=NP(I,2)+1 THEN 6610

```

```

6250          IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 6290
6260          IF TIPO$(NP(I,1),AUX)="F3" THEN
6230
6265          D=DY : IF
TIPO$(NP(I,1),AUX)="F2" THEN D=D/2
6270
Q(I)=Q(I)+H(NP(I,1)+1,AUX)*D*(NT(PI(NP(I,1),AUX))-
TAMB)
6280          GOTO 6230
6290          IF TIPO$(NP(I,1),AUX)="F3" THEN
6230
6300          D=DY/DX
6310          IF TIPO$(NP(I,1),AUX)="F1"
THEN D=D/2
6320
Q(I)=Q(I)+K*D*(NT(PI(NP(I,1),AUX))-
CTE(NP(I,1)+1,AUX))
6330          GOTO 6230
6340          IF NP(I,2)<ΔNP(J,2) THEN PRINT "#6340
ERR0"
6350          IF NP(J,1)<NP(I,1) THEN 6480
6360          AUX=NP(I,1)-1
6370          AUX=AUX+1
6380          IF AUX=NP(J,1)+1 THEN 6610
6390          IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 6430
6400          IF TIPO$(AUX,NP(I,2))="F3"
THEN 6370
6405          D=DX : IF
TIPO$(AUX,NP(I,2))="F2" THEN D=D/2
6410
Q(I)=Q(I)+H(AUX,NP(I,2)+1)*D*(NT(PI(AUX,NP(I,2)))-
TAMB)
6420          GOTO 6370
6430          IF TIPO$(AUX,NP(I,2))="F3"
THEN 6370
6440          D=DX/DY
6450          IF TIPO$(AUX,NP(I,2))="F1"
THEN D=D/2
6460
Q(I)=Q(I)+K*D*(NT(PI(AUX,NP(I,2)))-
CTE(AUX,NP(I,2)+1))
6470          GOTO 6370
6480          REM NP(J,1)<NP(I,1)
6490          AUX=NP(J,1)-1
6500          AUX=AUX+1
6510          IF AUX=NP(I,1)+1 THEN 6610
6520          IF CC$(I)<Δ"CO" AND CC$(I)<Δ"co"
THEN 6560
6530          IF TIPO$(AUX,NP(I,2))="F3" THEN
6500
6535          D=DX : IF
TIPO$(AUX,NP(I,2))="F2" THEN D=D/2

```

```

6540      Q(I)=Q(I)+H(AUX,NP(I,2)-
1)*D*(NT(PI(AUX,NP(I,2)))-TAMB)
6550      GOTO 6500
6560      IF TIPO$(AUX,NP(I,2))="F3" THEN
6500
6570      D=DX/DY
6580      IF TIPO$(AUX,NP(I,2))="F1"
THEN D=D/2
6590
Q(I)=Q(I)+K*D*(NT(PI(AUX,NP(I,2)))-
CTE(AUX,NP(I,2)-1))
6600      GOTO 6500
6600      GOTO 6500
6610 NEXT I
6620      LOCATE 18,5 : PRINT "FACE          FLUXO
TOTAL INSTANTANEO"
6630 FOR I=1 TO A
6640      LOCATE 19+I,5
6650      J=I+1
6660      IF I=A THEN J=1
6670      PRINT " # ";I;"-";J;" : ";Q(I);" ,
";CC$(I);
6680 NEXT I
6690 IF FLUG<10 THEN 6990
6700 LPRINT CHR$(27);CHR$(66);CHR$(9+A+C)
6710 LPRINT "TIME :";TIME
6720 LPRINT
6730 LPRINT "FACE          FLUXO TOTAL INSTANTANEO"
6740 LPRINT
6750 FOR I=1 TO A
6760      J=I+1
6770      IF I=A THEN J=1
6780      LPRINT I;"-";J;" : ";Q(I);" , ";CC$(I)
6790 NEXT I
6800 LPRINT
6800 LPRINT
6990 RETURN

```

3.4.8. Anexos do módulo anterior.

Comentário: foram acrescentadas algumas instruções à estrutura apresentada no módulo anterior, para executar duas tarefas:

1. Fazer com que o programa execute vinte iterações de intervalo de tempo (TIME) igual a 10 segundos, no início do módulo anterior.
2. Obter uma impressão em papel para cada dez iterações.

Variáveis:

de 1:

FLIG - contador das vinte iterações iniciais.

FLOG - indicador de estado. 0 quando ainda estamos nas iterações iniciais, e 1 quando já passamos esta fase.

AUXDT - armazena o valor da variável DT, pelo definida pelo usuário, enquanto trabalhamos com DT igual a dez nas iterações iniciais.

de 2:

FLUG - contador das dez iterações antes de imprimirmos um resultado.

Diagrama de 1:

Inicialização de variáveis	L1910
Looping do tempo	L1920
Correção de DT (FLOG=1)	L1930
Desvio (FLOG=1)	L1940
Cálculo de TIME	L1950
Programa	L2000 L3780
Desvio (FLOG=1)	L3790
Correção de FLIG (FLOG=1)	L3800
Desvio (FLIG<=20)	L3810
Fim do Looping	L3820

Diagrama de 2:

Inicialização de variáveis	L1910
Looping do tempo	L1920
Incremento do contador	L1970
Programa	L2000 L3580
Desvio (FLUG<10)	L3590
Zeragem do contador	L3600
Programa	L3610 L3780
Fim do Looping	L3820

Observação: conferir estes diagramas na listagem do módulo anterior.

4. Como utilizar o programa.

Saindo de uma situação inicial, que seja, o microcomputador em operação, prompt piscando.

Como o programa foi desenvolvido na linguagem Basic, a primeira tarefa a realizar é a de acessar o Gwbasic digitando a palavra e teclando Enter. Em seguida devemos carregar o programa de inicialização digitando LOAD"TELA1 e teclando Enter. Aparecerá, então, uma tela de apresentação, e a partir daí começarão a aparecer perguntas a serem respondidas. São elas:

"Tipo de saída desejada (Tela ou Impressora e tela)"

A resposta a esta pergunta definirá o tipo de apresentação dos resultados obtidos.

Para apresentação apenas em tela tecle "T" ou "t", e para apresentação em tela e impressora tecle "I" ou "i".

Caso seja escolhida a saída em tela e impressora, o seguinte pedido será feito:

"Entre um nome para o exemplo"

O nome proposto pelo usuário será impresso na primeira folha de saída.

As perguntas feitas a seguir definem a figura com que o programa irá trabalhar e a forma como esta será discretizada.

Para ilustrar estas perguntas vamos tomar um exemplo:

Estudo de uma aleta retangular de lados 10×30 cm, com base a temperatura constante e lados e ponta submetidas a convecção.

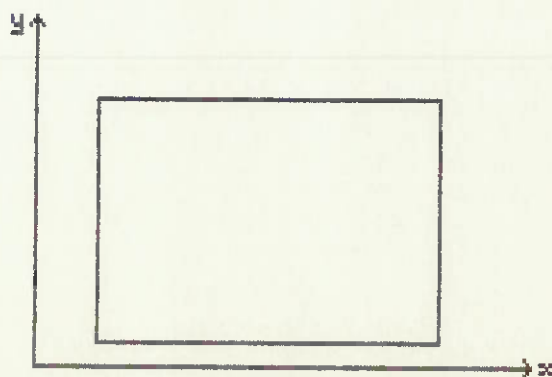
Observamos que é necessário que, desde o início, tenhamos em mente as condições de contorno que serão impostas à figura, devido a característica do programa que, para lados submetidos a temperatura constante, acrescenta, ou faz os cálculos como se a figura proposta tivesse uma divisão a mais.

A primeira pergunta:

"Número de divisões (X,Y)"

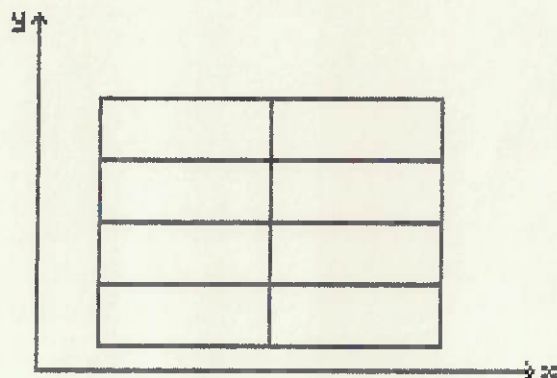
Sugere que desenhemos a figura do nosso exemplo em um plano cartesiano devidamente orientado nas direções X e Y.

FIGURA 5



Supondo o número de divisões em X, Y igual a 2, 4 teríamos a figura dividida da seguinte forma:

FIGURA 6



A esta figura denominaremos de malha proposta pelo usuário. Cada cruzamento entre as linhas horizontais e verticais será considerado um ponto nodal e terá a sua temperatura calculada em função do tempo.

A segunda pergunta:

"Número de arestas"

Para o nosso exemplo a resposta é quatro (4).

A seguir o programa envia a seguinte mensagem à tela:

" A seguir serão pedidas as coordenadas das arestas. Estas deverão coincidir com a malha proposta pelo número de divisões em X e Y.

Não alinhe três ou mais pontos.

Obedecer o sentido horário."

Seguindo estas instruções poderíamos digitar a sequência de coordenadas (1,1.06) (1,1.3) (1.1,1.3) (1.1,1.06) para o nosso exemplo.

Lembrando a observação feita alguns parágrafos atrás, corrigimos as coordenadas do primeiro e último pontos de (1,1) para (1,1.06) e (1.1,1) para (1.1,1.06) para satisfazer tanto o exemplo quanto a característica do programa. Estamos imaginando que o lado formado pelos primeiro e último pontos será a base da aleta e, portanto, estará submetido à condição de temperatura constante.

São mostrados os valores digitados e pede-se uma confirmação. Em caso afirmativo o programa seguirá adiante, caso contrário voltaremos à primeira pergunta.

Neste ponto, o programa passa a calcular a parte física do problema, ou seja, a malha de pontos definida anteriormente pelo usuário. Ao término envia à tela do microcomputador a saída obtida e pede confirmação. Em caso afirmativo o programa seguirá adiante, caso contrário é encerrada a sessão.

Terminada esta primeira etapa, passamos a uma nova sequência de perguntas. Estes dados deverão ter suas unidades expressas, bem como todos os outros, no Sistema Internacional, como indicado.

Os valores que seguem são os adotados para nosso exemplo.

"Condutividade térmica ($W/M^{\circ}C$)"	$K=50$
"Calor Específico ($J/Kg^{\circ}C$)"	$CE=500$
"Densidade (Kg/M^3)"	$RO=7500$
"Temperatura ambiente ($^{\circ}C$)"	$T_{amb}=20$
"Intervalo total de tempo (s)"	$t=1E20$
"Intervalo parcial de tempo (s)"	$Dt=2000$
"Calor gerado internamente (W)"	$Q'''=0$
"Temperatura inicial ($^{\circ}C$)"	$T_{INIC}=50$
"Condição de contorno entre aresta 1 e 2"	Convecção
"Coeficiente de Convecção($W/M^2^{\circ}C$)"	$H=100$
"Condição de contorno entre aresta 2 e 3"	Convecção
"Coeficiente de Convecção($W/M^2^{\circ}C$)"	$H=99$
"Condição de contorno entre aresta 3 e 4"	Convecção
"Coeficiente de Convecção($W/M^2^{\circ}C$)"	$H=99$
"Condição de contorno entre aresta 4 e 1"	Temp. Cte.
"Temperatura na face($^{\circ}C$)"	$CTE=50$

Além destas entradas, a seguinte pergunta é feita:

"Calcular fluxos instantâneos de calor (S/N)"

Responderemos positivamente a esta pergunta (S).

Após digitarmos estes dados vem a pergunta:

"Estes dados estão corretos (S/N)"

Respondendo "N" ou "n" retornamos ao início desta série de dados.

Caso afirmativo passaremos à fase de cálculo iterativo. Permaneceremos no mesmo até que a condição de convergência do programa seja satisfeita (ver 3.4 CA2).

Apresentaremos a seguir, em anexo, o resultado ou, a saída obtida para o exemplo que propusemos neste capítulo.

Programacao do Metodo de Crank-Nicholson

Aleta Retangular (10x30)

02-05-1992 21:22:34

PONTO	X	Y
1	1	1.06
2	1	1.3
3	1.1	1.3
4	1.1	1.06

NUMERO DE PONTOS DA MALHA 15
CONDUTIVIDADE TERMICA (K) 50
CALOR ESPECIFICO (CE) 500
DENSIDADE (RO) 7500
TEMPERATURA AMBIENTE 20
INTERVALO TOTAL DE TEMPO (T) 1E+20
INTERVALO PARCIAL DE TEMPO (DT) 2000
PASSO EM X (DX) 5.0000001E-02
PASSO EM Y (DY) .06
CALOR GERADO 0
Condicao de contorno entre arestas 1 E 2
Conveccao . H= 100
Condicao de contorno entre arestas 2 E 3
Conveccao . H= 99
Condicao de contorno entre arestas 3 E 4
Conveccao . H= 99
Condicao de contorno entre arestas 4 E 1
Temperatura Constante . TC= 50
TEMPERATURA INICIAL - IGUAL P/ TODOS OS PONTOS 50

TIME : 100

FACE FLUXO TOTAL INSTANTANEO

1 - 2	803.8417	, CO
2 - 3	294.9682	, CO
3 - 4	795.8611	, CO
4 - 1	-10.12135	, TC

VARIACAO MAXIMA : 4.458852E-02

1	6	9	12	2
49.7729	49.7699	49.7699	49.7699	49.7821

5	7	10	13	15
49.9830	49.9826	49.9825	49.9780	49.8068

4	8	11	14	3
49.7752	49.7722	49.7721	49.7722	49.7833

TIME : 200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	798.1241	, CO
2 - 3	292.9843	, CO
3 - 4	790.2528	, CO
4 - 1	-20.47913	, TC

VARIACAO MAXIMA : 4.317951E-02

1	6	9	12	2
49.5681	49.5567	49.5565	49.5561	49.5667

5	7	10	13	15
49.9383	49.9351	49.9344	49.9192	49.6209

4	8	11	14	3
49.5723	49.5611	49.5608	49.5604	49.5691

TIME : 20200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	416.7947 , CO
2 - 3	108.4652 , CO
3 - 4	412.98 , CO
4 - 1	-651.9882 , TC

VARIACAO MAXIMA : 2.421597

1	6	9	12	2
41.7303	36.7861	33.7205	31.8414	30.7752
5	7	10	13	15
42.6145	37.5586	34.3317	32.3325	31.1318
4	8	11	14	3
41.7453	36.8010	33.7334	31.8527	30.7856

TIME : 40200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	355.602	, CO
2 - 3	76.18255	, CO
3 - 4	352.3396	, CO
4 - 1	-736.0004	, TC

VARIACAO MAXIMA : .4656941

1	6	9	12	2
40.7334	34.8908	31.1124	28.7720	27.5167

5	7	10	13	15
41.5955	35.6206	31.6658	29.2034	27.8702

4	8	11	14	3
40.7475	34.9039	31.1228	28.7803	27.5238

TIME : 60200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	345.3131	, CO
2 - 3	70.74991	, CO
3 - 4	342.1436	, CO
4 - 1	-750.1108	, TC

VARIACAO MAXIMA : 8.023897E-02

1	6	9	12	2
40.5660	34.5724	30.6740	28.2557	26.9683

5	7	10	13	15
41.4243	35.2949	31.2176	28.6769	27.3213

4	8	11	14	3
40.5800	34.5851	30.6839	28.2635	26.9749

TIME : 80200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	343.5827	, CO
2 - 3	69.83624	, CO
3 - 4	340.4288	, CO
4 - 1	-752.4838	, TC

VARIACAO MAXIMA : 1.354983E-02

1	6	9	12	2
40.5379	34.5188	30.6002	28.1688	26.8760
5	7	10	13	15
41.3955	35.2402	31.1422	28.5884	27.2290
4	8	11	14	3
40.5518	34.5315	30.6101	28.1766	26.8826

NIVEL DE CONVERGENCIA ATINGIDO

Conclusão a respeito do exemplo:

As temperaturas da malha convergiram para uma situação de equilíbrio, ou seja, podemos dizer que a figura, com a qual trabalhamos, saindo de uma situação transitória chegou a uma situação de regime permanente, ou equilíbrio térmico. Podemos confirmar esta afirmação, fazendo a somatória dos fluxos instantâneos de calor obtidos para a última iteração impressa antes de ser atingido o nível de convergência, e verificar que o valor calculado é muito próximo de zero.

Isto nos dá crédito para afirmar que, dentro de suas limitações, o programa nos fornece saídas coerentes e aceitáveis.

A seguir anexaremos outros exemplos.

Continuaremos estudando o exemplo anterior variando apenas o comprimento da aleta.

Programacao do Metodo de Crank-Nicholson

Aleta Retangular (10x10)

02-05-1992 22:50:43

PONTO	X	Y
1	1	1.02
2	1	1.1
3	1.1	1.1
4	1.1	1.02

NUMERO DE PONTOS DA MALHA 15
CONDUTIVIDADE TERMICA (K) 50
CALOR ESPECIFICO (CE) 500
DENSIDADE (RO) 7500
TEMPERATURA AMBIENTE 20
INTERVALO TOTAL DE TEMPO (T) 1E+20
INTERVALO PARCIAL DE TEMPO (DT) 2000
PASSO EM X (DX) 5.0000001E-02
PASSO EM Y (DY) 2.0000001E-02
CALOR GERADO 0
Condicao de contorno entre arestas 1 E 2
Conveccao . H= 100
Condicao de contorno entre arestas 2 E 3
Conveccao . H= 99
Condicao de contorno entre arestas 3 E 4
Conveccao . H= 99
Condicao de contorno entre arestas 4 E 1
Temperatura Constante . TC= 50
TEMPERATURA INICIAL - IGUAL P/ TODOS OS PONTOS 50

TIME : 100

FACE FLUXO TOTAL INSTANTANEO

1 - 2	269.2351	, CO
2 - 3	295.4243	, CO
3 - 4	266.5495	, CO
4 - 1	-9.649036	, TC

VARIACAO MAXIMA : 3.394079E-02

1	6	9	12	2
49.9243	49.9213	49.9212	49.9189	49.8635
5	7	10	13	15
49.9981	49.9980	49.9973	49.9845	49.8180
4	8	11	14	3
49.9251	49.9222	49.9220	49.9197	49.8639

TIME : 200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	268.4927	, CO
2 - 3	293.9962	, CO
3 - 4	265.8208	, CO
4 - 1	-18.76354	, TC

VARIACAO MAXIMA : 2.967215E-02

1	6	9	12	2
49.8561	49.8451	49.8441	49.8353	49.7313

5	7	10	13	15
49.9931	49.9917	49.9874	49.9460	49.6614

4	8	11	14	3
49.8576	49.8467	49.8456	49.8369	49.7321

TIME : 20200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	219.2416	, CO
2 - 3	217.9763	, CO
3 - 4	217.1875	, CO
4 - 1	-579.8315	, TC

VARIACAO MAXIMA : .5177002

1	6	9	12	2
47.3400	45.3047	43.7182	42.4836	41.5485

5	7	10	13	15
48.0162	46.2318	44.7114	43.4626	42.4782

4	8	11	14	3
47.3502	45.3199	43.7356	42.5017	41.5663

TIME : 40200

FACE	FLUXO TOTAL INSTANTANEO
1 - 2	214.7704 , CO
2 - 3	211.1175 , CO
3 - 4	212.7564 , CO
4 - 1	-631.7535 , TC

VARIACAO MAXIMA : 4.891569E-02

1	6	9	12	2
47.1254	44.8947	43.1494	41.8060	40.8194
5	7	10	13	15
47.8156	45.8491	44.1828	42.8405	41.8221
4	8	11	14	3
47.1353	44.9095	43.1662	41.8233	40.8364

NIVEL DE CONVERGENCIA ATINGIDO

Programacao do Metodo de Crank-Nicholson

Aleta Retangular (10x50)

02-06-1992 17:16:39

PONTO	X	Y
1	1	1.1
2	1	1.5
3	1.1	1.5
4	1.1	1.1

NUMERO DE PONTOS DA MALHA 15
CONDUTIVIDADE TERMICA (K) 50
CALOR ESPECIFICO (CE) 500
DENSIDADE (RO) 7500
TEMPERATURA AMBIENTE 20
INTERVALO TOTAL DE TEMPO (T) 1E+20
INTERVALO PARCIAL DE TEMPO (DT) 2000
PASSO EM X (DX) 5.0000001E-02
PASSO EM Y (DY) 9.999999E-02
CALOR GERADO 0
Condicao de contorno entre arestas 1 E 2
Conveccao . H= 100
Condicao de contorno entre arestas 2 E 3
Conveccao . H= 99
Condicao de contorno entre arestas 3 E 4
Conveccao . H= 99
Condicao de contorno entre arestas 4 E 1
Temperatura Constante . TC= 50
TEMPERATURA INICIAL - IGUAL P/ TODOS OS PONTOS 50

TIME : 100

FACE FLUXO TOTAL INSTANTANEO

1 - 2	1333.601	, CO
2 - 3	294.5513	, CO
3 - 4	1320.419	, CO
4 - 1	-10.36134	, TC

VARIACAO MAXIMA : 7.114536E-02

1	6	9	12	2
49.6290	49.6260	49.6260	49.6265	49.7051

5	7	10	13	15
49.9547	49.9540	49.9540	49.9516	49.7993

4	8	11	14	3
49.6326	49.6297	49.6297	49.6302	49.7070

TIME : 200

FACE	FLUXO TOTAL INSTANTANEO
1 - 2	1318.911 , CO
2 - 3	292.1077 , CO
3 - 4	1306.009 , CO
4 - 1	-21.29107 , TC

VARIACAO MAXIMA : .0647715

1 49.3035	6 49.2925	9 49.2924	12 49.2938	2 49.4177
5 49.8415	7 49.8365	10 49.8363	13 49.8291	15 49.5920
4 49.3103	8 49.2993	11 49.2992	14 49.3006	3 49.4215

TIME : 20200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	443.4343	, CO
2 - 3	57.14002	, CO
3 - 4	439.3917	, CO
4 - 1	-632.3805	, TC

VARIACAO MAXIMA : 3.191499

1	6	9	12	2
36.9675	30.6863	27.6273	26.2158	25.6930

5	7	10	13	15
37.7305	31.1868	27.9590	26.4607	25.8477

4	8	11	14	3
36.9810	30.6963	27.6346	26.2218	25.6985

TIME : 40200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	346.938	, CO
2 - 3	25.52751	, CO
3 - 4	343.7674	, CO
4 - 1	-679.5777	, TC

VARIACAO MAXIMA : .4578777

1	6	9	12	2
36.0361	28.9152	25.1800	23.3017	22.5215

5	7	10	13	15
36.7745	29.3647	25.4396	23.4630	22.6344

4	8	11	14	3
36.0488	28.9235	25.1849	23.3048	22.5239

TIME : 60200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	335.4301	, CO
2 - 3	21.70915	, CO
3 - 4	332.3635	, CO
4 - 1	-685.1103	, TC

VARIACAO MAXIMA : 5.611801E-02

1	6	9	12	2
35.9272	28.7064	24.8887	22.9516	22.1383
5	7	10	13	15
36.6622	29.1494	25.1393	23.1026	22.2464
4	8	11	14	3
35.9397	28.7145	24.8933	22.9544	22.1403

NIVEL DE CONVERGENCIA ATINGIDO

Programacao do Metodo de Crank-Nicholson

Aleta Retangular (10x100)

02-05-1992 23:20:43

PONTO	X	Y
1	1	1.2
2	1	2
3	1.1	2
4	1.1	1.2

NUMERO DE PONTOS DA MALHA 15
CONDUTIVIDADE TERMICA (K) 50
CALOR ESPECIFICO (CE) 500
DENSIDADE (RO) 7500
TEMPERATURA AMBIENTE 20
INTERVALO TOTAL DE TEMPO (T) 1E+20
INTERVALO PARCIAL DE TEMPO (DT) 2000
PASSO EM X (DX) 5.0000001E-02
PASSO EM Y (DY) .2
CALOR GERADO 0
Condicao de contorno entre arestas 1 E 2
Conveccao . H= 100
Condicao de contorno entre arestas 2 E 3
Conveccao . H= 99
Condicao de contorno entre arestas 3 E 4
Conveccao . H= 99
Condicao de contorno entre arestas 4 E 1
Temperatura Constante . TC= 50
TEMPERATURA INICIAL - IGUAL P/ TODOS OS PONTOS 50

TIME : 100

FACE FLUXO TOTAL INSTANTANEO

1 - 2	2638.699	, CO
2 - 3	293.4997	, CO
3 - 4	2612.891	, CO
4 - 1	-10.7791	, TC

VARIACAO MAXIMA : .1299679

1	6	9	12	2
49.2954	49.2926	49.2926	49.2934	49.5218

5	7	10	13	15
49.8388	49.8376	49.8376	49.8370	49.7692

4	8	11	14	3
49.3023	49.2995	49.2994	49.3003	49.5255

TIME : 200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	2587.055	, CO
2 - 3	289.9006	, CO
3 - 4	2562.185	, CO
4 - 1	-22.46381	, TC

VARIACAO MAXIMA : .1149892

1	6	9	12	2
48.7107	48.7004	48.7003	48.7031	49.0765

5	7	10	13	15
49.4863	49.4786	49.4785	49.4785	49.4858

4	8	11	14	3
48.7224	48.7122	48.7121	48.7148	49.0833

TIME : 20200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	329.078	, CO
2 - 3	15.22862	, CO
3 - 4	326.0818	, CO
4 - 1	-511.3157	, TC

VARIACAO MAXIMA : 2.430271

1	6	9	12	2
29.3292	23.3564	21.6730	21.3349	21.5209

5	7	10	13	15
29.7615	23.5004	21.7272	21.3672	21.5549

4	8	11	14	3
29.3373	23.3595	21.6746	21.3362	21.5223

TIME : 40200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	262.6868	, CO
2 - 3	1.870625	, CO
3 - 4	260.2923	, CO
4 - 1	-518.7084	, TC

VARIACAO MAXIMA : .10995

1	6	9	12	2
29.0302	22.7985	20.8915	20.3217	20.1851

5	7	10	13	15
29.4692	22.9396	20.9344	20.3343	20.1927

4	8	11	14	3
29.0381	22.8011	20.8923	20.3220	20.1853

NIVEL DE CONVERGENCIA ATINGIDO

Verificamos que a distribuição das temperaturas bidimensional se aproxima do comportamento do caso unidimensional, com o aumento do comprimento. De fato, para o exemplo "Aleta Retangular (10cm x 10cm)" encontramos uma variação de no mínimo 0.8 °C, nas seções transversais ao comprimento da aleta, já para o exemplo "Aleta Retangular (10cm x 100cm)" temos, para as mesmas seções, variação mínima da ordem de um centésimo de grau. Ainda podemos verificar que, como no primeiro exemplo estudado, a somatória dos fluxos de calor obtidos na última iteração impressa, para cada um dos tres exemplos, continua tendendo a zero, mostrando que, também nesses exemplos, convergimos para uma situação de equilíbrio. Portanto, novamente, a saída, obtida pelo programa, se mostra coerente com a realidade física.

Como último exemplo, colocaremos o resfriamento de uma peça. Saindo de uma temperatura inicial, observaremos que as temperaturas no interior do corpo serão sempre maiores do que as externas, indicando um fluxo de calor de dentro para fora em cada face. Como este fluxo depende do diferencial de temperaturas, no final do processo, cada fluxo de cada face deverá tender a zero.

Programacao do Metodo de Crank-Nicholson

Resfriamento de um quadrado de lado 20 cm

02-07-1992 12:27:39

PONTO	X	Y
1	1	1
2	1	1.2
3	1.2	1.2
4	1.2	1

NUMERO DE PONTOS DA MALHA 25
CONDUTIVIDADE TERMICA (K) 206
CALOR ESPECIFICO (CE) 896
DENSIDADE (RO) 2707
TEMPERATURA AMBIENTE 20
INTERVALO TOTAL DE TEMPO (T) 1E+20
INTERVALO PARCIAL DE TEMPO (DT) 100
PASSO EM X (DX) 5.000001E-02
PASSO EM Y (DY) 5.000001E-02
CALOR GERADO 0
Condicao de contorno entre arestas 1 E 2
Conveccao . H= 400
Condicao de contorno entre arestas 2 E 3
Conveccao . H= 399
Condicao de contorno entre arestas 3 E 4
Conveccao . H= 398
Condicao de contorno entre arestas 4 E 1
Conveccao . H= 397
TEMPERATURA INICIAL - IGUAL P/ TODOS OS PONTOS 50

TIME : 100

FACE FLUXO TOTAL INSTANTANEO

1 - 2	2120.291	, CO
2 - 3	2115.264	, CO
3 - 4	2110.591	, CO
4 - 1	2105.558	, CO

VARIACAO MAXIMA : .7178023

1	8	13	18	2
45.9682	46.6137	46.8268	46.6105	45.9589

5	9	14	19	23
46.6265	47.7560	48.0867	47.7517	46.6147

6	10	15	20	24
46.8445	48.0927	48.4649	48.0880	46.8321

7	11	16	21	25
46.6297	47.7604	48.0914	47.7560	46.6179

4	12	17	22	3
45.9774	46.6255	46.8392	46.6223	45.9682

TIME : 200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	1902.459	, CO
2 - 3	1897.981	, CO
3 - 4	1894.147	, CO
4 - 1	1889.662	, CO

VARIACAO MAXIMA : .59176

1	8	13	18	2
43.1299	43.9076	44.1905	43.9016	43.1164
5	9	14	19	23
43.9211	45.0229	45.3758	45.0159	43.9058
6	10	15	20	24
44.2120	45.3845	45.7618	45.3772	44.1961
7	11	16	21	25
43.9272	45.0298	45.3831	45.0229	43.9118
4	12	17	22	3
43.1435	43.9230	44.2065	43.9170	43.1299

TIME : 1200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	662.2443	, CO
2 - 3	660.687	, CO
3 - 4	659.5273	, CO
4 - 1	657.9681	, CO

VARIACAO MAXIMA : 3.33335

1	8	13	18	2
28.0357	28.3257	28.4324	28.3220	28.0285

5	9	14	19	23
28.3310	28.7225	28.8499	28.7187	28.3235

6	10	15	20	24
28.4420	28.8542	28.9886	28.8504	28.4344

7	11	16	21	25
28.3347	28.7263	28.8537	28.7225	28.3272

4	12	17	22	3
28.0430	28.3332	28.4400	28.3295	28.0357

TIME : 2200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	230.7063	, CO
2 - 3	230.1638	, CO
3 - 4	229.7602	, CO
4 - 1	229.2169	, CO

VARIACAO MAXIMA : 1.483144

1	8	13	18	2
22.7994	22.9004	22.9376	22.8991	22.7969
5	9	14	19	23
22.9023	23.0387	23.0830	23.0373	22.8997
6	10	15	20	24
22.9409	23.0846	23.1314	23.0832	22.9383
7	11	16	21	25
22.9036	23.0400	23.0844	23.0387	22.9009
4	12	17	22	3
22.8019	22.9030	22.9402	22.9018	22.7994

TIME : 3200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	80.3718	, CO
2 - 3	80.18274	, CO
3 - 4	80.04219	, CO
4 - 1	79.85308	, CO

VARIACAO MAXIMA : .5718937

1	8	13	18	2
20.9752	21.0104	21.0234	21.0100	20.9744
5	9	14	19	23
21.0111	21.0586	21.0740	21.0581	21.0102
6	10	15	20	24
21.0245	21.0746	21.0909	21.0741	21.0236
7	11	16	21	25
21.0115	21.0591	21.0745	21.0586	21.0106
4	12	17	22	3
20.9761	21.0113	21.0243	21.0109	20.9752

TIME : 4200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	27.99969 , CO
2 - 3	27.93386 , CO
3 - 4	27.885 , CO
4 - 1	27.81904 , CO

VARIACAO MAXIMA : .2069362

1	8	13	18	2
20.3398	20.3520	20.3565	20.3519	20.3394
5	9	14	19	23
20.3522	20.3688	20.3742	20.3686	20.3519
6	10	15	20	24
20.3569	20.3744	20.3800	20.3742	20.3566
7	11	16	21	25
20.3524	20.3690	20.3743	20.3688	20.3521
4	12	17	22	3
20.3401	20.3523	20.3568	20.3522	20.3398

TIME : 5200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	9.754984	, CO
2 - 3	9.731966	, CO
3 - 4	9.71513	, CO
4 - 1	9.692044	, CO

VARIACAO MAXIMA : 7.309546E-02

1	8	13	18	2
20.1184	20.1226	20.1242	20.1226	20.1183

5	9	14	19	23
20.1227	20.1285	20.1304	20.1284	20.1226

6	10	15	20	24
20.1244	20.1304	20.1324	20.1304	20.1242

7	11	16	21	25
20.1228	20.1285	20.1304	20.1285	20.1227

4	12	17	22	3
20.1185	20.1228	20.1243	20.1227	20.1184

TIME : 6200

FACE FLUXO TOTAL INSTANTANEO

1 - 2	3.398972	, CO
2 - 3	3.390989	, CO
3 - 4	3.385223	, CO
4 - 1	3.37719	, CO

VARIACAO MAXIMA : 2.557874E-02

1	8	13	18	2
20.0412	20.0427	20.0433	20.0427	20.0412
5	9	14	19	23
20.0428	20.0448	20.0454	20.0448	20.0427
6	10	15	20	24
20.0433	20.0454	20.0461	20.0454	20.0433
7	11	16	21	25
20.0428	20.0448	20.0454	20.0448	20.0427
4	12	17	22	3
20.0413	20.0428	20.0433	20.0428	20.0412

NIVEL DE CONVERGENCIA ATINGIDO

5. Conclusão Final

Este programa, que desenvolvemos ao longo do semestre, é um programa relativamente simples, apesar de um pouco extenso.

Comporta um desenvolvimento maior em pelo menos três aspectos; a saber:

- passos variáveis em ambas as direções. Este desenvolvimento proporcionaria uma grande facilidade no estudo do problema do pé da aleta.
- condições de contorno variáveis com o tempo. O que aproximaria o modelo usado à realidade física.
- a possibilidade de trabalharmos com figuras mais elaboradas, com faces inclinadas e/ou curvas.

Ampliariamos, assim, a utilização do programa.

Devido a uma questão de tempo, ou a sua falta, não foi possível realizar estas tarefas.

Uma outra questão a ser colocada é a de um levantamento mais aprimorado das reais possibilidades e limitações do programa através de uma bateria de testes onde todos os dados de entrada pudessem ser analisados isoladamente e em diversas situações.

Por exemplo, foi constatado que para passos no tempo (DT) iguais a 10.000 s o programa diverge, a capacidade de memória do programa é outro importante fator limitante.

Mas, apesar dos fatos descritos acima, acreditamos que, devidamente utilizado, este programa pode se transformar em uma excelente e interessante ferramenta de trabalho para alunos do curso de graduação, que estejam matriculados em matérias afins com o Método de Crank-Nicholson.

6. Bibliografia

- BENNET, Carroll Osborn. Fenômenos de Transporte. Quantidade de movimento, calor e massa. São Paulo: McGraw-Hill, 1978, 812 p.
- HOLMAN, Jack Philip. Transferência de Calor. São Paulo: McGraw-Hill, 1983, 629 p.
- Notas de aula da matéria Transferência de Calor e Massa (PMC442) do curso de Engenharia Mecânica da Escola Politécnica da U.S.P.



