PUBLICAÇÃO 1/65

DO

CENTRO DO CÁLCULO NUMÉRICO DA U. S. P.

Instrutores: DIRCEU D. SALVETTI

VALDEMAR W. SETZER

DIAGRAMA DE BLOCOS

NOTAS DE AULA - 1965

PUBLICAÇÃO 1/65 DO CENTRO

DO CÁLCULO NUMÉRICO DA USP

Instrutores: Dirceu D. Salvetti

Valdemar W. Setzer

Notas de Aula - 1965

DIAGRAMAS DE BLOCOS

A presente apostila foi redigida com a finalidade de auxiliar em classe o desenvolvimento do curso de Cálculo Numérico na parte de programação. É de se ressaltar que a mesma não dispensa as explicações a serem dadas pelos professôres. Devemos observar ainda que os diagramas apresentados nem sempre representam a solução mais elegante, tendo em vista a finalidade didática a que se destinam.

Recordamos que os diagramas de blocos, também referidos como cartas de fluxo, não passam de um algoritmo, um verdadeiro código de computador escrito numa linguagem universal, independente da linguagem particular de qualquer computador.

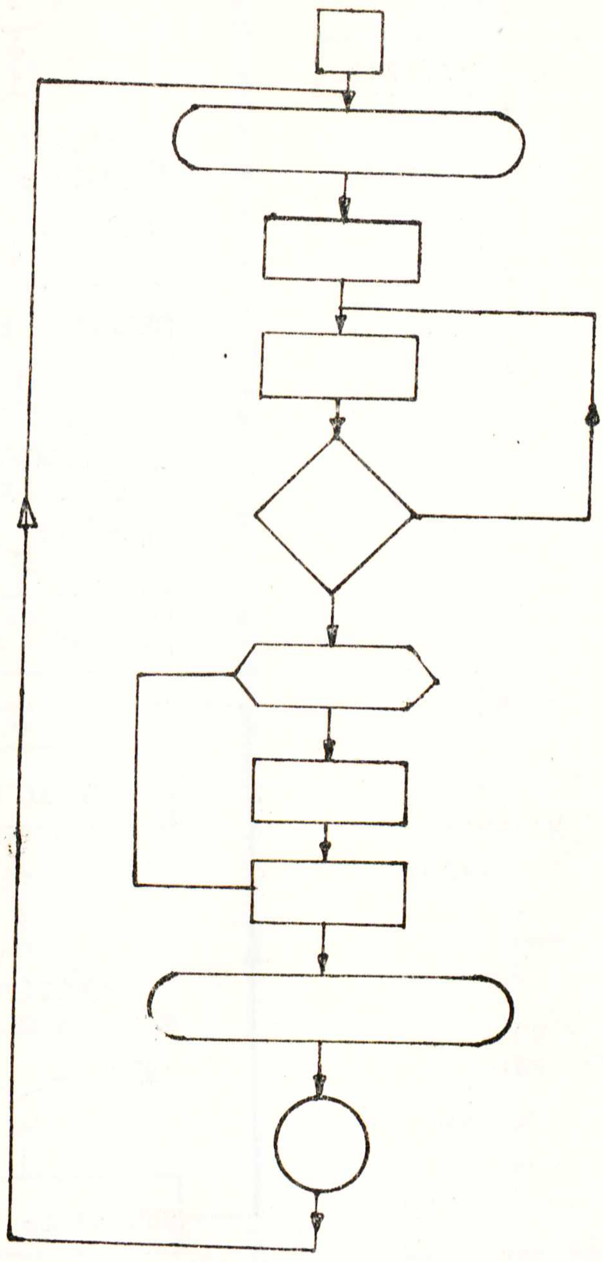
Os diagramas de blocos podem se apresentar de uma forma resumida ou detalhada. Apenas neste último caso, entretanto, se prestam como auxiliar direto na codificação de um programa.

A maior parte dos diagramas foram apresentados nesta forma e, além disso, com vistas à programação em FORTRAN-1, que é a linguagem simbólica válida para o caso particular do IBM-1620. Desejamos com esta observação sublinhar que as expressões dos diversos blocos de cada diagrama foram redigidos de acôrdo com os códigos da programação em FORTRAN, respeitando inclusive a distinção existente entre variável de ponto fixo e de ponto flutuante.

Tôda crítica tendente a melhorar o aspecto didático desta publicação será bem recebida pelos autores que, antecipadamente, agradece.

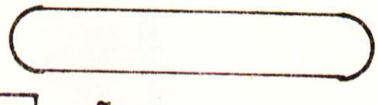
Desejamos também agradecer à colaboração que nos foi prestada pelos colegas Alvaro Puga Paz e Arthur Schultz Azevedo.

ATENÇÃO: Na confecção dos diagramas de blocos utilizar as setas que indicam a direção do fluxo do processamento conforme se encontra ilustrado abaixo.



Início é <u>representado</u> por um quadrado 

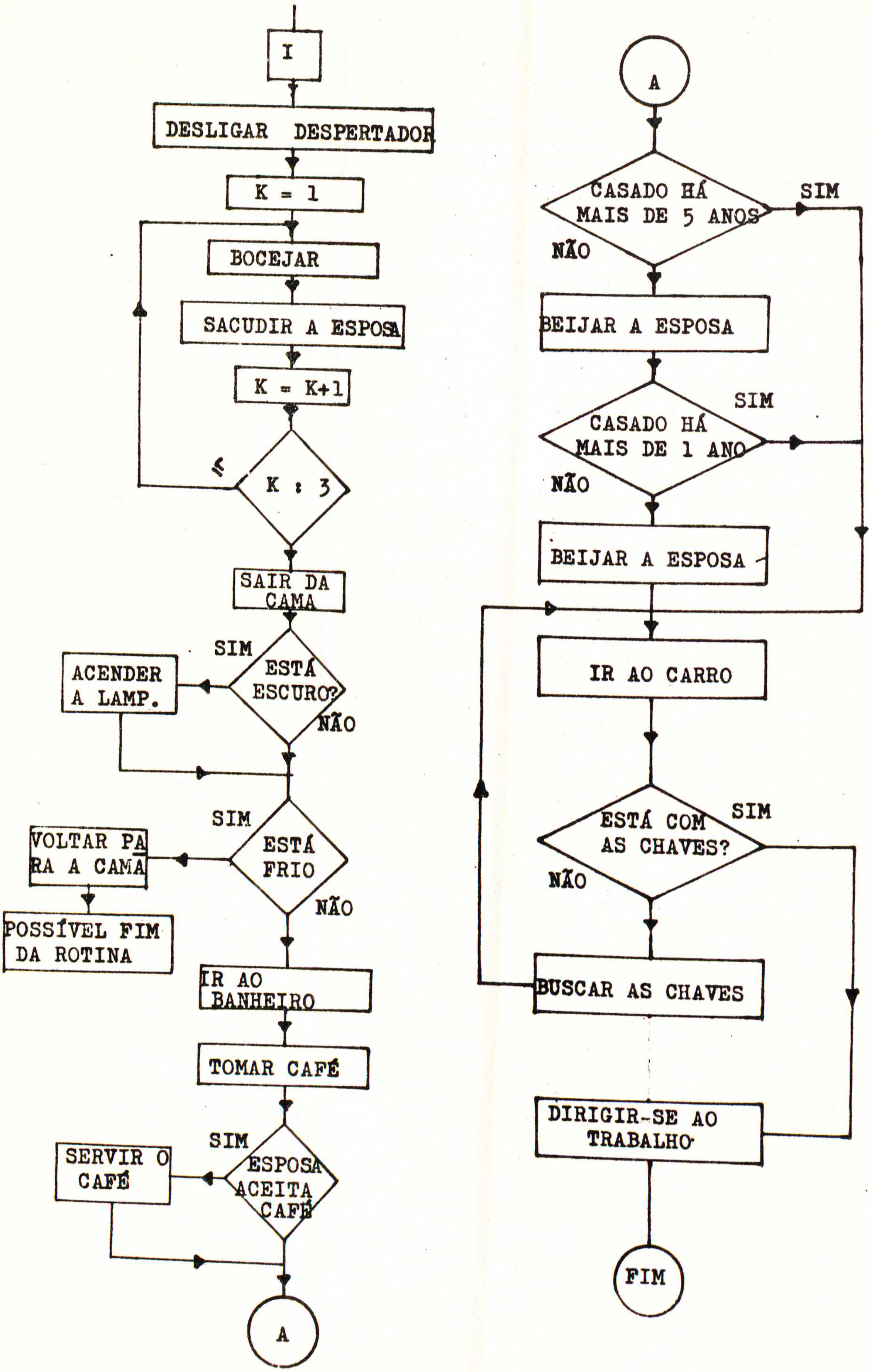
As instruções <u>LEIA</u> e <u>IMPRIMA</u>, de entrada e saída, por um bloco SEMPRE abaulado nas extremidades, isto é,



Pare ou Pausa são representados por um círculo 

EXEMPLO CLÁSSICO

Como proceder para levantar de manhã e ir ao trabalho.



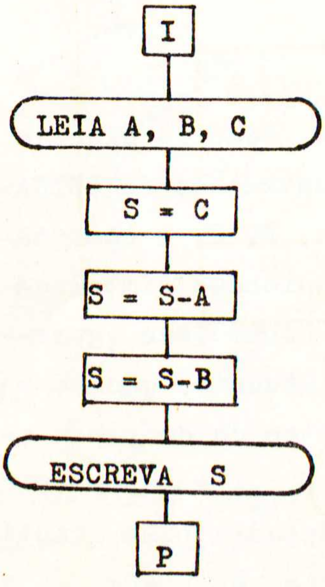
EXERCÍCIO 1

Somar duas quantidades A e B e subtrair o resultado de outra C.

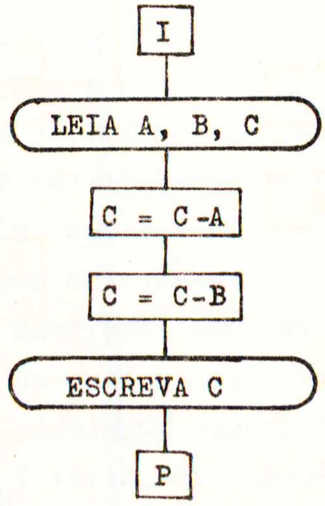
Os diagramas 1 e 2 serão adotados para uso em programação em linguagem de máquina. Está prevista para cada operação aritmética a ser executada uma instrução correspondente. A letra S representa o lugar da memória onde ficará armazenada a resposta do problema. LEIA é uma instrução de ENTRADA, isto é, as quantidades A, B, C perfuradas, por exemplo, num cartão, serão introduzidas na memória do computador. A instrução de SAÍDA, ESCREVA, permite que a quantidade S armazenada na memória seja fornecida ao operador, através, por exemplo, da máquina de escrever.

No segundo diagrama de blocos a solução do problema está representada pela letra C que foi sucessivamente alterada de acôrdo com o problema proposto, de modo a corresponder à solução. Devemos desde já nos habituar com a representação do tipo C = C - A que não é válida no sentido algébrico. Significa simplesmente que a quantidade representada pela letra C à esquerda da igualdade passa a valer a quantidade representada por C-A à direita dessa igualdade.

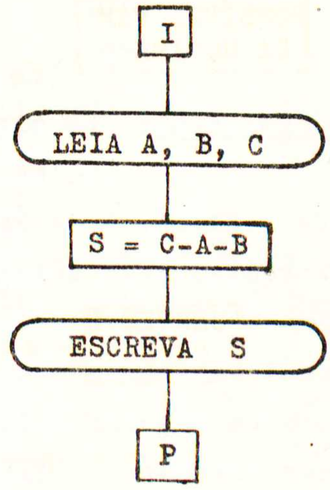
O terceiro diagrama de blocos é para uso em linguagem simbólica. Observamos que várias operações aritméticas podem ser comandadas por uma única instrução (S = C - A - B). O diagrama para uso em linguagem simbólica deve ser elaborado de tal modo que de sua interpretação (feita por mera tradução de cada operação indicada pela correspondente instrução codificada) resulte imediatamente a programação simbólica.



1



2



3

Programa em básico (Diagrama 1)

Mapa da Memória

A B C S

$\bar{X}\,X\,X\,X\,\bar{X}\,X\,X\,X\,\bar{X}\,X\,X\,X\,X\,X\,\bar{0}\,0\,0\,0\,0\,0$

0	0	0	0	0
2	2	2	2	2
0	0	0	0	0
0	0	0	1	1
0	3	7	3	9

10000	36	02000	00100	Leia A, B, C, S pela maq. escrever
10012	26	02019	02012	S = C
10024	22	02019	02002	S = S - A
10036	22	02019	02006	S = S - B
10048	15	02020	0000≠	Coloca marca de registro na posição 02020
10060	38	02013	00100	Escreva S pela maq. escrever
10072	48			Pare

Observação: Quando a saída é feita pela máquina de escrever é necessária a presença da marca de registro ≠. Em função da ins-trução 38 02013 00100 a máquina de escrever registra a informação armazenada na memória, a partir da posição 02013, processando da esquerda para a direita, prosseguindo até encontrar uma marca de registro ≠.

Programa em FORTRAN (Diagrama 3)

```
      READ 10, A, B, C
      S = C - A - B
      PRINT 20, S
   10 FORMAT (   )
   20 FORMAT (   )
      STOP
      END
```

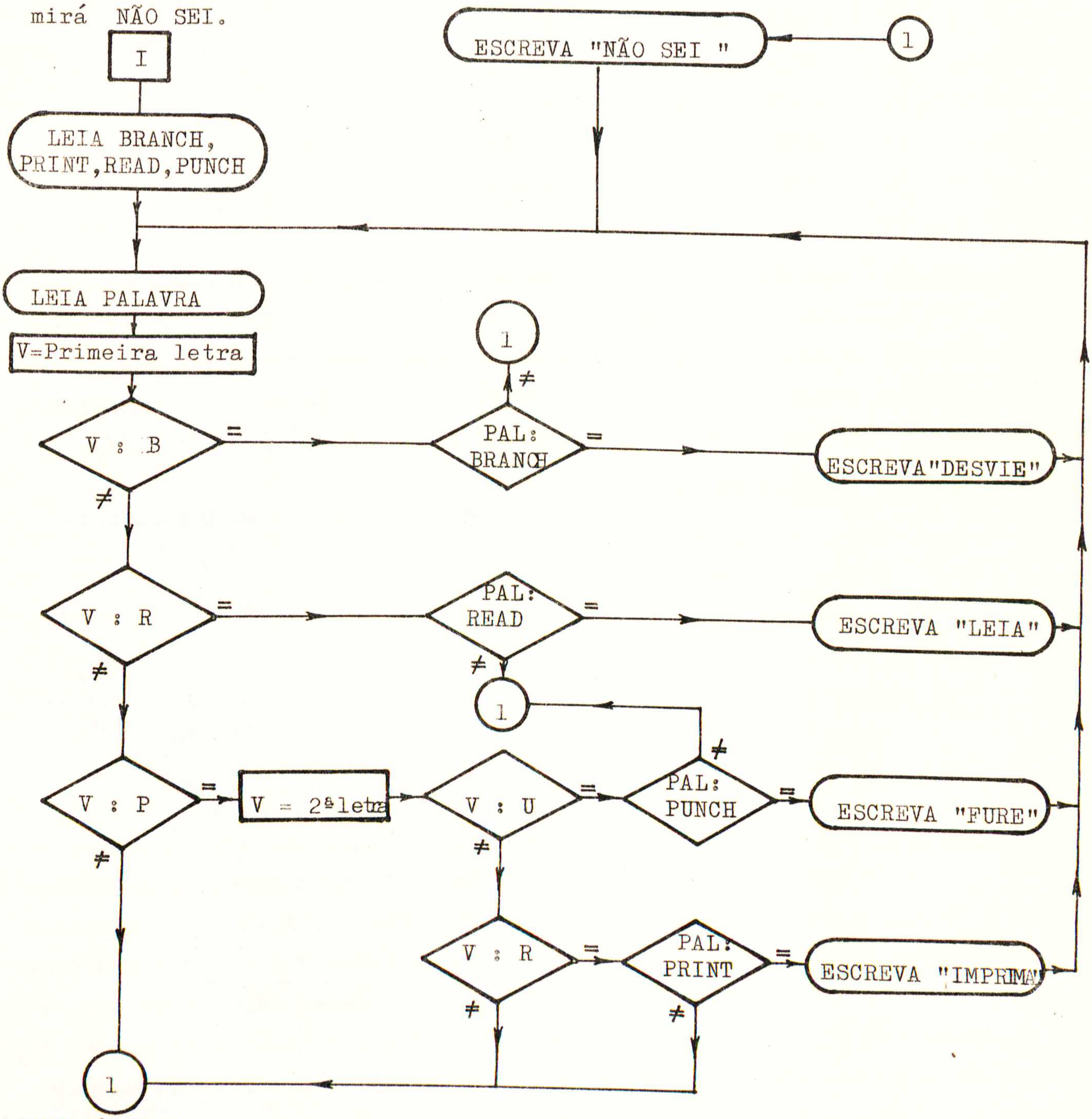
Observação: Em alguns tipos de programação simbólica, como no exemplo acima, encontra-se relacionada com as instruções de ENTRADA e SAÍDA, a de FORMATO. Quando dizemos READ 10, A, B, C, o computador dispõe-se a introduzir nas regiões da memória, indicadas simbòlicamente por A, B e C, as informações numéricas a serem fornecidas, guar-dando-as segundo a modalidade (ponto fixo ou ponto flutuante) que se encontra especificada pela instrução FORMAT (   ). Dentro do do parênteses colocamos o tipo de formado escolhido: I_w, $F_{w.d}$, $E_{w.d}$; a letra w indica sempre o número de casas ou colunas a serem ocupadas pela quantidade a ser armazenada. Nos formatos de ponto flutuante, $E_{w.d}$ e $F_{w.d}$, a letra d indica o número de casas decimais depois da vírgula.

EXERCÍCIO 2

Programar a tradução para o português das palavras inglesas READ (LEIA), PUNCH (FURE), PRINT (IMPRIMA) e BRANCH (DESVIE).

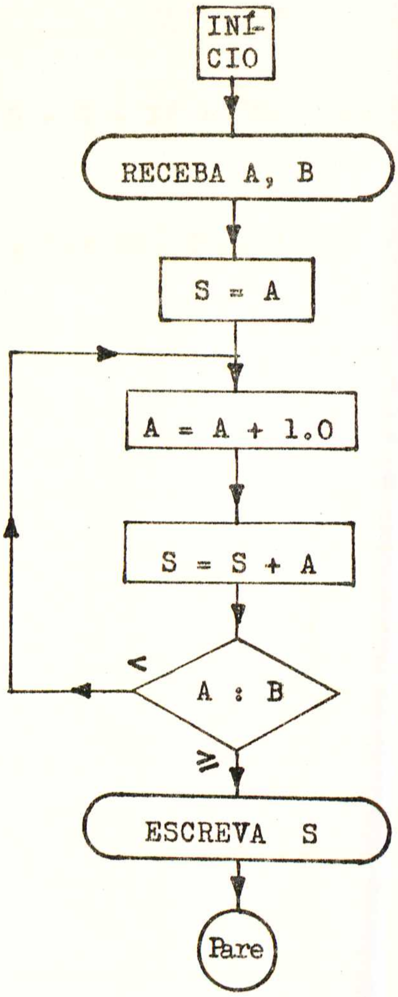
Rotina

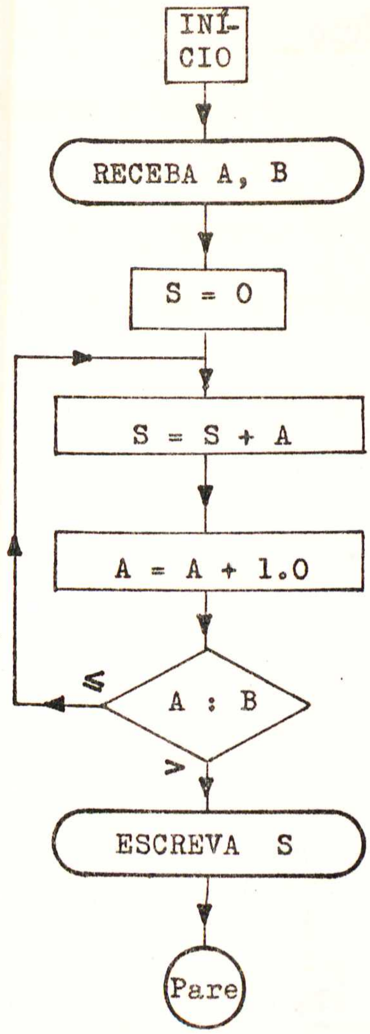
Inicialmente introduzimos na memória as palavras inglesas que devem ser traduzidas. O diagrama de blocos prevê o seguinte comportamento ao ser lida uma palavra inglesa qualquer: imprimirá a tradução em português se fôr uma das palavras que consta do "dicionário" ou imprimirá NÃO SEI.



EXERCÍCIO 3

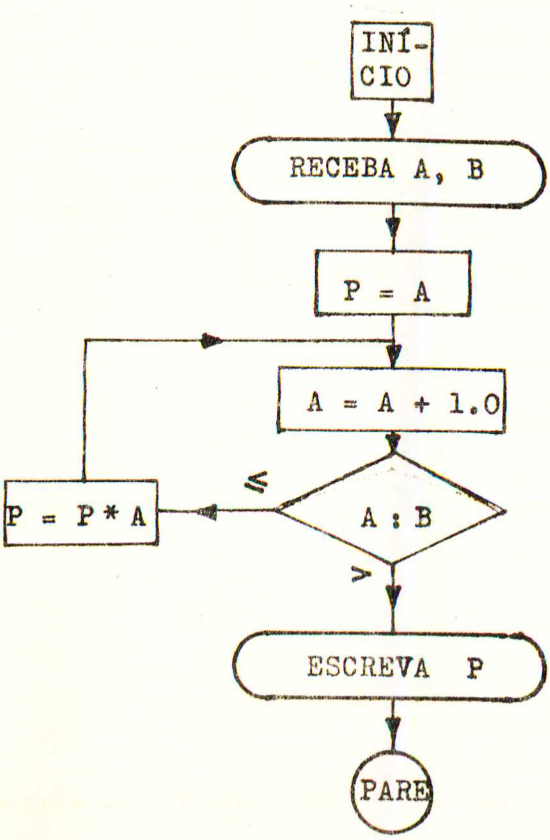
Dados os números inteiros A e B, calcular a soma dos consecutivos de A a B.

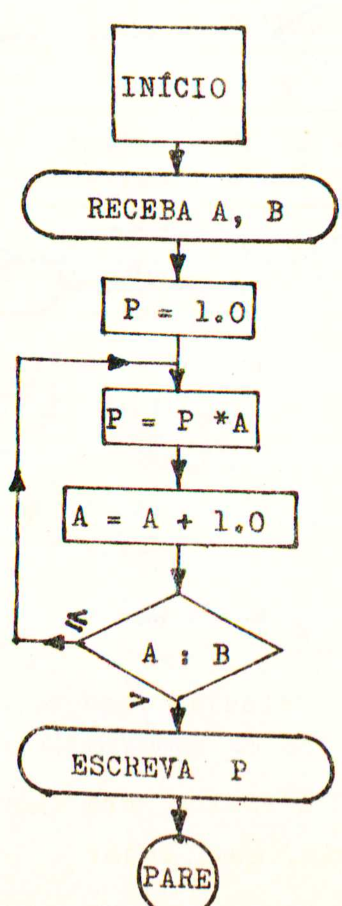




EXERCÍCIO 4

Dados os inteiros A e B, calcular o produto dos consecutivos de A a B.





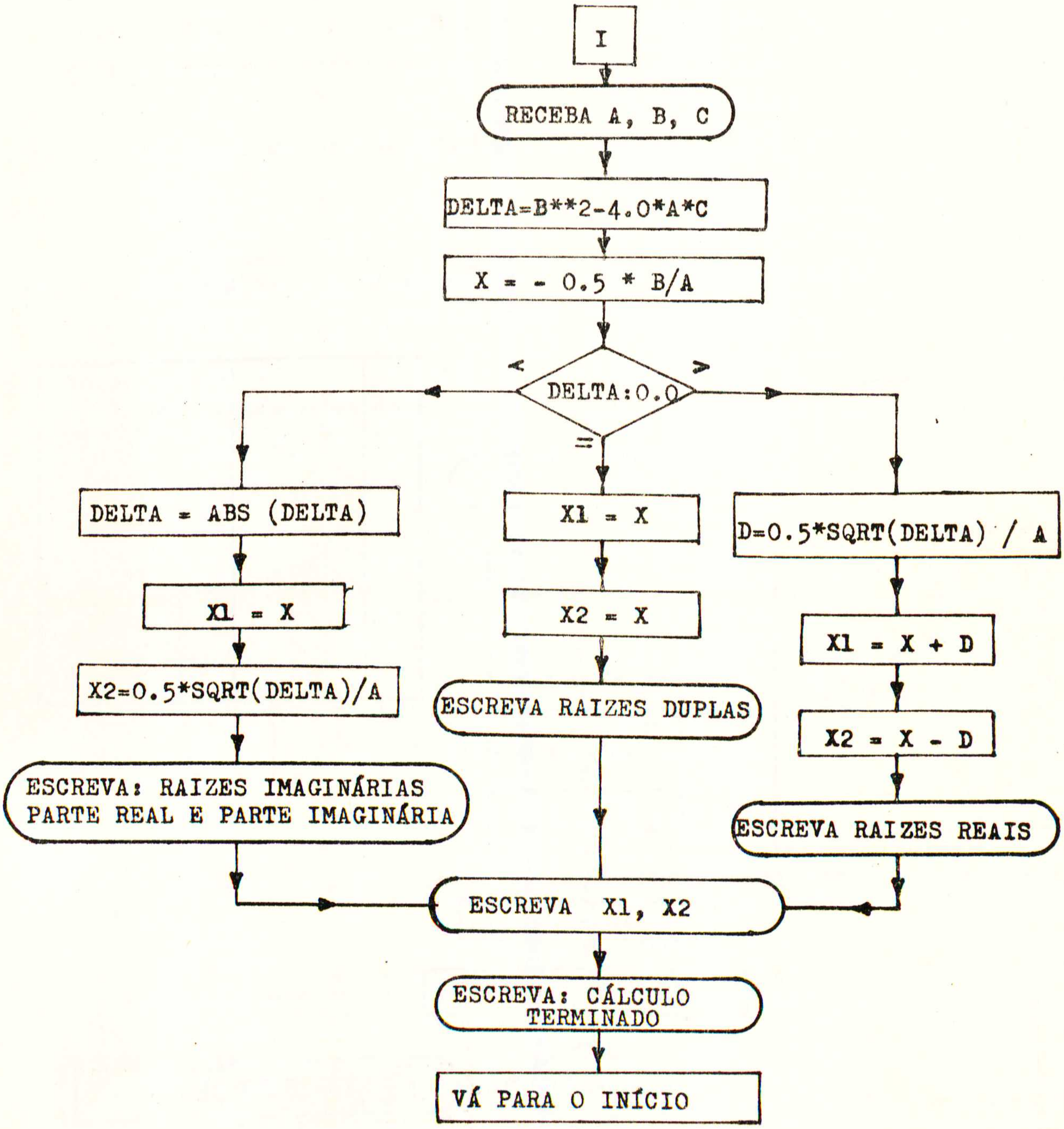
EXERCÍCIO 5

Resolver a equação do segundo grau $AX^2 + BX + C = 0$

Rotina

$DELTA = B^2 - 4.AC$; $D = \sqrt{DELTA}/(2 \cdot A) = 0.5 * SQRT(DELTA)/A$

$X = - B/2 \cdot A$; $X1 = X + D$; $X2 = X - D$



EXERCÍCIO 6

Calcular sen x com aproximação dada (ÊRRO).

Rotina

A rotina será baseada no desenvolvimento de sen x em série de McLaurin, qual seja:

$$\text{sen } x = \sum_{n=0}^{\infty} (-1)^n \, x^{2n+1} \,/\, (2n+1)!$$

Como essa série convergente é alternada, o êrro que se comete, quando se toma como valor aproximado da série uma reduzida qualquer, é menor que o valor absoluto do primeiro têrmo diferente de zero desprezado. Portanto, para se calcular sen x, com uma certa aproximação,basta somarmos os têrmos do desenvolvimento de McLaurin, até aquele cujo valor absoluto seja menor ou igual à aproximação pedida, a qual indicamos por ERRO.

Notemos que qualquer têrmo da série é obtido do têrmo precedente pela relação:

$$\frac{X^{2N+1}}{(2N+1)!} = \frac{X^{2N-1}}{(2N-1)!} \cdot \frac{X^2}{(2N)(2N+1)} \text{ ; } X = X\ X2/2N(2N+1) = X.X2/AN(AN+1), X2=X*X$$

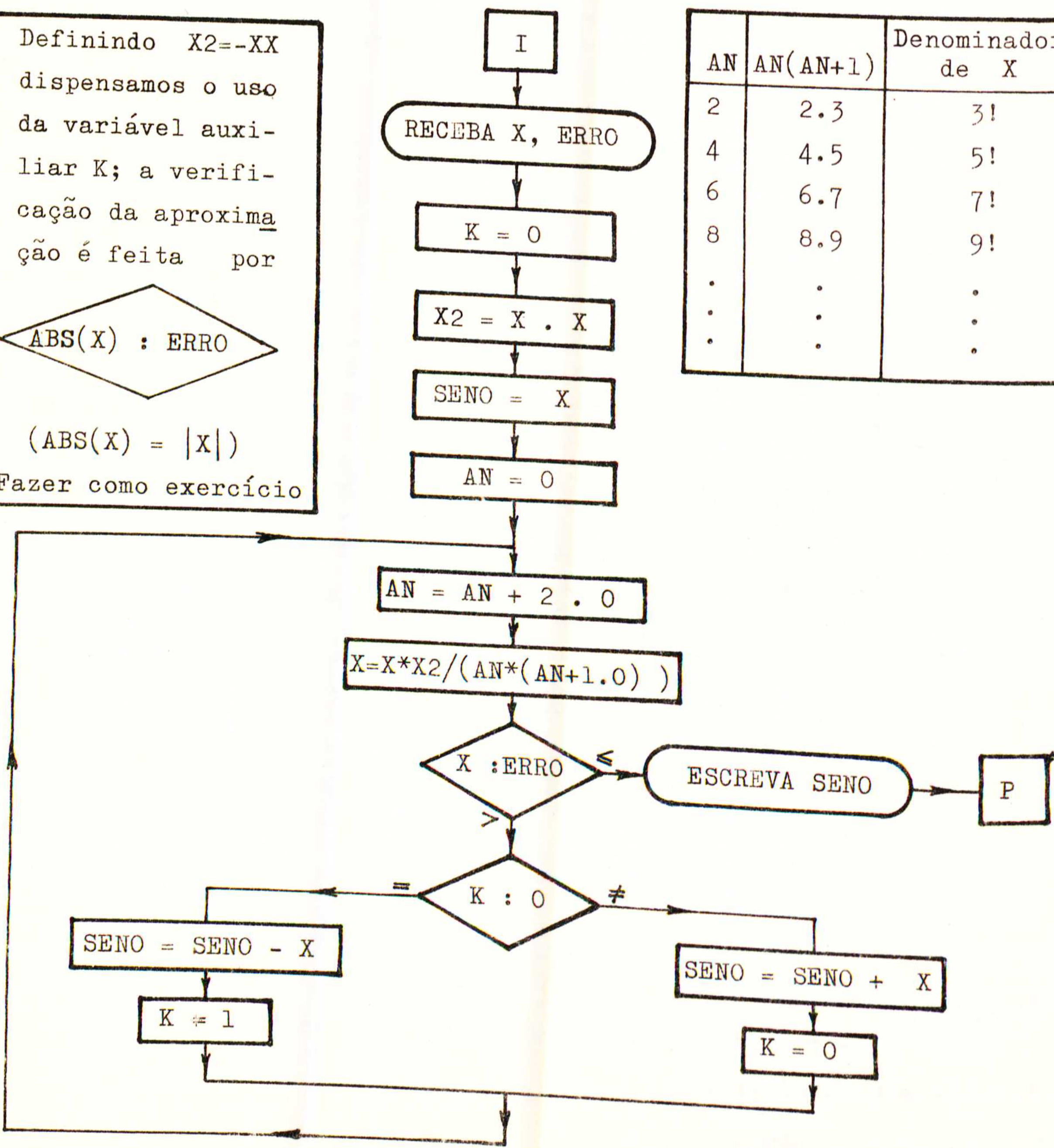
Definindo X2=-XX dispensamos o uso da variável auxiliar K; a verificação da aproximação é feita por

ABS(X) : ERRO

(ABS(X) = |X|)

Fazer como exercício

AN	AN(AN+1)	Denominador de X
2	2.3	3!
4	4.5	5!
6	6.7	7!
8	8.9	9!
.	.	.
.	.	.
.	.	.

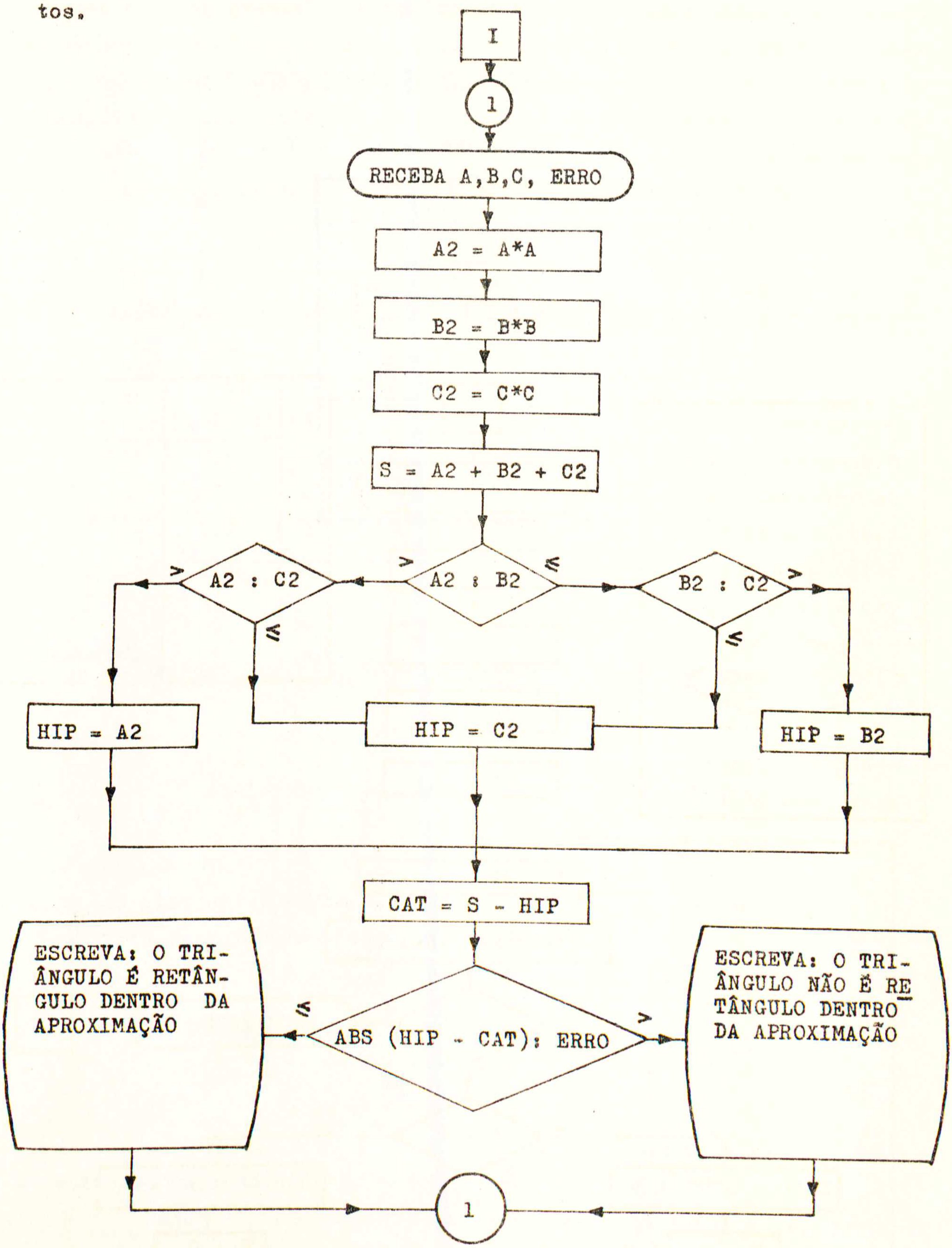


EXERCÍCIO 7

Sabendo-se que A, B e C são os lados de um triângulo, verificar se dentro de uma certa aproximação (ÊRRO) êle é retângulo.

Rotina

Se |HIP - CAT| ≤ ERRO, o triângulo é retângulo, sendo HIP o valor da hipotenusa ao quadrado e CAT a soma dos quadrados dos catetos.



Exercícios propostos

1. Tabelar arctg x dentro de uma certa aproximação (ÊRRO) com x variando de XMIN até XMAX com passo H dado. Adotar como rotina o desenvolvimento

$$\text{arctg } x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots \qquad (\text{convergente para } -1 \leqslant x \leqslant 1)$$

2. Sendo $S = 1 + \frac{1}{2.3} + \frac{1.3}{2.4.5} + \frac{1.3.5}{2.4.6.7} + \frac{1.3.5.7}{2.4.6.8.9} + \dots$

imprimir I, T (I), S (I) para cada valor de I = 1,2,...,20

I	T(I)	S (I)
1	1	1
2	$\frac{1}{2.3}$	$1 + \frac{1}{2.3}$
3	$\frac{1.3}{2.4.5}$	$1 + \frac{1}{2.3} + \frac{1.3}{2.4.5}$
⋮		
20		

3. Calcular N!

EXERCÍCIO 8

Dado N inteiro positivo, calcular $\sqrt{N}$ por subtrações de ímpares consecutivos.

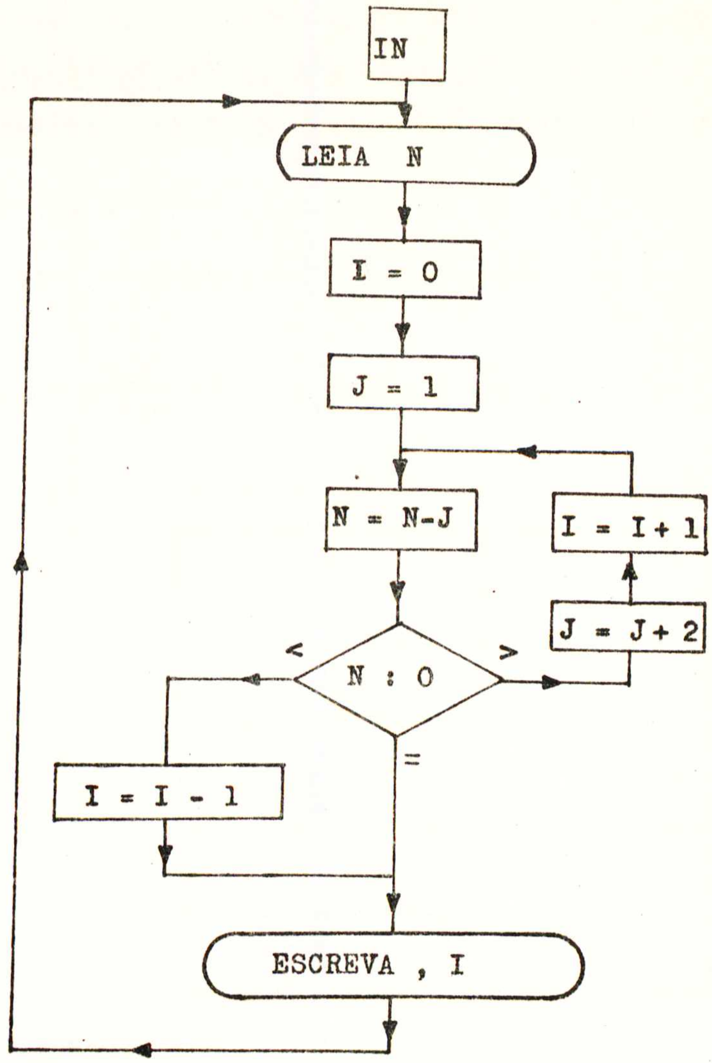
Rotina

A rotina de cálculo está baseada na propriedade:

$$N = M^2 = 1 + 3 + 5 + \dots + (2M - 1)$$

O diagrama de blocos abaixo prevê a extração da raíz de N de modo exato se N fôr quadrado perfeito. Caso contrário calcula $\sqrt{N}$ com aproximação por falta. (Por exemplo, $\sqrt{10} = 3$).

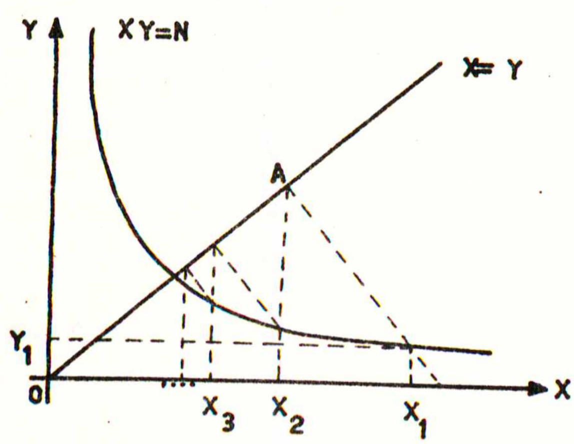
(V. página seguinte)



EXERCÍCIO 9

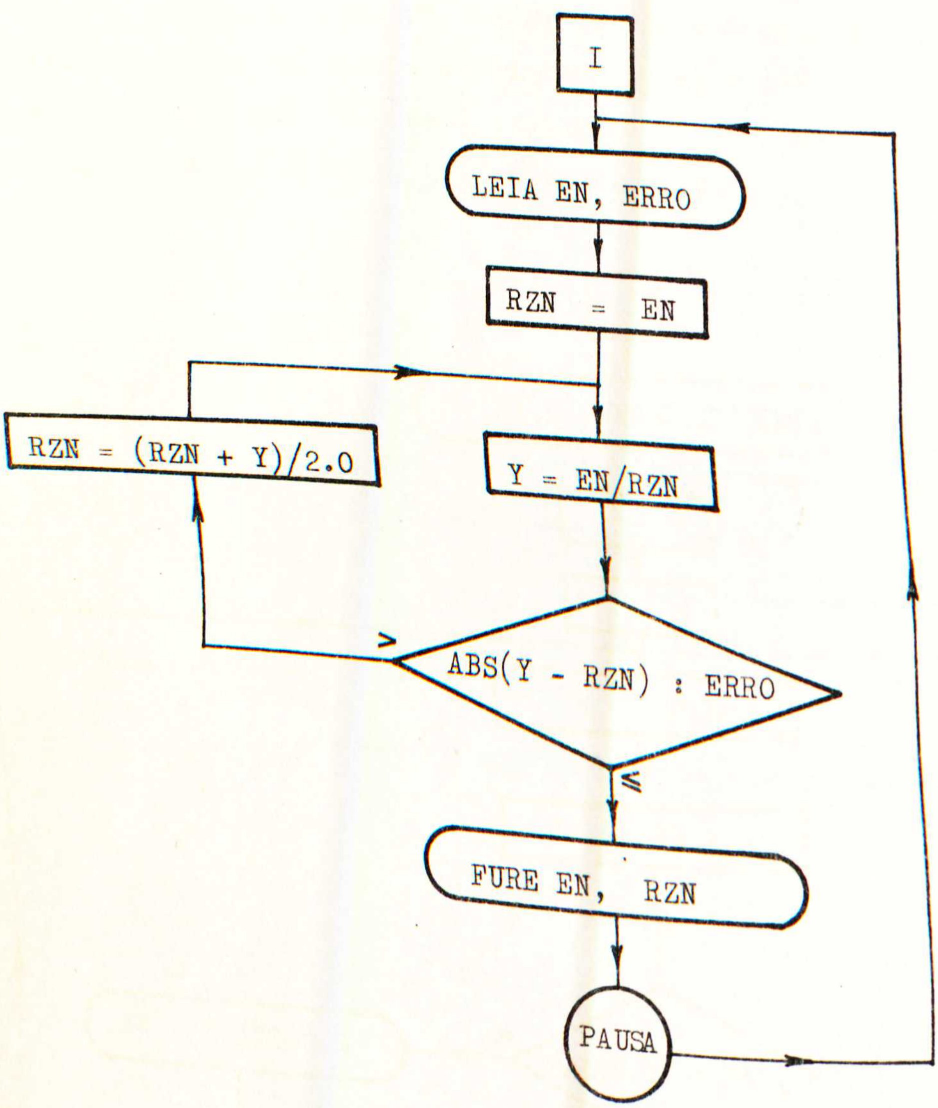
Calcular raiz de N pelo método de Newton.

Rotina



A raiz de N está dada pela abscissa ou ordenada do ponto de intersecção da reta X = Y com a parábola XY = N.

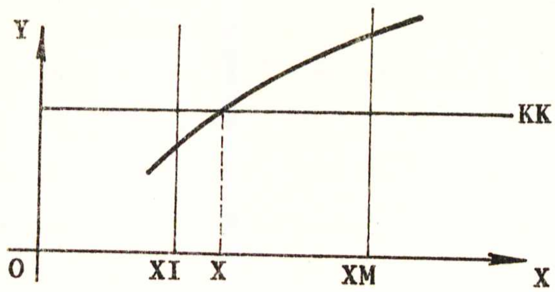
Seja X1 uma aproximação arbitrária da abscissa do ponto de intersecção. Traçando-se por X1 uma paralela ao eixo OY determinamos o ponto pertencente à parábola, cuja ordenada Y1 satisfaz a relação Y1 = N/X1. Dêsse ponto baixamos uma perpendicular à reta, interceptando-a no ponto. A. Verifica-se fàcilmente que a abscissa X2 de A é média aritmética entre X1 e Y1. A abscissa X2 resulta uma melhor aproximação à raiz de N do que X1. Dispensando a X2 o mesmo tratamento, determinamos X3. A sequência $\{X_n\}$ assim construída é convergente para $\sqrt{N}$. É possível determinar-se $\sqrt{N}$ com uma aproximação pré-fixada a qual indicaremos no diagrama de blocos por ÊRRO. É claro que no caso de programação simbólica a aproximação está sujeita às limitações do próprio compilador.



EXERCÍCIO 10

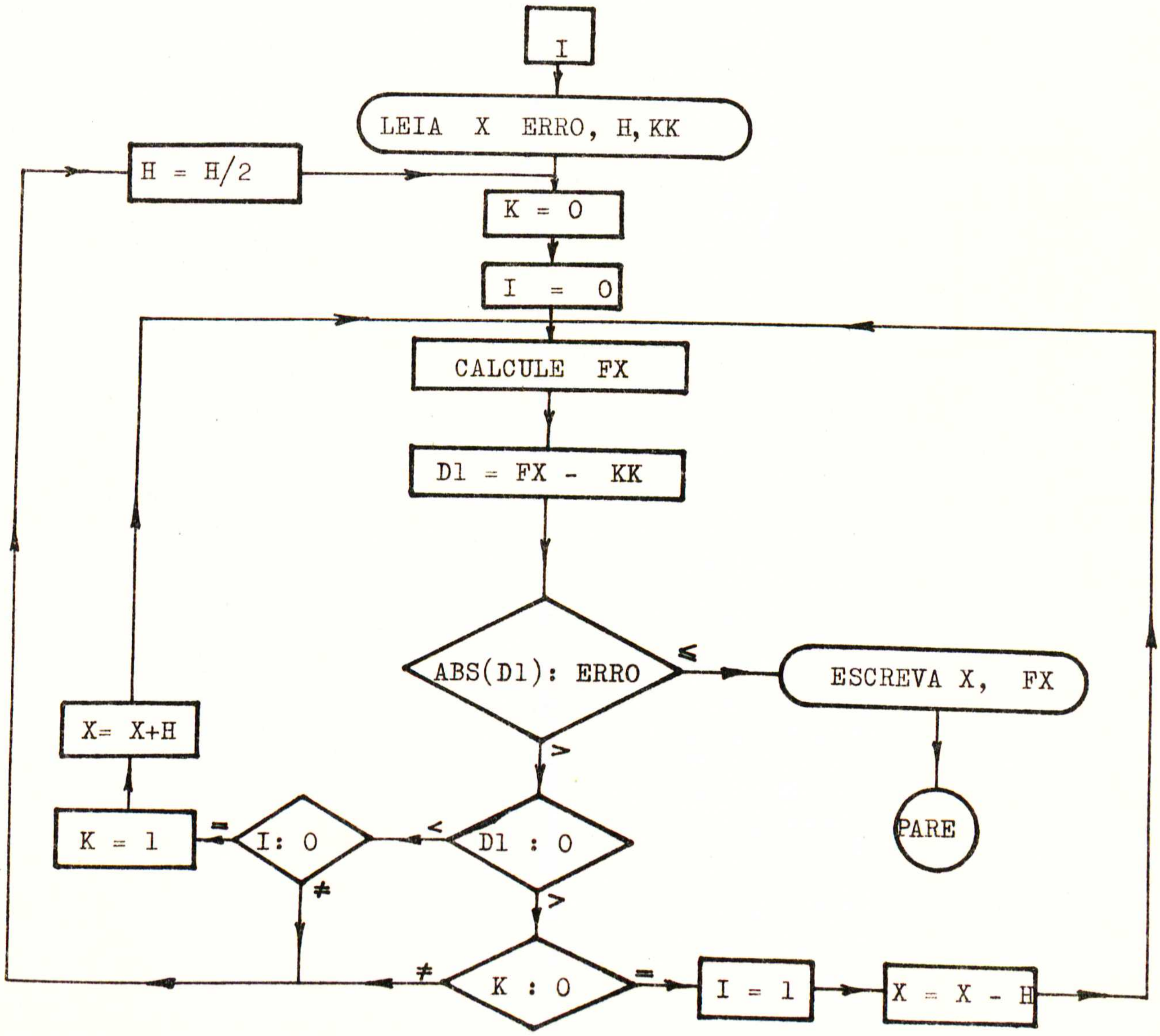
Sabendo-se que a função contínua F(x) é monótona crescente no intervalo XI⊢⊣XM, determinar, com uma aproximação dada (ÊRRO), o valor de X tal que F(X) = K.K

Rotina



Devemos determinar X tal que $|F(X) - KK| \leq$ ÊRRO

Será fornecido ao computador um valor X pertencente ao intervalo XI ⊢⊣ XM, bem como o valor do passo H. Calculando-se o valor de $F(\bar{X})$, se $|F(\bar{X}) - KK| \leq$ ERRO, tentaremos com outro valor de X: $\bar{X}$+H ou $\bar{X}$ - H. Se os valores de X segundo o passo H não permitirem solucionar o problema, repetiremos o processo com passo H/2 e assim por diante.



EXERCÍCIO 11

Calcular a soma de 100 números.

Programação básica

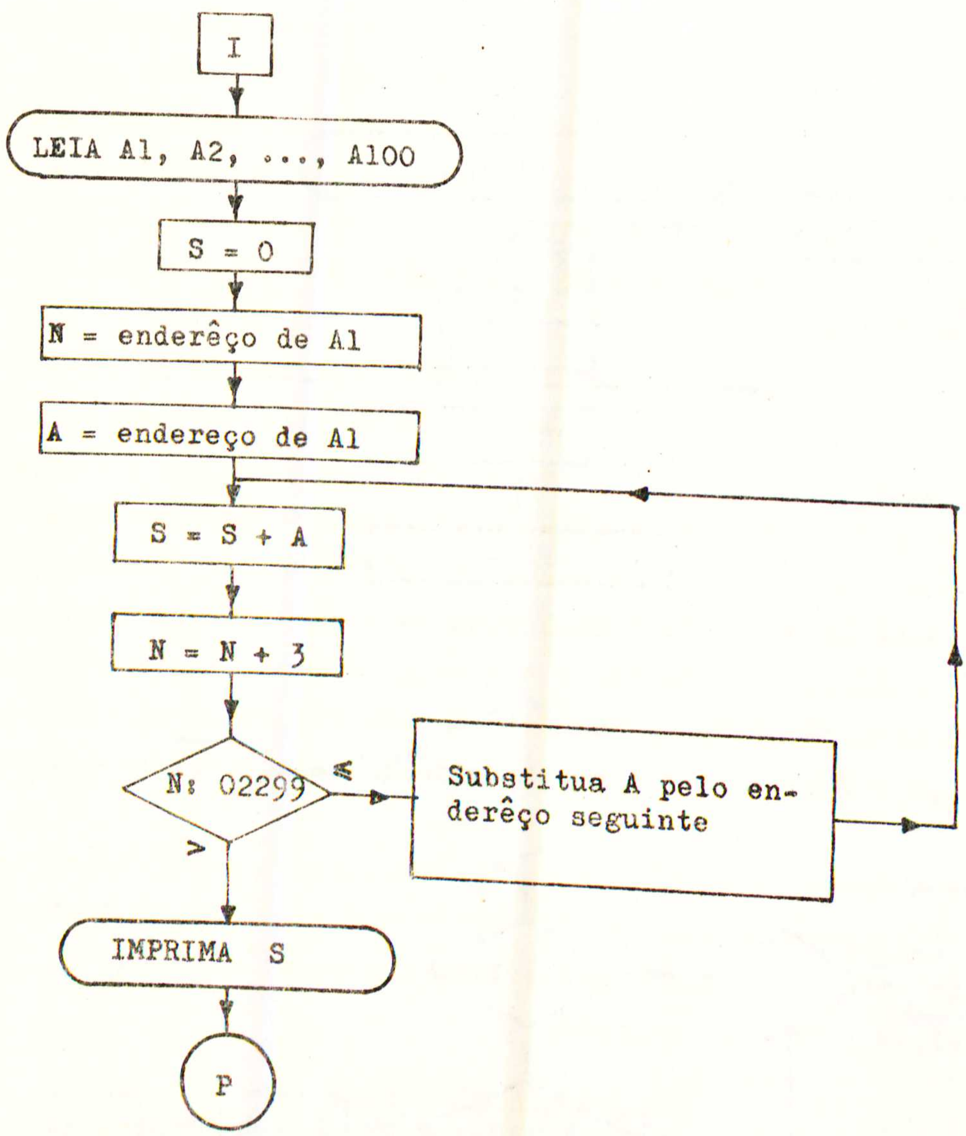
Mapa da memória

O programa que faremos permitirá o cálculo da soma de quaisquer 100 números, desde que seja respeitado o dimensionamento adotado. Vamos supor que cada número a ser somado não utilize em sua representação mais do que 3 dígitos decimais. Damos em seguida a área da memória onde os dados serão armazenados.

```
0   0     0         0        1       1
2   2     1         2        4       5
0   0     0         2        9       0
0   0     0         9        9       0
0   2     5         9        6       0
X X X X X X ... X X X        X X X X X

A1    A2    ... A100             5
```

Diagrama de Blocos



Sendo N o endereço de um certo dado, ao fazermos N = N + 3, estaremos considerando o endereço do dado imediatamente seguinte, pois, por hipótese, cada número armazenado ocupa sempre 3 posições da memória.

O maior número sob nosso dimensionamento é representado por $\bar{9}99$ e o zero por $\bar{0}00$.

Programa básico

10000	36	02000	00500	Leia os dados por cartão
10012	16	15000	$\bar{0}0000$	S = 0
10024	21	15000	$0\bar{2}002$	S = S + A
10036	11	10035	$000\bar{0}3$	N = N + 3
10048	14	10035	$0\bar{2}299$	Compara N com endereço de A100: 02299
10060	47	10024	01100	Se N $\leqslant$02299 desvie para a posição 10024
10072	38	14996	00400	Perfura S
10084	48			

Observações

1 - Quando dizemos, por exemplo, instrução colocada na posição 10012 do programa, desejamos dizer que a instrução se encontra armazenada na área que vai de 10012 a 10023.

2 - Um dos princípios de programação básica é que não trabalhamos diretamente com os dados, porém, com os endereços dêsses dados. Entretanto, as chamadas instruções imediatas (colocadas nas posições 10012, 10036 e 10048 de nosso programa) nos proporciona a possibilidade de trabalhar diretamente com os dados, desde que, 1º) ocupe o lugar reservado à parte Q da instrução; 2º) seja escrito com no máximo 5 dígitos.

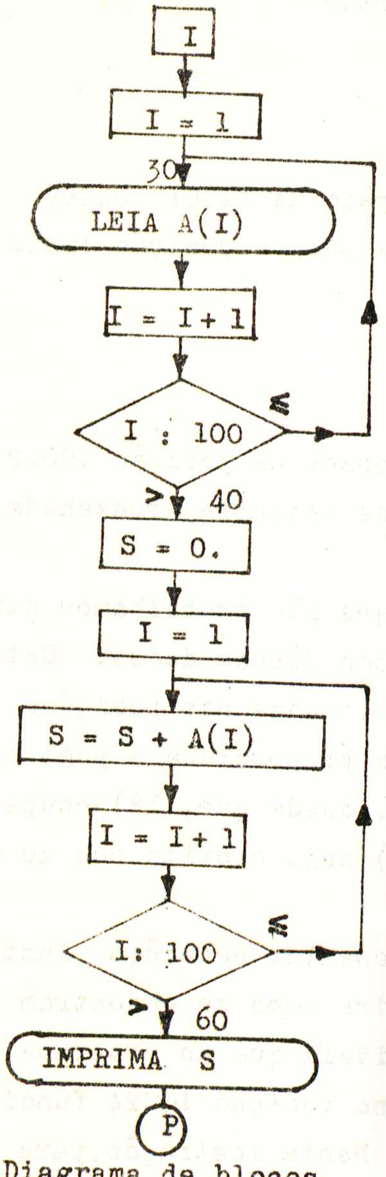
3 - Os números colocados na memória podem representar endereço, instrução ou dado, dependendo ùnicamente da maneira como se encontram relacionados no programa. É nesta ordem de idéia que no programa acima o endereço 02002 da instrução colocada na posição 10024 funciona como dado da instrução colocada em 10036. Nesta instrução,para nos referirmos ao dado 02002 necessitamos citar seu endereço que, òbviamente, é 10035. Portanto, o endereço 02002 da instrução 10024 é dado da instrução 10036. Por êste motivo colocamos em 02002 a marca de campo ($0\bar{2}002$), a qual não interfere com a interpretação da instrução 10024 por parte do computador.

4 - Graças a êste recurso conseguimos traduzir as duas etapas assinaladas no diagrama de blocos por a) "N = N + 3" e b) "Substitua A pelo endereço seguinte", pela única instrução imediata 11, colocada na posição 10036 do programa.

Programa em Fortran-1

Na programação simbólica introduzimos o conceito de variável indixada. Para cada valor do índice o computador considera o valor da variável que lhe corresponde. Por exemplo, representamos cada um dos 100 números a serem somados por A (I), I = 1, 2, ..., 100. Para cada valor de I, referímo-nos ao valor correspondente da variável indixada A (I).

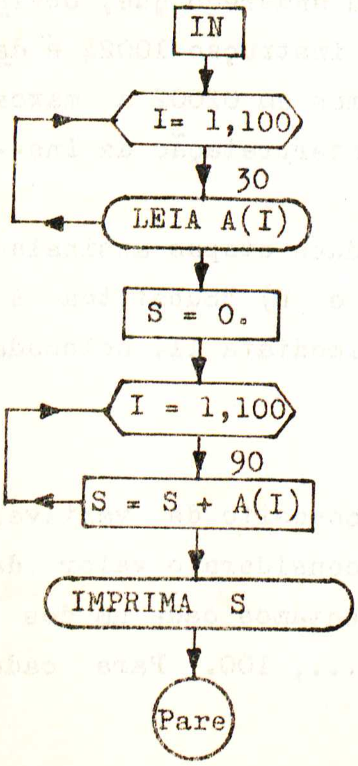
Diagrama de blocos



Programa em Fortran-1

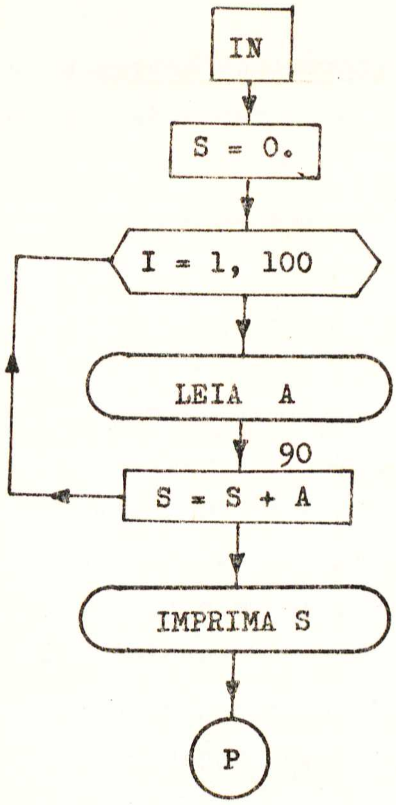
```
      DIMENSION A(100)
      I = 1
   30 READ 10, A(I)
      I = I + 1
      IF (I-100) 30,30,40
   40 S = 0.
      I = 1
   50 S = S + A(I)
      I = I + 1
      IF (I-100) 50,50,60
   60 PUNCH 70, S
   10 FORMAT (F4.0)
   70 FORMAT (E14.7)
      STOP
      END
```

Diagrama de blocos



Programa em Fortran-1

```
      DIMENSION A(100)
      DO 30 I = 1, 100
   30 READ 10, A(I)
      S = 0.
      DO 90 I = 1, 100
   90 S = S + A(I)
      PRINT 70, S
   10 FORMAT (F4.0)
   70 FORMAT (E14.7)
      STOP
      END
```

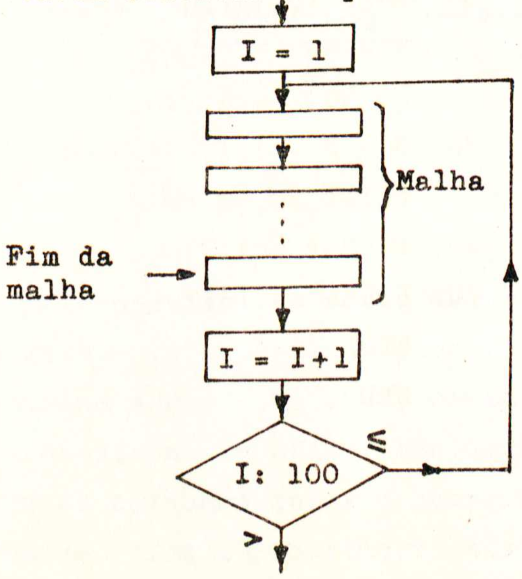



```
      S = 0.
      DO 90 I = 1, 100
      REAL 10, A
90    S = S + A
      PRINT 70, S
10    FORMAT (F.4.0)
70    FORMAT (E14.7)
      STOP
      END
```

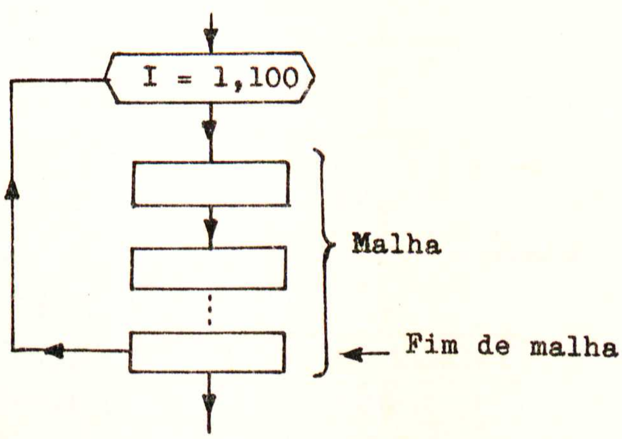
Observações

1- A instrução de especificação DIMENSION encontra-se relacionada com o uso da variável indixada. Em DIMENSION A(100) informamos que o <u>nú-mero máximo</u> de valores indixados da variável A (I) é 100. O computador, por efeito dessa instrução, reserva na área da memória as posições necessárias para poder colocar os valores numéricos de cada A (I), I = 1, 2, ..., 100. Esta instrução deve ser dada sempre antes que a variável indixada seja utilizada no programa.

2- Os diagramas de blocos foram formulados para uso em fortran. A última representação é mais econômica e deve ser usada preferentemente. Convém observar a equivalência das representações

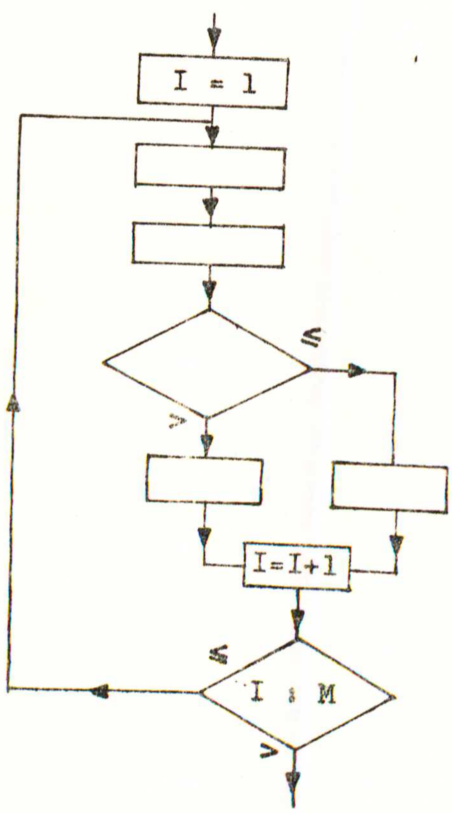


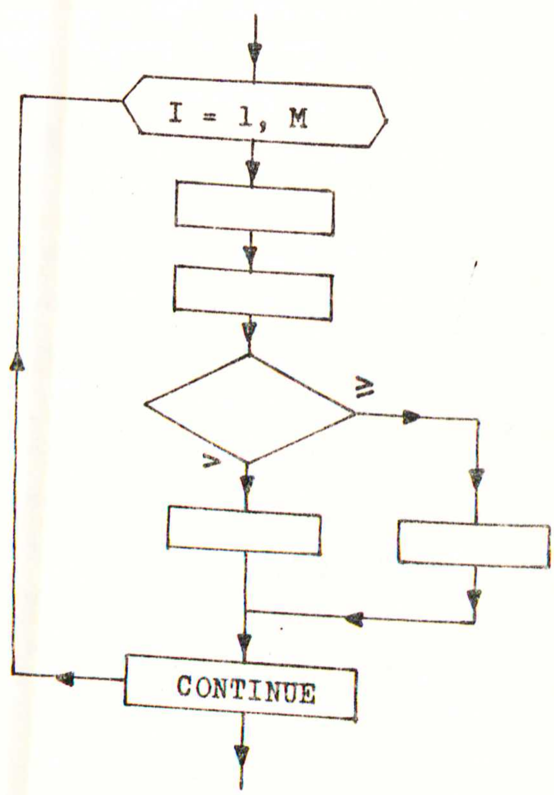
<u>Diagrama para emprêgo do comando DO</u>



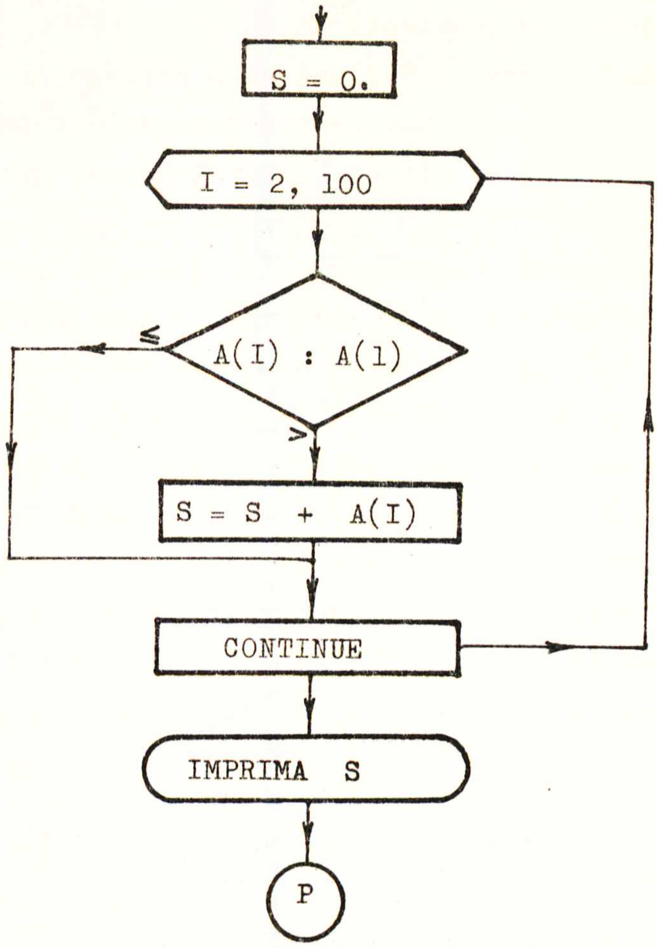
Traduzidos para linguagem simbólica evidencia-se que a segunda utili za um número menor de comandos.

3- Os diagramas da observação 2 ilustram o emprêgo da contagem; em cada ciclo uma instrução (ou grupo de instruções) é repetida para novos va lores redefinidos das variáveis. A contagem em geral se processa a-té o último valor da variável conta ciclo o que se encontra ilustra-do com o exemplo dêste exercício. Às vêzes, entretanto, aparece a ne cessidade de se utilizar como instrução a ser repetida (ou última ins trução de um grupo de instrução que deve ser repetida) a de desvio. É claro que êste não poderá ser incondicional, pois neste caso a va-riável conta ciclo não teria nenhuma função. O desvio condicionado (comando IF em Fortran) não poderá ser utilizado como instrução do fim da malha porque após a execução da mesma o fluxo apresentará mais do que uma alternativa, as quais fariam realmente o papel de última instrução a ser executada. O impasse poderá ser resolvido se a par-tir de cada uma das alternativas instruirmos para que se continue a execução do comando DO. Exemplificamos com os seguintes esquemas.





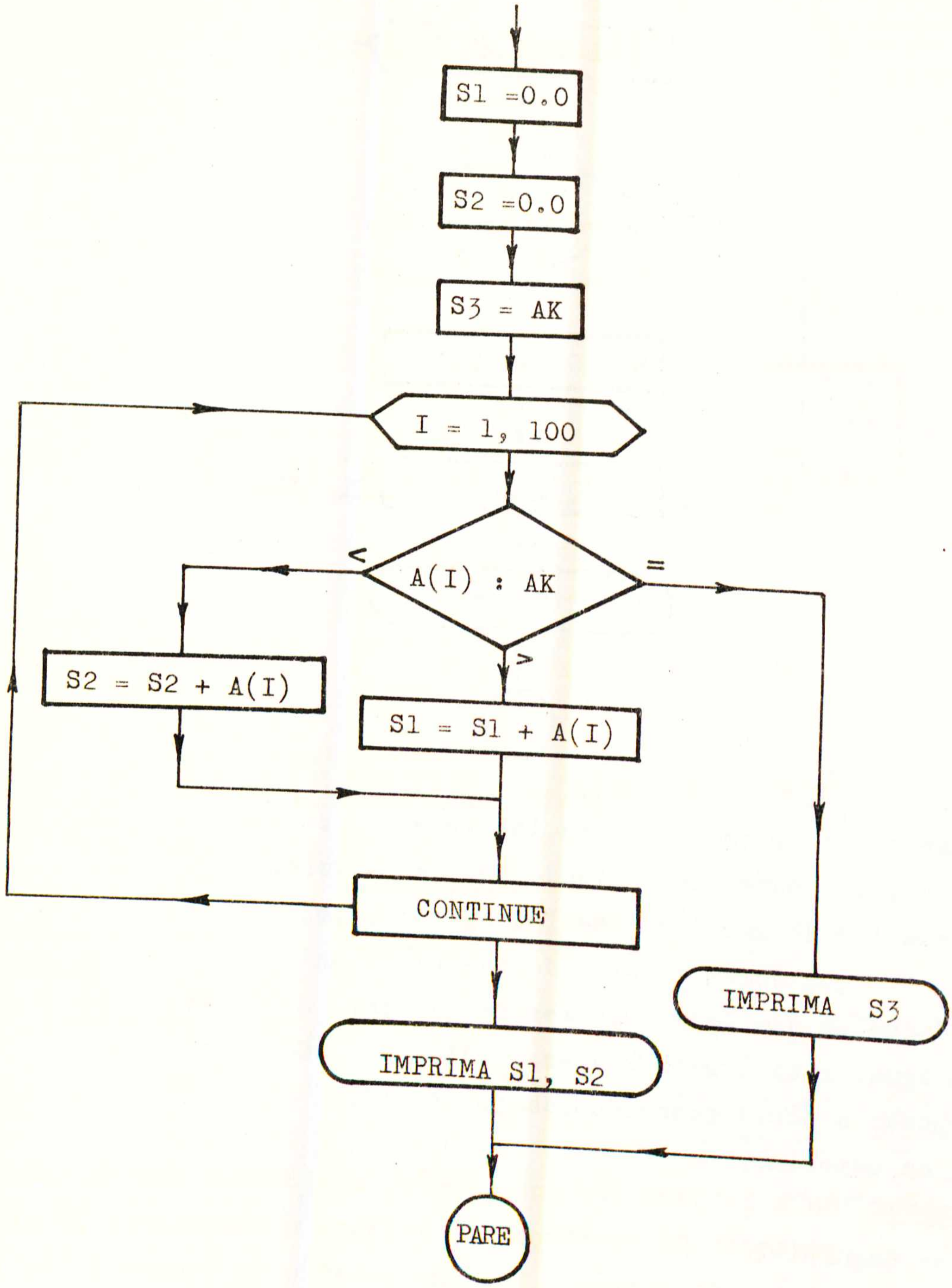
Portanto, se a instrução (ou última de um grupo de instruções) do fim da malha tiver que ser uma instrução de desvio condicionado, a dificul-dade ficará superada pelo emprêgo do comando CONTINUE. Vamos mostrar o uso dêsse novo comando com o seguinte problema: dada uma série de 100 números, imprimir a soma daqueles que superam o valor numérico do pri-meiro têrmo da série. Supomos que a série, representada pela variável indixada A (I), já se encontre armazenada na memória.



4- O comando IF permite que se interrompa a contagem dos ciclos do comando DO, se houver necessidade lógica na resolução de um problema.

Problemas: dada uma série de 100 números, imprimir a soma S1 daqueles que forem maiores do que uma quantidade constante AK e a soma S2 daqueles que forem menores do que AK. Se houver um só elemento da série igual a AK imprimir apenas S3 = AK. Como no problema precedente supomos a série representada por A (I), I = 1, 2, ..., 100, já guardada na memória.

(V. página seguinte)



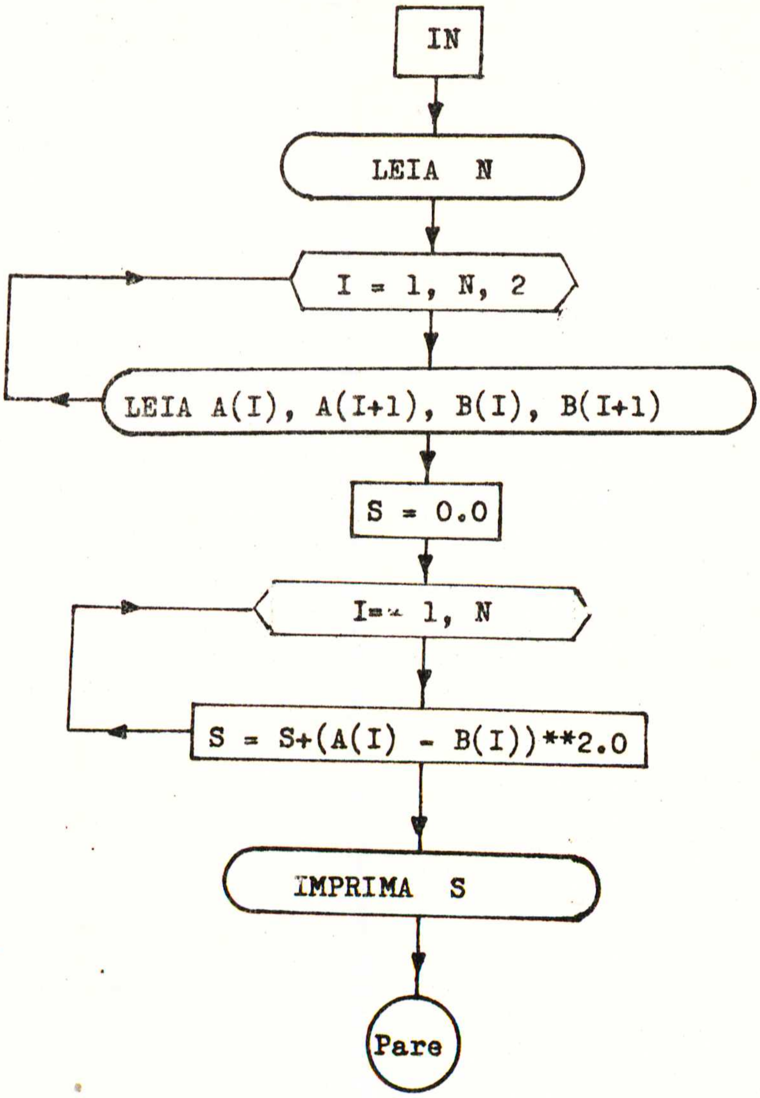
EXERCÍCIO 12

Dadas as matrizes A e B abaixo, calcular a expressão

$$S = \sum_{I=1}^{N} (A(I) - B(I))^2$$

$$A = \begin{pmatrix} A\ (1) \\ A\ (2) \\ \vdots \\ A\ (N) \end{pmatrix} \qquad B = \begin{pmatrix} B\ (1) \\ B\ (2) \\ \vdots \\ B\ (N) \end{pmatrix}$$

Diagrama de blocos

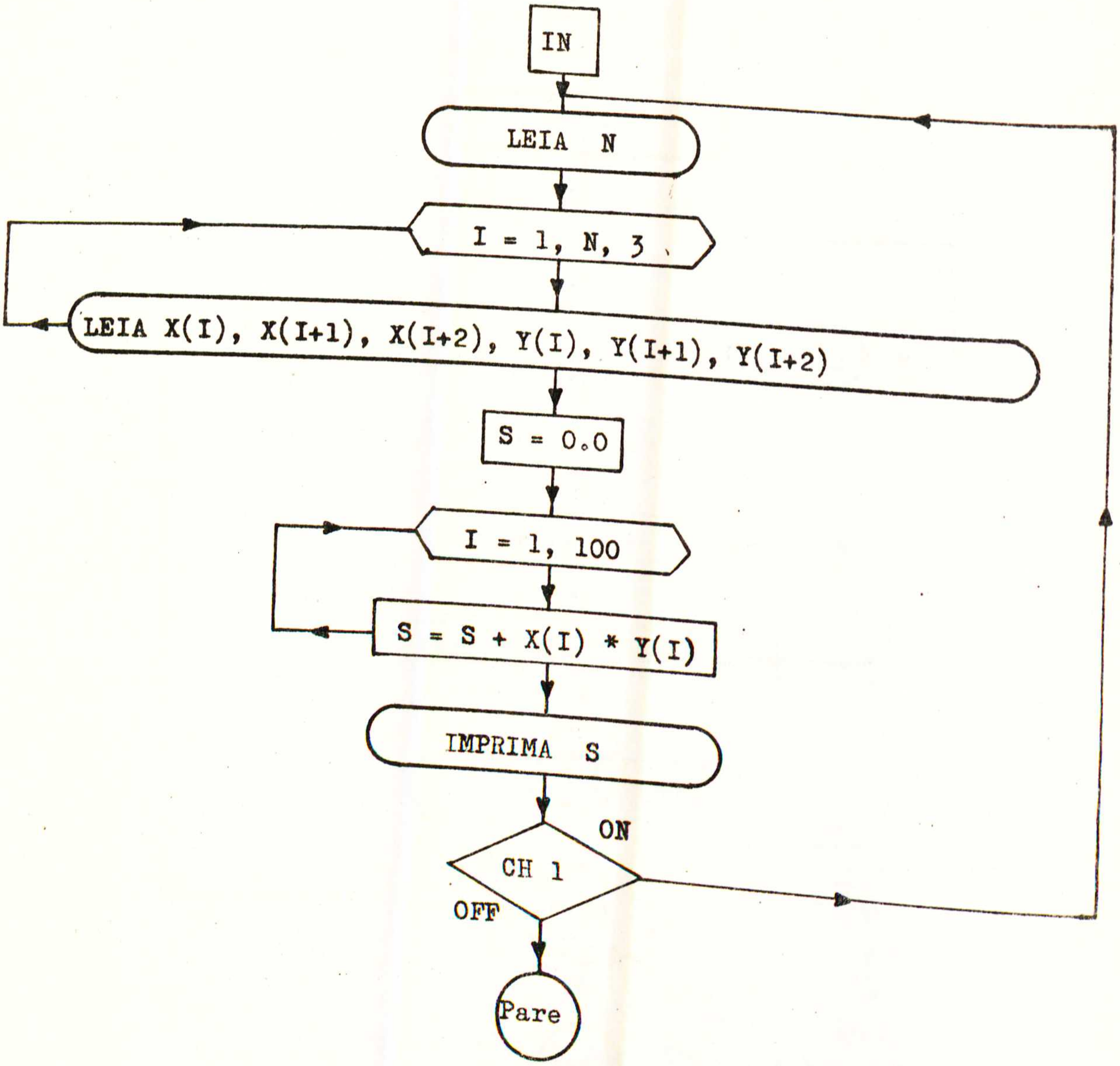


Observação: Se a leitura fôr por cartão, recordamos que em cada interpretação da instrução de leitura o computador lê as 80 posições do cartão. Por uma questão de economia (de tempo e material) devemos aproveitar, sempre que possível, tôdas as posições disponíveis. No diagrama acima prevemos a leitura de quatro dados por cartão.

EXERCÍCIO 13

Calcular $$S = \sum_{i=1}^{N} X_i \, Y_i$$

Diagrama de Blocos

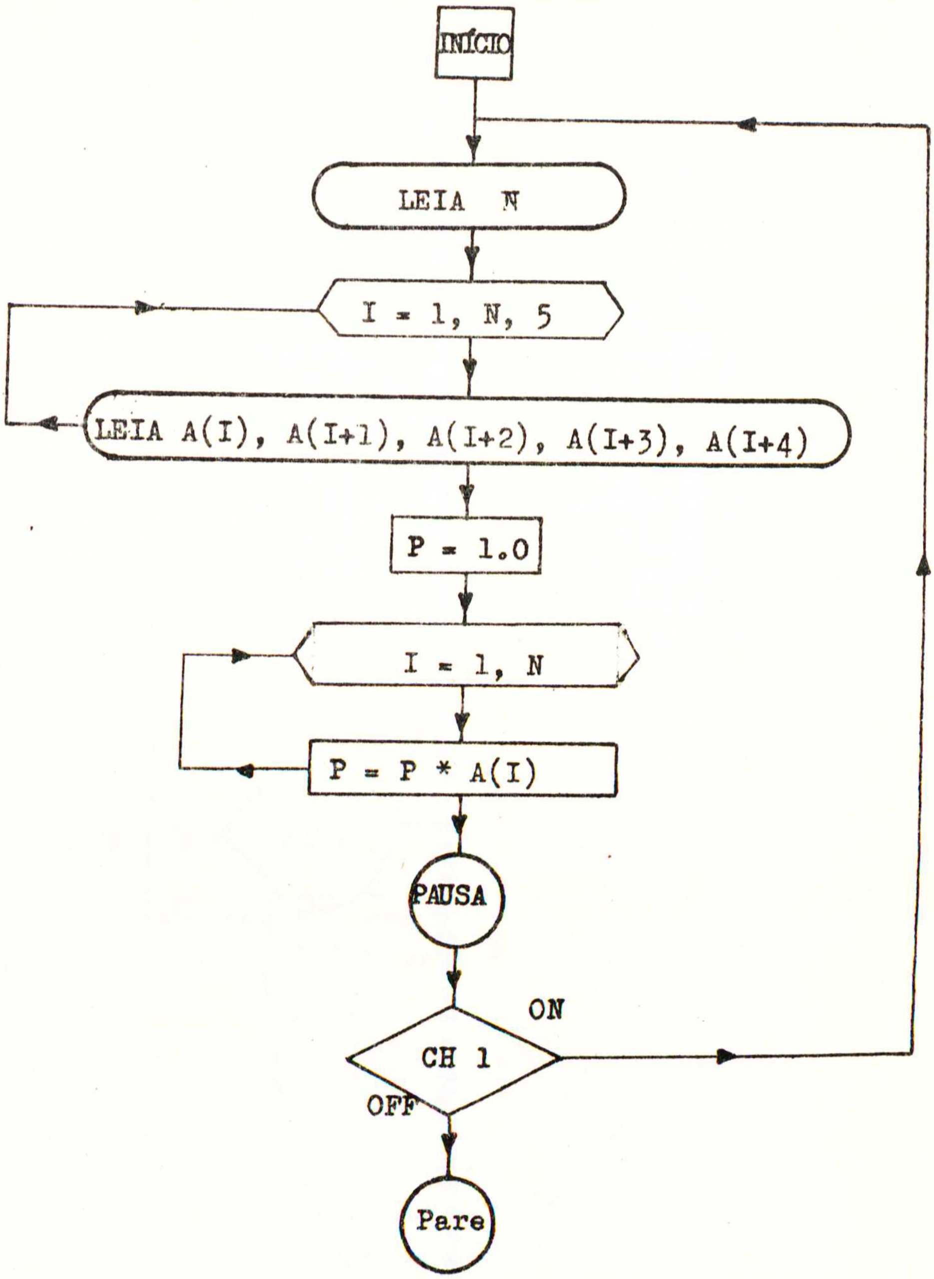


Observação: O computador IBM-1620 dispõe de quatro chaves de programa que são posicionadas manualmente. Conforme estejam liga-das ou não, desde que referidas na programação, determinam o desvio do fluxo do processamento para diferentes pontos do programa. No diagrama acima se a chave 1 estiver liga-da o computador se prepara para ler novo valor de N. Se desligada, para.

EXERCÍCIO 14

Calcular $P = \prod_{i=1}^{N} a_i$

Diagrama de blocos



Observação: O comando PAUSA (PAUSE) faz com que o computador pare até que apertemos a tecla START do controle. Esta parada temporária permite que o operador decida se o programa deve ser repetido para novo valor de N ou não. Se deseja parar definitivamente, antes de dar a partida, desliga a chave de programa, CHAVE 1.

EXERCÍCIO 15

Resolver com comandos em FORTRAN-1

15-1) Colocar o (i-1)-ésimo valor de A (I) no lugar do i-ésimo.

15-2) Transmitir para A (I) o valor de AMIN.

15-3) Transmitir para AMIN o i-ésimo valor de A (I).

15-4) Permutar as quantidades colocadas em A (I) e AMIN.

Solução

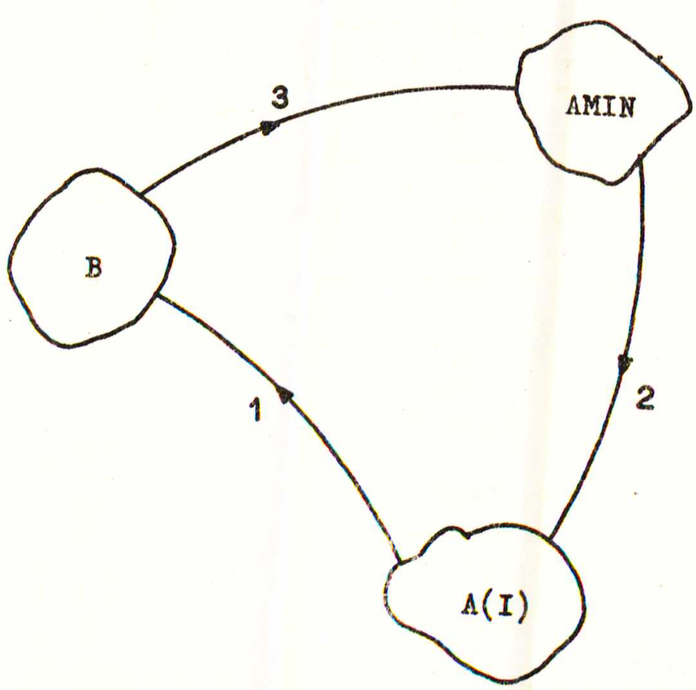
15-1) A (I) = A (I - 1)

15-2) A (I) = AMIN

15-3) AMIN = A (I)

15-4)
```
      B = A (I)
  A (I) = AMIN
  AMIN  = B
```

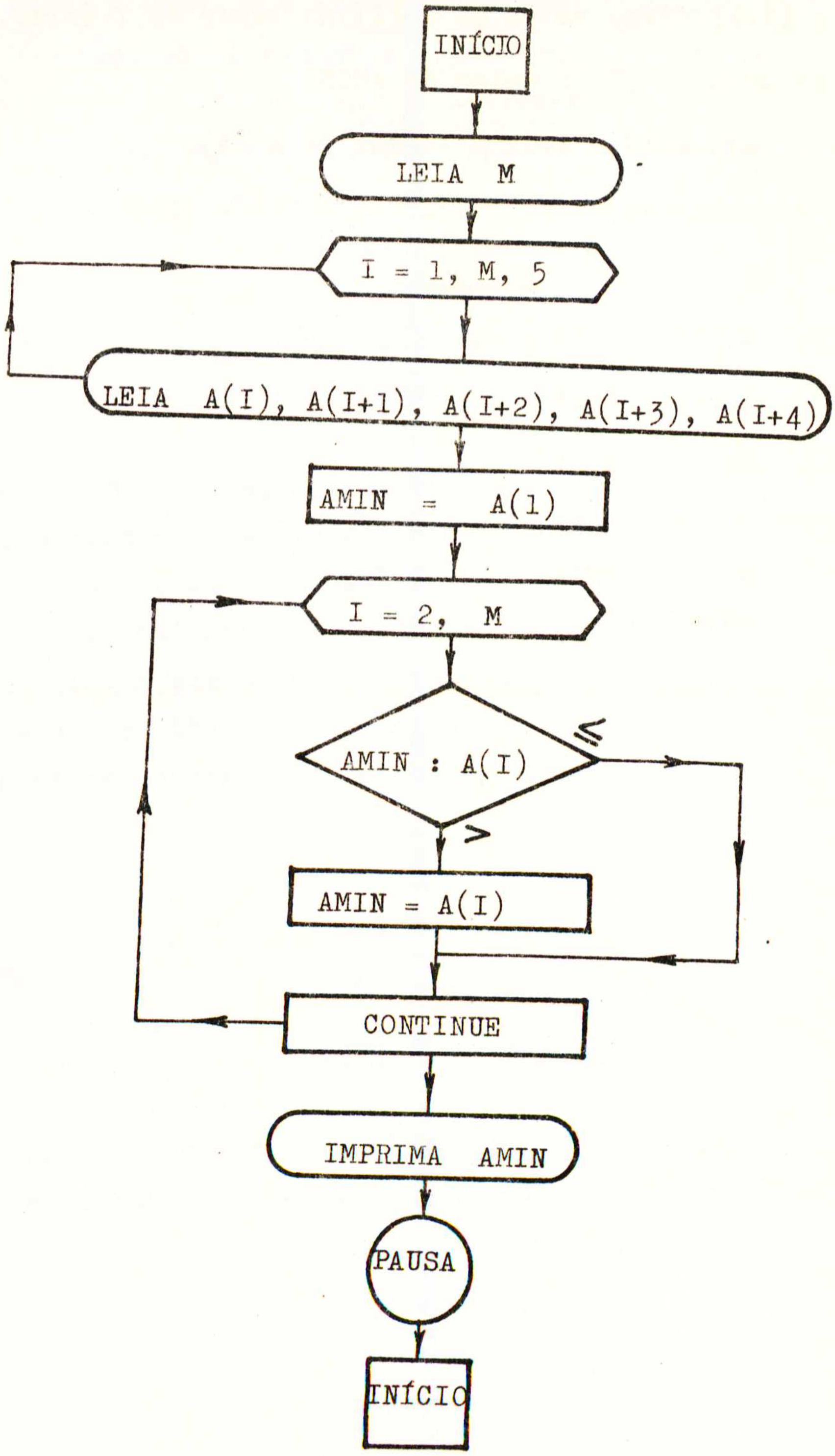
O que acontece internamente na memória está representado no esquema abaixo.



EXERCÍCIO 16

Calcular o mínimo de M números dados.

Diagramas de blocos



Observação: Notemos que o uso da variável AMIN é dispensável. O programa permanece válido eliminando-se o bloco contendo AMIN = A (1) e substituindo-se nos demais blocos AMIN por A (1).

EXERCÍCIO 17

Ordenar M números dados segundo os valores numéricos crescentes.

Rotina

Vamos exemplificar a rotina a ser adotada considerando os cin os primeiros naturais. Seja 3-5-4-1-2 a disposição dada.

Primeira etapa: comparamos sucessivamente cada número com o seguinte, permutando quando o primeiro fôr maior. Após compararmos dêsse modo todos os elementos do conjunto, resulta a seguinte disposição

3-4-1-2-5

Segunda etapa: repetimos o processo anterior e obtemos:

3-1-2-4-5

Terceira etapa: Idem, resultando, finalmente,

1-2-3-4-5

É fácil verificar que se a disposição fôsse a mais desfavorável, isto é, 5-4-3-2-1, a ordenação procurada exigiria quatro vêzes a repetição das comparações sucessivas. Portanto, se M fôsse o número de elementos do conjunto, o processo deveria repetir-se M-1 vêzes. Todavia, se a disposição não fôr a mais desfavorável, um número possìvelmente bem menor de repetição será necessário. A fim de se evitar repetir desnecessàriamente o processo descrito, introduzimos uma variável auxiliar, K. Tôda vez que houver uma permuta definimos K = 1. Na primeira repetição do processo em que não haja permuta definimos K = 0. Em seguida podemos imprimir os números que estarão ordenados conforme o problema pede.

Diagrama de blocos

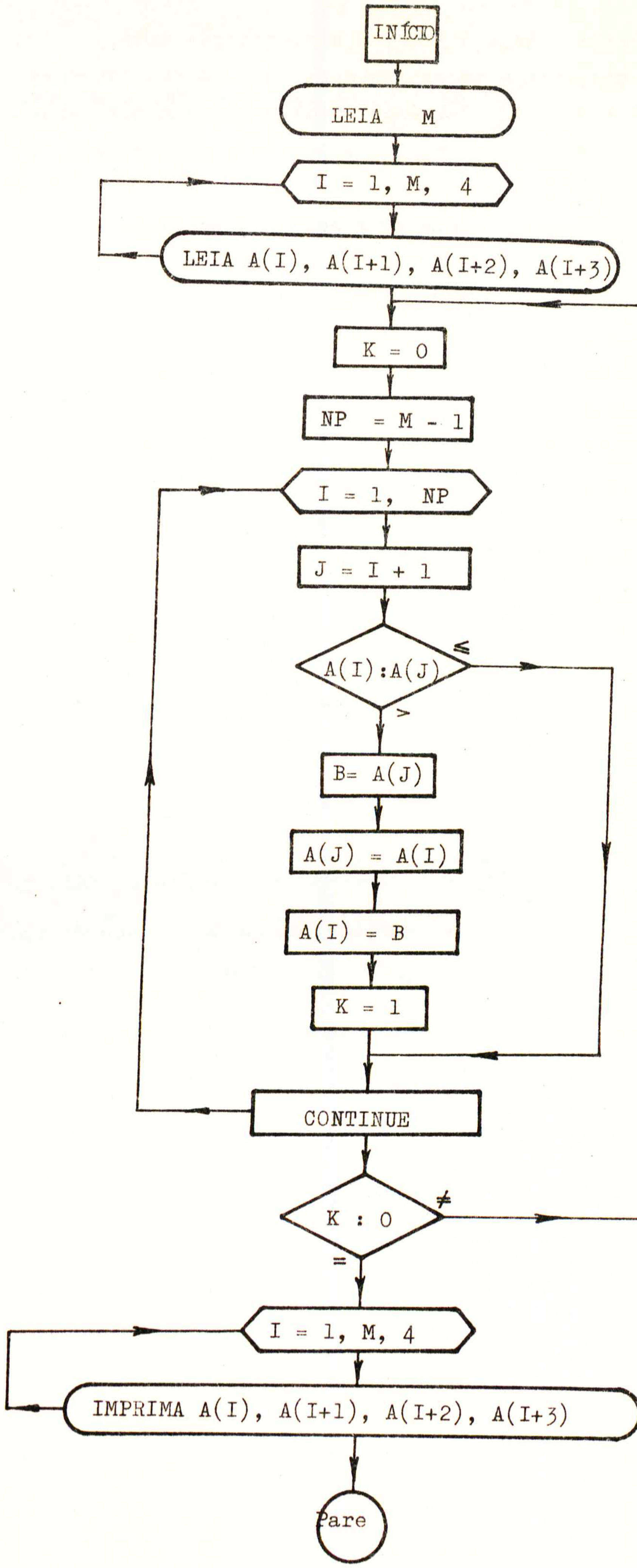
Observação: Notemos que a solução também pode ser encontrada permutando-se apenas os índices em vez de as respectivas quantidades. Bastaria fazermos:

IB = J

J = I

I = IB

(V. página seguinte)


INÍCIO
LEIA M
I = 1, M, 4
LEIA A(I), A(I+1), A(I+2), A(I+3)
K = 0
NP = M - 1
I = 1, NP
J = I + 1
A(I):A(J)
≤
>
B= A(J)
A(J) = A(I)
A(I) = B
K = 1
CONTINUE
K : 0
≠
=
I = 1, M, 4
IMPRIMA A(I), A(I+1), A(I+2), A(I+3)
Pare

EXERCÍCIO 18

Tabelar um polinômio de grau N para X variando entre IXMIN e IXMAX, com passo H.

Rotina

Seja o polinômio

$$P(X) = A(1)\ X^N + A(2)\ X^{N-1} + \dots\ A(N)\ X + A(N+1)$$

e consideremos a seguinte fatoração:

$$P(X) = \left[\left\{\left[A(1)\ X + A(2)\right]\ X + A\ (3)\right\}\ X + \dots + A(N)\right]\ X + A(N+1)$$

Façamos

$$B\ (1) = A\ (1)$$
$$B\ (2) = B\ (1)\ X + A\ (2)$$
$$B\ (3) = B\ (2)\ X + A\ (3)$$
$$\vdots$$
$$B(N+1) = B(N)\ X + A(N+1)$$

Resulta

$$P(X) = B(N+1)$$

Diagrama de blocos

Observação 1: A passagem $\boxed{X = I}$ deve-se ùnicamente ao fato de em FORTRAN-1 não ser permitido o uso misturado de variáveis de ponto fixo e de ponto flutuante, para a execução das operações indicadas no bloco

$$\boxed{B(J) = B(J-1) * X + A(J)}$$

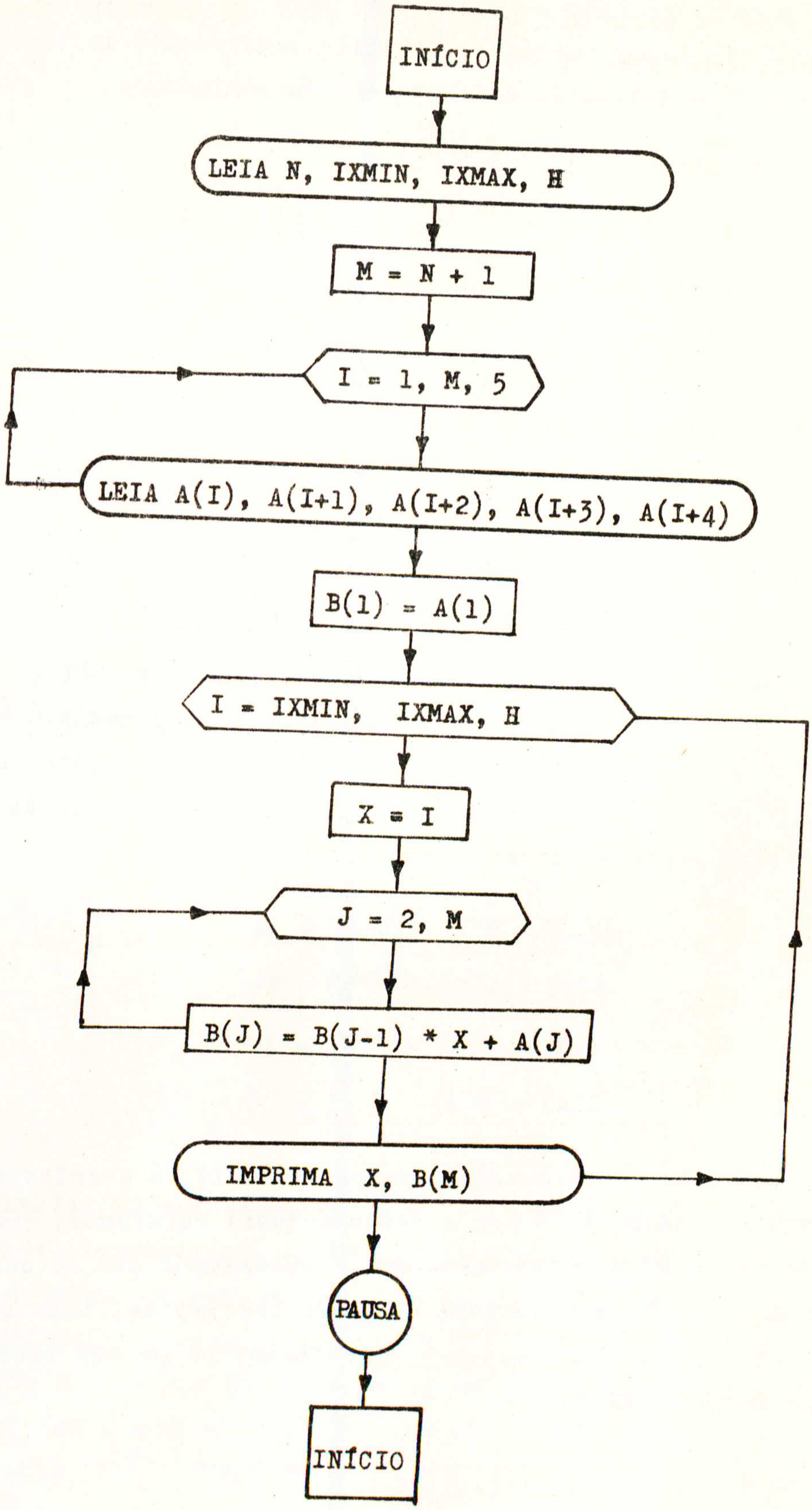
Observação 2: Em vez de N podemos ler diretamente M, eliminando o bloco $\boxed{M = N + 1}$. Devemos observar ainda não ser possível usar

$$\boxed{I = 1,\ N + 1}$$

no lugar de

$$\boxed{I = 1,\ M}$$

página seguinte)


INÍCIO
LEIA N, IXMIN, IXMAX, H
M = N + 1
I = 1, M, 5
LEIA A(I), A(I+1), A(I+2), A(I+3), A(I+4)
B(1) = A(1)
I = IXMIN, IXMAX, H
X = I
J = 2, M
B(J) = B(J-1) * X + A(J)
IMPRIMA X, B(M)
PAUSA
INÍCIO

Exercício 19

Tabelar um polinômio de grau N para X variando de XMIN a XMAX, segundo um passo unitário. Utilizar como rotina de trabalho o comportamento de uma tabela de diferenças para polinômios.

Rotina:

	x	x^3	Δx^3	$\Delta^2 x^3$	$\Delta^3 x^3$
	1	1 ←	7 ←	12 ←	6
XMIN —	2	8	19	18	6
	3	27	37	24	6
	4	64	61	30	
XMAX —	5	125			

Tabela de Diferenças de $f(x) = x^3$

Representemos os elementos de cada linha por D(I), I = = 1,2,3,4. Inicialmente são dados os valores correspondentes à primeira linha D(1) = 1, D(2) = 7, D(3) = 12 e D(4) = 6; para obtermos o valor do polinônio no ponto XMIN, modificamos os valores de D(I) , I = 1, 2, 3 e 4, conforme as relações:

$$D(1) = D(1) + D(2) = 8$$

$$D(2) = D(2) + D(3) = 19$$

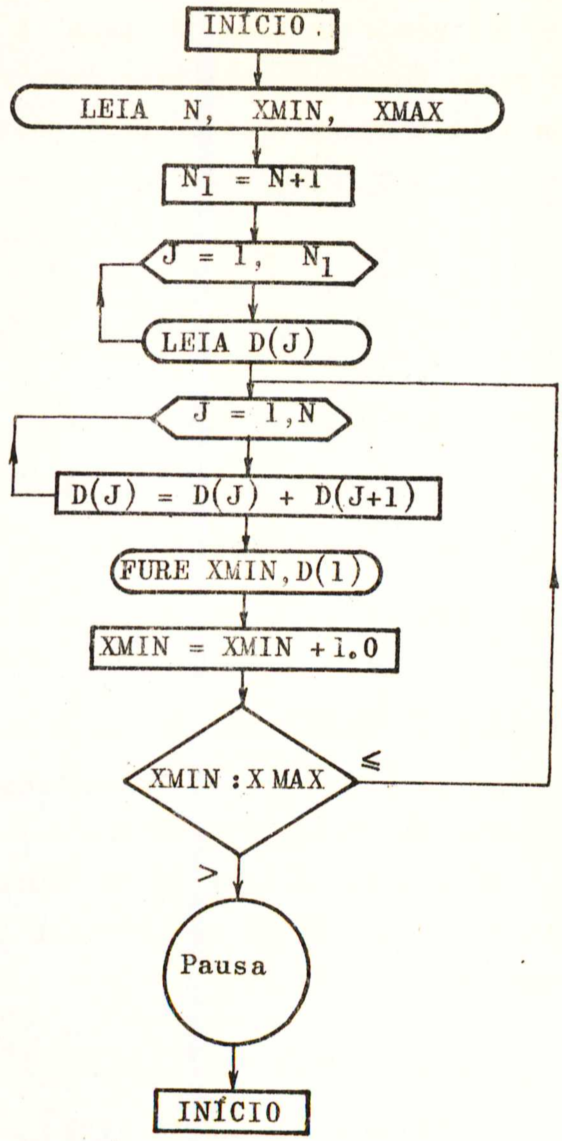
$$D(3) = D(3) + D(4) = 18$$

$$D(4) = D(4) \qquad = 6$$

Após essa substituição o valor de D(1) é o valor do polinômio em XMIN. Prosseguindo com o mesmo esquema obteremos, sucessiva-mente, todos valores a serem tabelados. Observemos que no caso de um polinômio de grau N, a diferença de ordem (n+1)ésima, isto é, D(N+1), permanece constante, o que aliás é característico de uma tabela de diferenças para polinômios.

Diagrama de Blocos

(v. página seguinte)



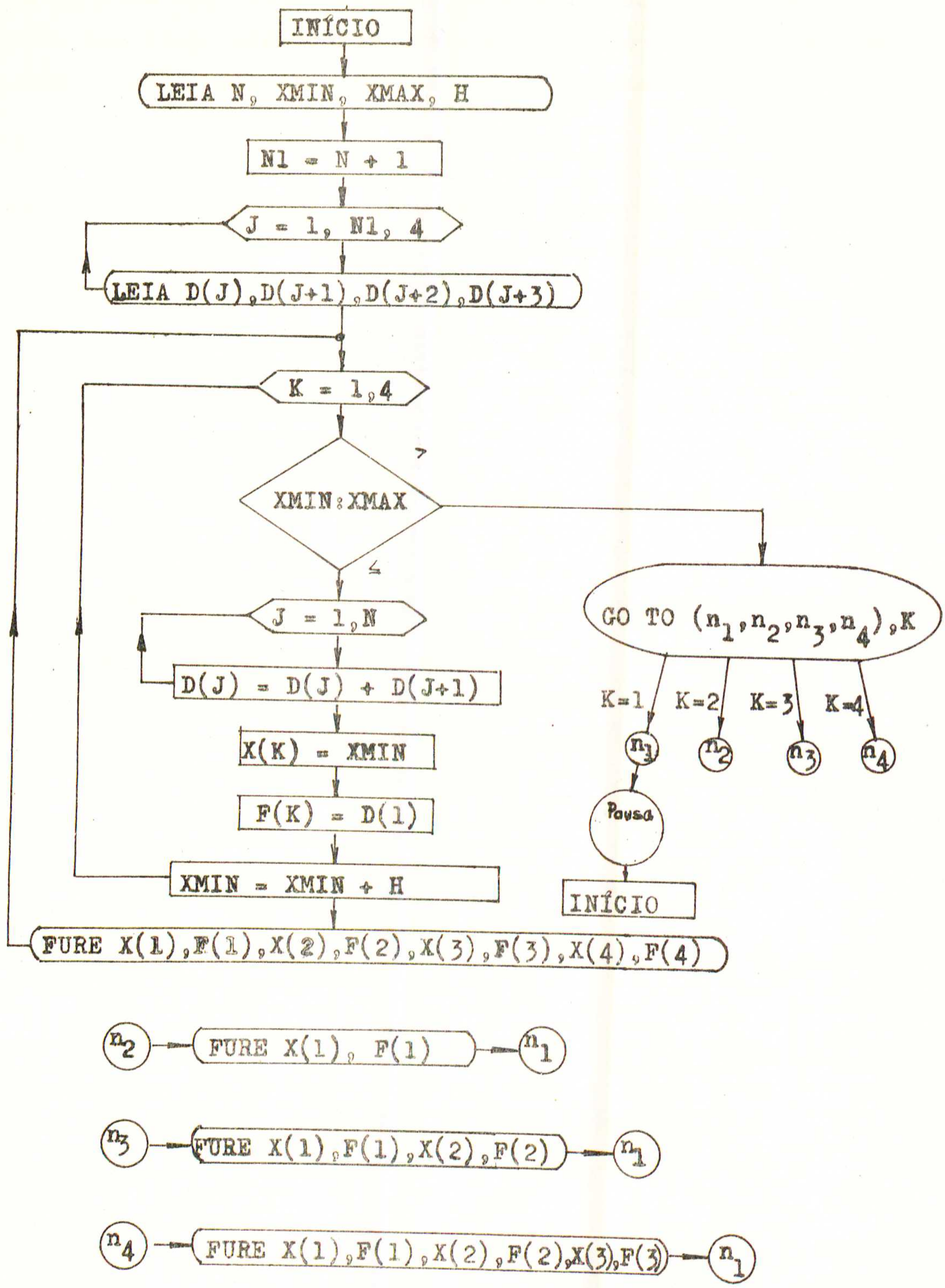
Exercício 20

Tabelar um polinômio de grau N segundo a rotina do exercício predecente, imprimindo quatro resultados por linha (ou cartão). Considerar um passo $H \neq 1$.

Rotina:

Será necessário guardar-se os valores tabelados de 4 em 4, o que faremos com o uso da variável indexada F(K). Imprimiremos cada grupo de 4 pares (X(K), F(K); K = 1, 2, 3, 4). Entretanto, a última linha (ou cartão) a ser impressa poderá conter 1, 2, 3 ou 4 pares de valôres. Para imprimir 1, 2 ou 3 pares de valores na última linha, faremos uso do comando COMPUTED TO GO. Notemos que cada comando de impressão ou perfuração de resultados corresponde a uma nova linha (ou cartão).

Diagrama de Blocos



Exercício 21

Dados M números, imprimir o número X(I) que mais se aproxima da respectiva média aritmética, sem, entretanto, utilizar o valor dessa média. Imprimir também o valor absoluto da soma S(I), abaixo definido.

Rotina:

Vamos basear a rotina na seguinte propriedade da média aritmética: "a soma dos resíduos é nula". Isto é,

$$S' = \sum_{j=1}^{M} (\bar{X} - X_j) = 0$$

sendo,

média aritmética $\bar{X} = (\sum_{i=1}^{M} X_i)/M$

resíduo do j-ésimo elemento $= (\bar{X} - X_j)$, $j = 1, 2, \ldots, M$

Se, no lugar de $\bar{X}$ colocarmos X_i, definimos M somas S(I), I = 1, 2, ..., M; então, o número X_i que mais se aproxima de $\bar{X}$ é aquele que gera o mínimo do conjunto $\{ |S(I)| \}$.

Expandindo a soma

$$S' = \sum_{j=1}^{M} (\bar{X} - X_j) = M\bar{X} - \sum_{j=1}^{M} X_j$$

definimos S(I) por

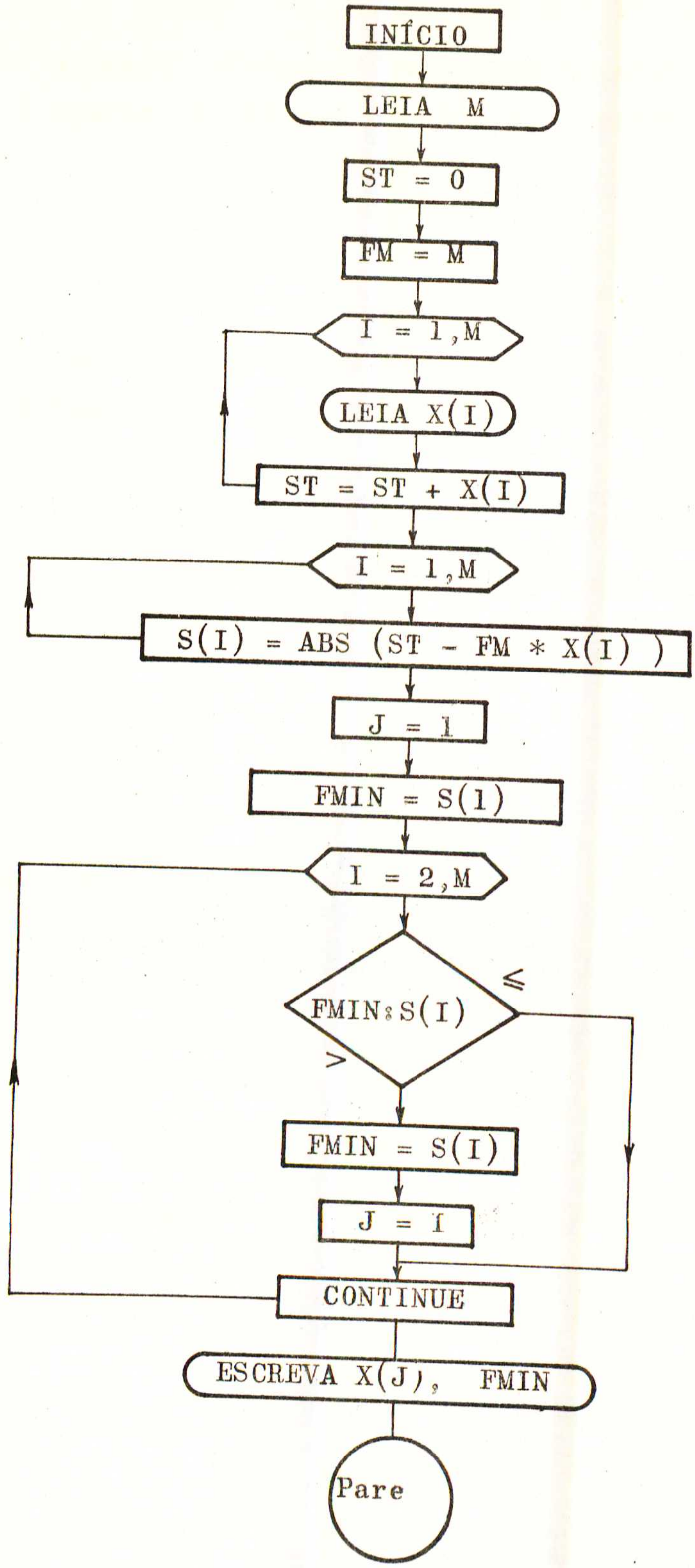
$$S(I) = MX(I) - ST ,$$

sendo

$$ST = \sum_{I=1}^{M} X(I)$$

Diagrama de Blocos

(v. página seguinte)



Exercício 22

Resolver com comandos em FORTRAN.

a) Colocar em IPINT a parte inteira da quantidade escrita em modalidade de ponto flutuante que se encontra armazenada na região A da memória. Por exemplo, se em A está guardada a quantidade 5.12 , em IPINT deverá constar 5.

Solução: IPINT = A

Chamamos a esta propriedade "Fixação". Podemos também fazer a "flutuação". Por exemplo, supondo-se que em I esteja 3 e, em N, 4, flutuamos escrevendo

A = I * N

e em A ficará o número 12.0.

b) Determinar se o número em ponto flutuante colocado em A tem parte inteira par. Prossiga para o comando 2 se A é impar e para o comando 3 se A fôr par.

Solução:

```
      IA = A
      IF (IA - IA/2 * 2) 2, 3, 2
    2 ...
      ...
      ...
      ...
    3 ...
```

c) Calcular e guardar em J o resto, em ponto fixo, da divisão IA/IB, sendo IA e IB números inteiros.

Solução:

```
      B(1) = IA/IB
      B(2) = IA
      B(3) = IB
      J = (B(2)/B(3) - B(1)) * B(3)
```

Se I = 5 e IB = 2, teremos conforme a solução acima, as seguintes passagens:

```
      2.0 = 5/2
      5.0 = 5
      2.0 = 2
      1   = (5.0/2.0 - 2.0) * 2.0
```

d) Determinar se o número colocado em IA é divisível por um número inteiro colocado em IB. Prossiga para o comando 2 se IA fôr divisível por IB. Caso contrário, para o comando 3.

Solução: IF (IA - IA/IB * IB) 3, 2, 3

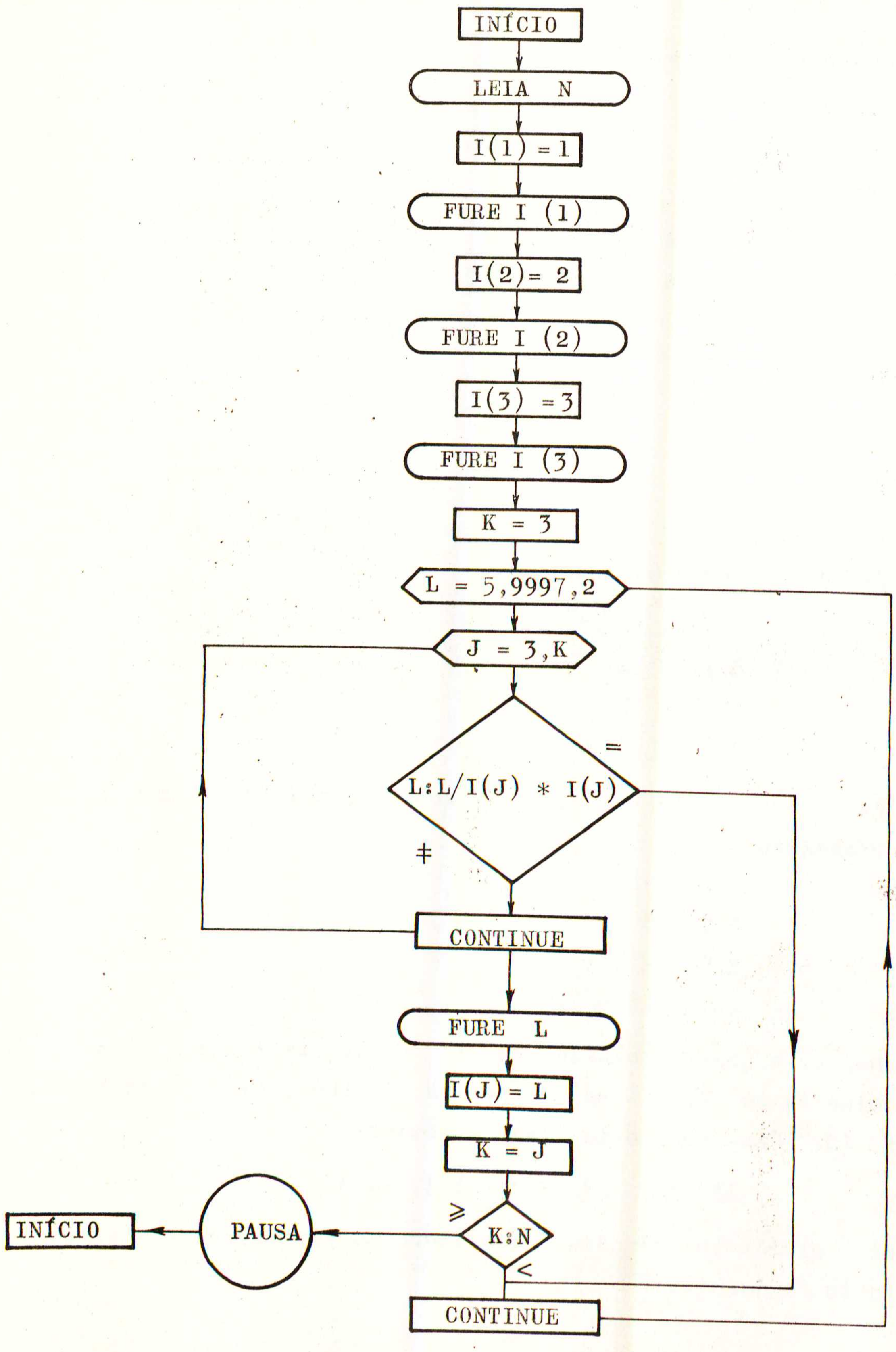
e) Resolver o exercício precedente considerando variáveis na modalidade de ponto flutuante.

Exercício 23

Imprimir uma tabela dos N primeiros números primos.

Rotina:

Para determinarmos se um número é primo ou não, adotaremos como rotina, a divisão dêsse número pelos primos anteriores. Observe-se que N é limitado, pois o maior primo que se pode pelo diagrama abaixo é inferior a 9999.



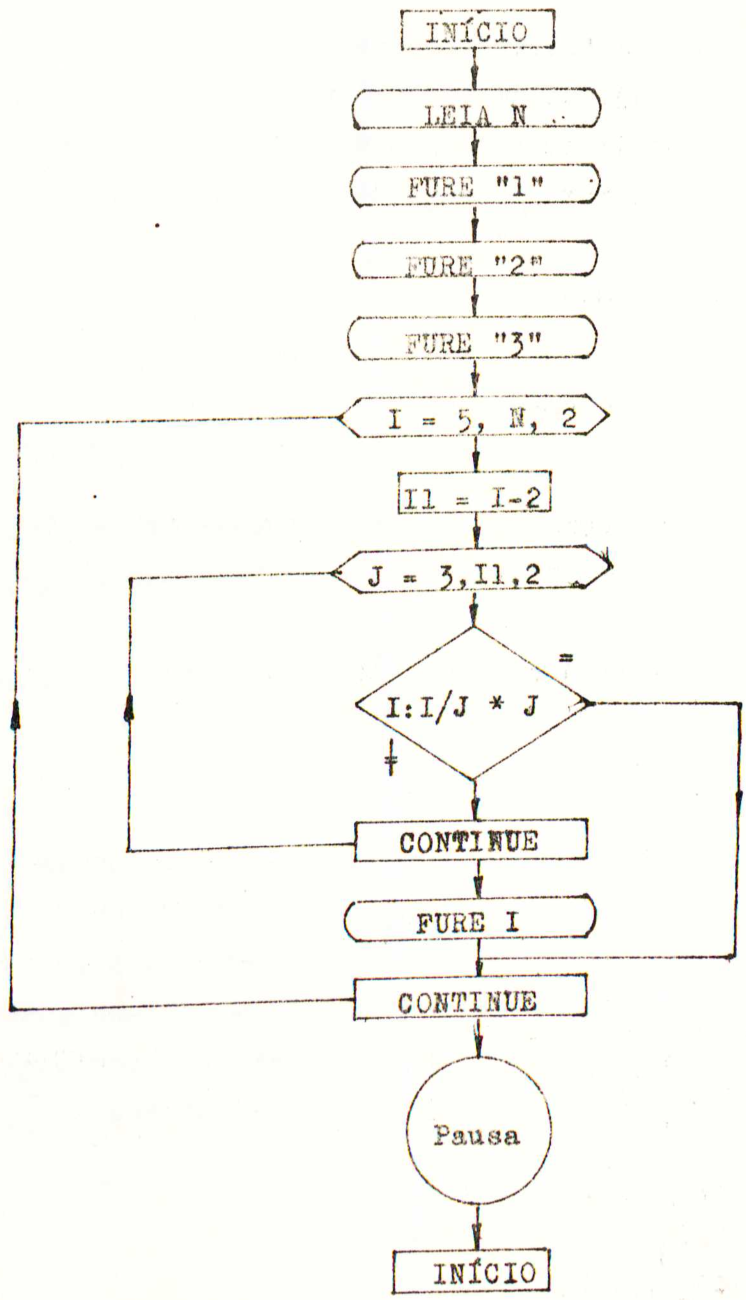
Exercício 24

Dado N, imprimir a tabela dos números primos de 1 a N.

Rotina:

Um número primo sòmente é divisível pela unidade e por si mesmo. Dado um número ímpar qualquer, testaremos a divisibilidade pelos ímpares anteriores, iniciando o processo pelo divisor 3.

Diagrama de blocos



Exercício proposto: Dado N, imprimir uma tabela de números primos de 1 a N, dispondo a tabela na forma de 4 colunas por linha. Não devem ser impressos zeros na última linha, na hipótese desta não comportar 4 elementos. Vide exercício 20. Elaborar o diagrama de blocos baseado na seguinte propriedade: "Se N não fôr divisível por nenhum primo entre 2 e $\sqrt{N}$, inclusive, então N é primo".

Exercício 25

Leitura dos elementos de uma matriz.

Rotina:

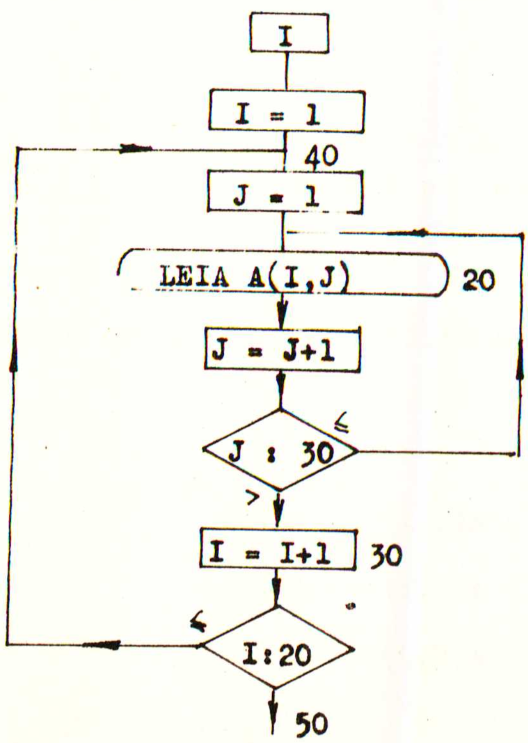
Do mesmo modo que fizemos uso da variável indixada A(I) para nos referirmos a um conjunto de elementos dispostos segundo uma única direção, também nos utilizamos do conceito de variável bi-indexada, que indicamos por A(I,J), para representar bi-dimensionalmente os elementos de uma matriz.

Por exemplo a matriz

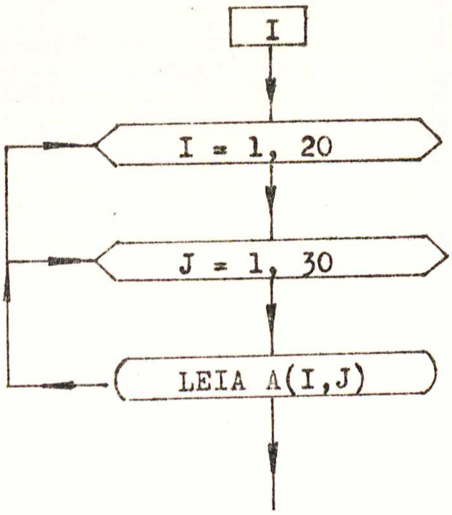
$$\begin{vmatrix} 1 & 2 & 3 \\ 7 & 2 & 8 \\ 4 & 7 & 9 \end{vmatrix} \quad \text{será representada por} \quad \begin{vmatrix} A(1,1) & A(1,2) & A(1,3) \\ A(2,1) & A(2,2) & A(2,3) \\ A(3,1) & A(3,2) & A(3,3) \end{vmatrix}$$

Quando nos referimos a A(3,2) o computador localiza na memória o valor de A(I,J) correspondente às coordenadas (3,2), considerando, portanto, o valor 7.

Vejamos como introduzir na memória os elementos de uma matriz 20 x 30.



Os elementos da matriz serão guardados linha após linha. É possível a leitura, coluna após coluna, bastando permutar a ordem dos índices I e J.



Esta representação, equivalente à primeira, é mais concisa e presta-se naturalmente à tradução em FORTRAN pela instrução DO.

<u>Programa em FORTRAN para leitura da matriz 20 x 30</u>

```
      DIMENSION A (20,30)
      I = 1
40    J = 1
20    READ 10, A(I,J)
      ...
      J = J + 1
      IF (J-30) 20,20,30
30    I = I + 1
      IF (I-20) 40,40,50
50    ...
      .
      .
      .
10    FORMAT (10,0)
```

```
      DIMENSION A(20,30)
10    FORMAT (F10.0)
      DO  20 I = 1,20
      DO  20 J = 1,30
20    READ 10 A(I,J)
      ...
```

```
      DIMENSION A(20,30)
10    FORMAT (F10.0)
      DO 20 I = 1,20
      DO 20 J = 1,20
20    READ 10, A(I,J)
```

<u>Exercícios propostos</u>:

1- Imprimir a soma de duas matrizes An x m e Bn x m.

2- Imprimir o produto de duas matrizes An x m e Bn x m.

<u>Exercício 26</u>

Verificar se uma matriz é simétrica, dentro de uma aproximação dada (ERRO).

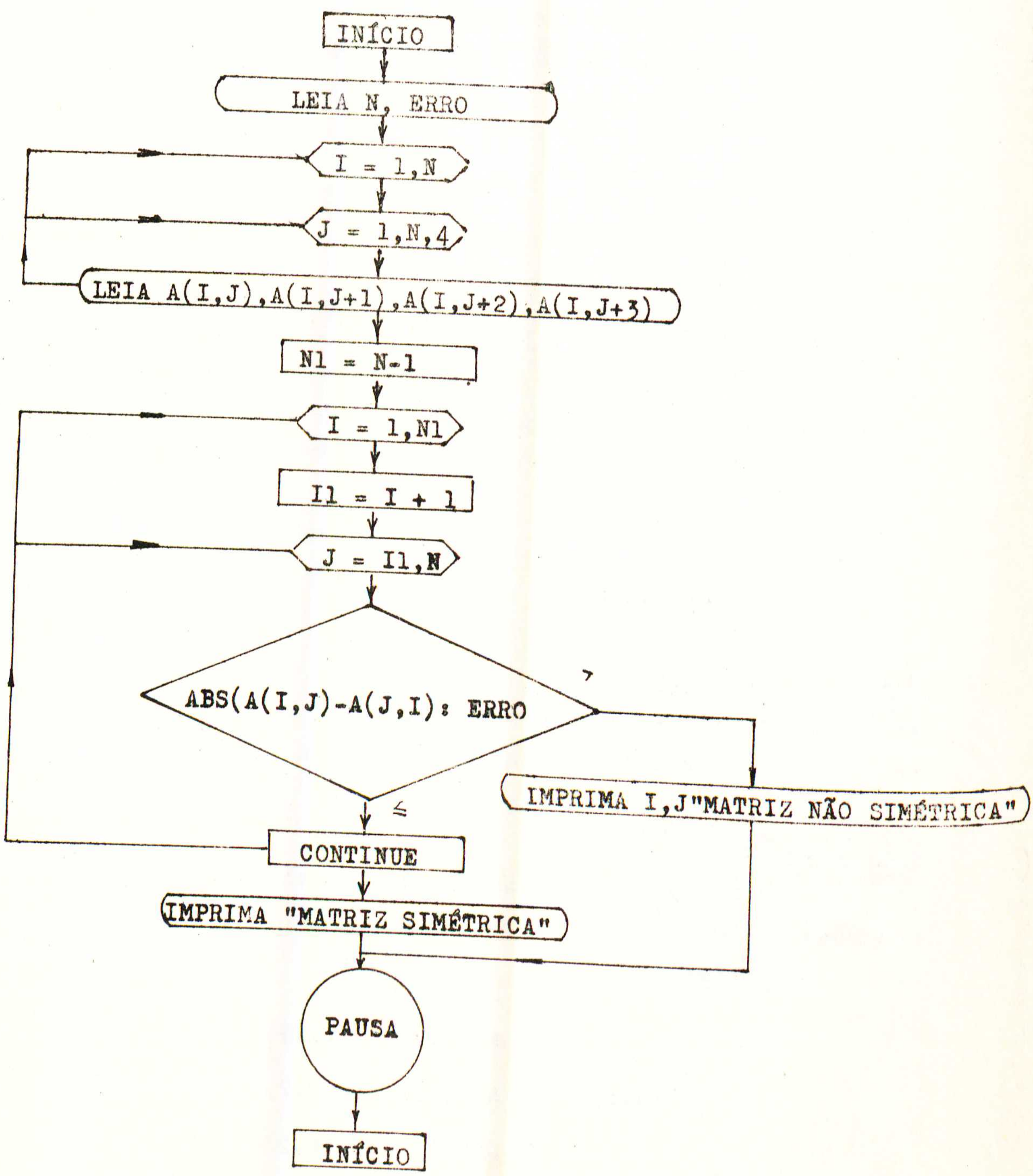
<u>Rotina</u>:

$$\text{A matriz} \quad \begin{pmatrix} A(1,1) & A(1,2) & A(1,N) \\ A(2,1) & A(2,2) & A(2,N) \\ A(N,1) & A(N,2) & A(N,N) \end{pmatrix}$$

é simétrica se A(I,J) = A(J,I), para todo I,J.

Vamos adotar o seguinte esquema para verificação das igualdades: Fixamos a I-ésima linha e consideramos apenas os têrmos que ficam à direita do têrmo A(I,I) pertencente à diagonal principal. É claro que repetiremos êsse processo apenas (N-1) vêzes, isto é, para I = 1, 2, ..., (N-1).

Diagrama de blocos



Exercício 27

Reduzir um sistema de N equações a N incógnitas à forma triangular.

Rotina:

Seja dada a matriz NXM, M = n + 1

$$\begin{bmatrix} A(1,1) & A(1,2) & \dots & A(1,N) & A(1,M) \\ A(2,1) & A(2,2) & \dots & A(2,N) & A(2,M) \\ \vdots & & & & \\ A(N,1) & A(N,2) & \dots & A(N,N) & A(N,M) \end{bmatrix}$$

Inicialmente reduziremos a zero os elementos da primeira coluna, exceto o primeiro, simplesmente pela substituição dos elementos de cada linha por:

$$A(I,J) = A(I,J - D2 * A(K,J), \qquad D2 = A(I,K)/A(K,K) \qquad (1)$$

onde

$$K = I, \quad K \leq J \leq M, \quad 2 \leq I \leq N$$

Se algum dos elementos A(I,K) fôr nulo, é desnecessária a substituição (1) para a I-ésima linha correspondente.

Obtemos, então, a seguinte situação:

$$\begin{bmatrix} A(1,1) & A(1,2) & \dots & A(1,N) & A(1,M) \\ 0 & A'(2,2) & \dots & A'(2,N) & A'(2,M) \\ 0 & A'(3,2) & \dots & A'(3,N) & A'(3,M) \\ \vdots & & & & \\ 0 & A'(N,2) & \dots & A'(N,N) & A'(N,M) \end{bmatrix}$$

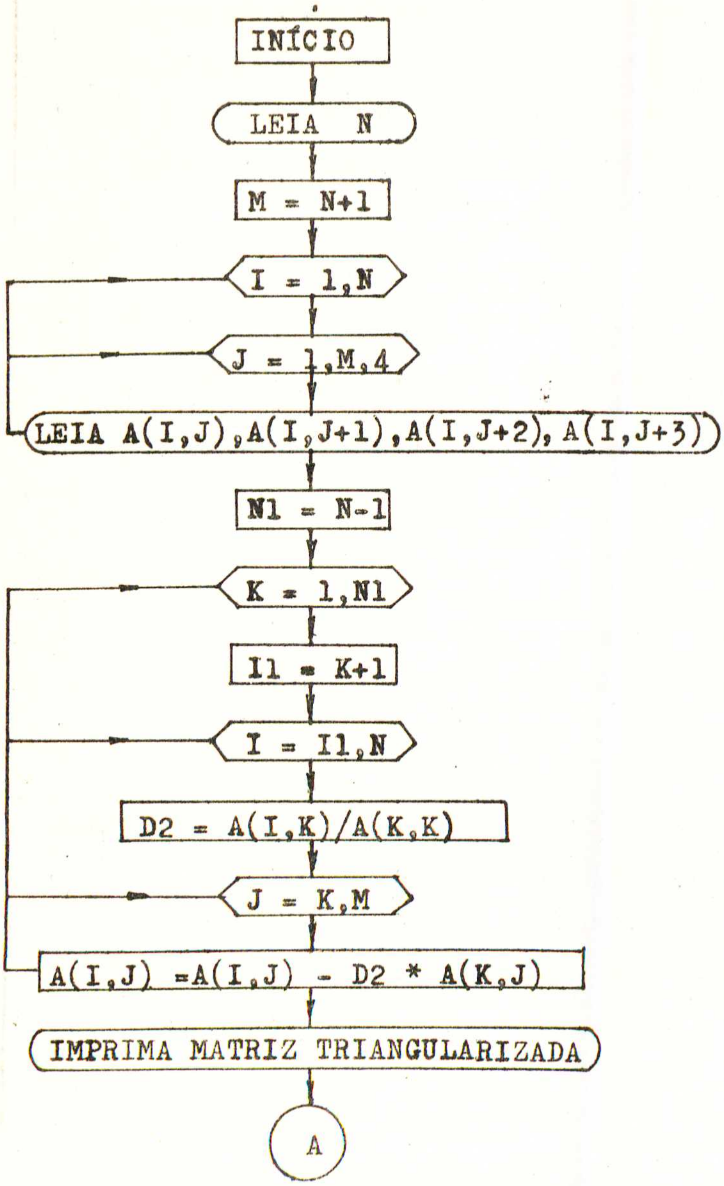
Aplicamos, novamente, o processo em relação à matriz A' (N-1) x (M-1). Basta fazer em (1)

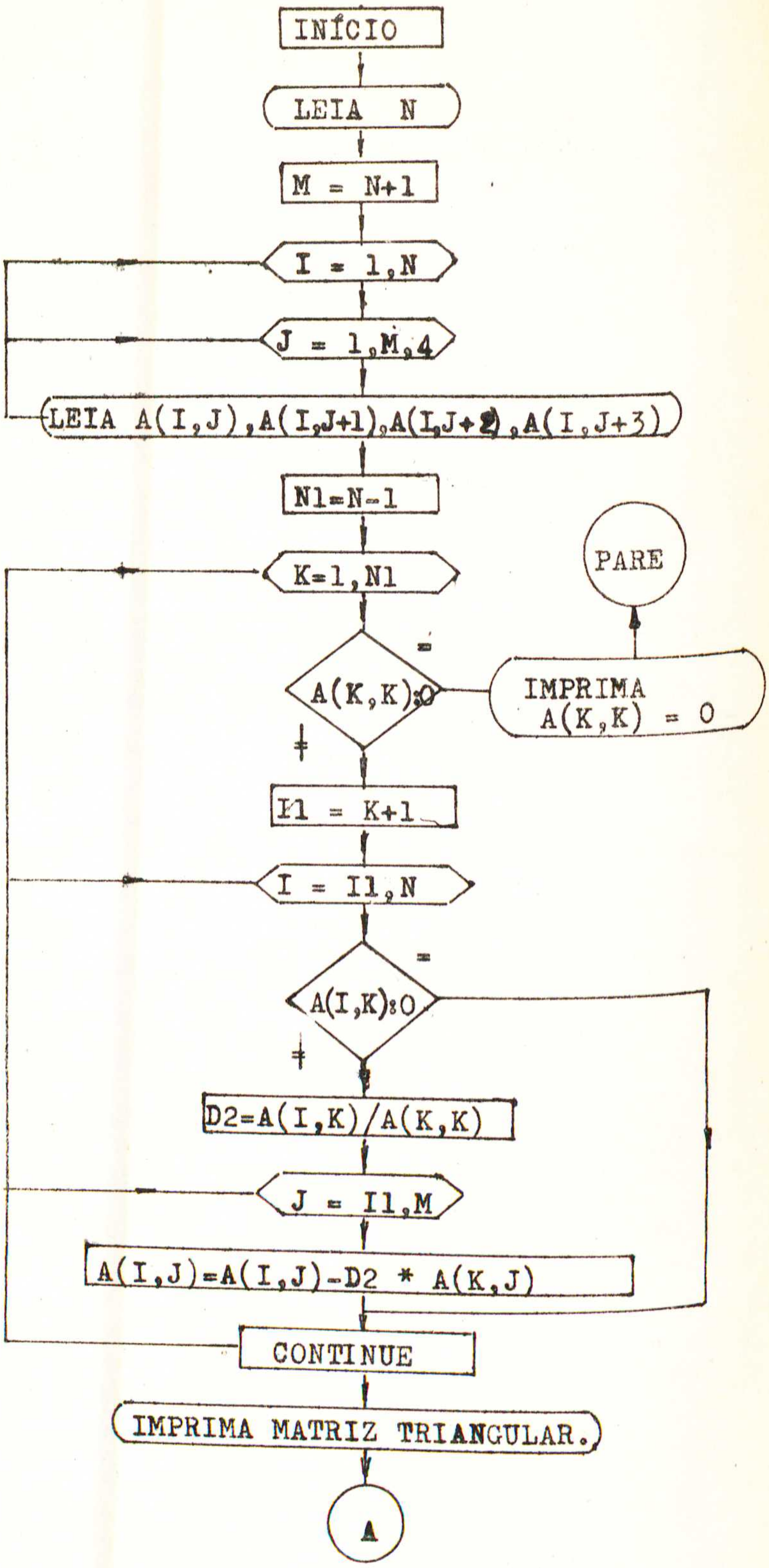
$$K = 2, \quad K \leq J \leq M, \quad 3 \leq I \leq N$$

Após a repetição dêsse processo (N-1) vêzes obteremos a solução de nosso problema. Notemos que as transformações adotadas não alteram a solução do sistema de equações.

Diagrama de blocos

(v. página seguinte)





<u>Exercício proposto:</u>

Dizer das diferenças entre êstes dois diagramas e apontar quais as vantagens existentes no segundo em relação ao primeiro.

Exercício 28

Calcular o valor de um determinante.

Rotina:

Por meio de transformações convenientes esquematizamos abaixo cada uma das passagens que conduzem à triangularização da matriz, sem alteração do seu determinante. Recordemos que o valor de um determinante de uma matriz na forma triangular é o produto dos elementos da diagonal principal.

$$\begin{vmatrix} A(1,1) & A(1,2) & A(1,3) \\ A(2,1) & A(2,2) & A(2,3) \\ A(3,1) & A(3,2) & A(3,3) \end{vmatrix} \longrightarrow \begin{vmatrix} A(1,1) & A(1,2) & A(1,3) \\ 0 & A'(2,2) & A'(2,3) \\ 0 & A'(3,1) & A'(3,3) \end{vmatrix}$$

I II

$$\longrightarrow \begin{vmatrix} A(1,1) & A(1,2) & A(1,3) \\ 0 & A''(2,2) & A''(2,3) \\ 0 & 0 & A''(3,3) \end{vmatrix}$$

III

Se $A(1,1) \neq 0$ substituimos os elementos das outras linhas por:

I = 2 (Segunda linha)

a) A(2,J) = A(2,J) - D2 * A(1,J), D2 = A(2,1)/A(1,1), j = 1,2,3

I = 3 (Terceira linha)

b) A(3,J) = A(3,J) - D2 * A(1,3), D2 = A(3,1)/A(1,1), j = 1,2,3

Obtemos dêsse modo a situação II.

Repetindo o mesmo esquema com relação à matriz

$$\begin{vmatrix} A(2,2) & A(2,3) \\ A(3,1) & A(3,3) \end{vmatrix}$$

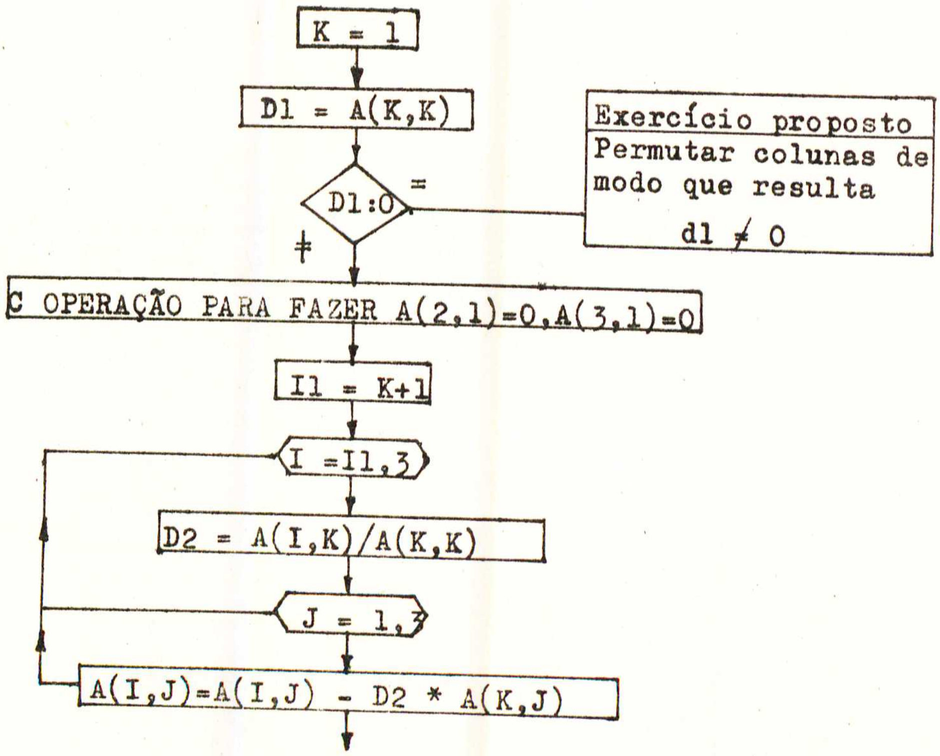
chegamos, finalmente, à situação III.

Observamos que as substituições da 2ª e 3ª linhas pelas combinações lineares feitas não alteram o valor do determinante.

Diagrama de blocos

(v. página seguinte)

<u>Diagrama de blocos que programa a passagem da situação 1 para a situação 2</u>

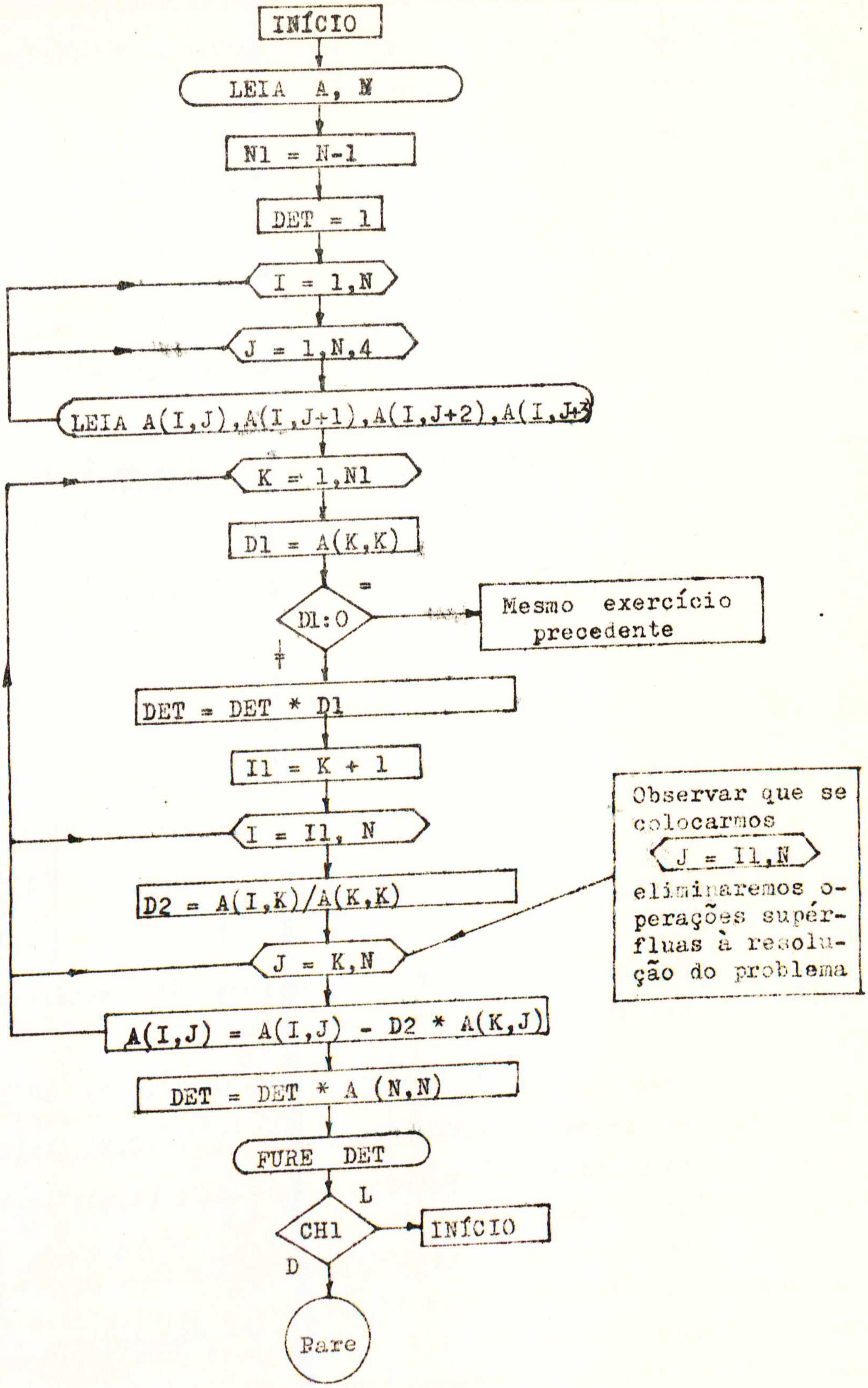


Daremos em seguida o diagrama de blocos que prevê o cálculo de um determinante de ordem qualquer, deixando, como exercício, o desenvolvimento da parte referente ao caso em que seja nulo o valor de algum elemento da diagonal principal.

Observamos que, no caso do exemplo acima, a triangularização do determinante de terceira ordem nos leva a repetir duas vêzes a rotina acima esquematizada (I = 1,2). De um modo geral, se N fôr a ordem do determinante, o processo deverá ser repetido N_1 vêzes, sendo $N_1 = N - 1$.

<u>Diagrama de blocos para cálculo de um determinante de ordem N</u>

(v. página seguinte)


INÍCIO
LEIA A, N
N1 = N-1
DET = 1
I = 1,N
J = 1,N,4
LEIA A(I,J),A(I,J+1),A(I,J+2),A(I,J+3)
K = 1,N1
D1 = A(K,K)
D1:0
=
≠
Mesmo exercício precedente
DET = DET * D1
I1 = K + 1
I = I1, N
D2 = A(I,K)/A(K,K)
J = K,N
A(I,J) = A(I,J) - D2 * A(K,J)
DET = DET * A (N,N)
FURE DET
CH1
L
D
INÍCIO
Pare
Observar que se colocarmos
J = I1,N
eliminaremos operações supérfluas à resolução do problema

Exercício 29

Resolver um sistema de N equações a N incógnitas pelo Método de Gauss por diagonalização.

Rotina:

Vamos exemplificar com um sistema de 3 equações a 3 incógnitas:

$$
\begin{aligned}
A(1,1)\ X(1) + A(1,2)\ X(2) + A(1,3)\ X(3) &= A(1,4) \\
A(2,1)\ X(1) + A(2,2)\ X(2) + A(2,3)\ X(3) &= A(2,4) \\
A(3,1)\ X(1) + A(3,2)\ X(2) + A(3,3)\ X(3) &= A(3,4)
\end{aligned}
\qquad (1)
$$

Dada a matriz 3 x 4

$$
\begin{pmatrix}
A(1,1) & A(1,2) & A(1,3) & A(1,4) \\
A(2,1) & A(2,2) & A(2,3) & A(2,4) \\
A(3,1) & A(3,2) & A(3,3) & A(3,4)
\end{pmatrix}
$$

desejamos chegar à matriz

$$
\begin{pmatrix}
1 & 0 & 0 & \bar{A}(1,4) \\
0 & 1 & 0 & \bar{A}(2,4) \\
0 & 0 & 1 & \bar{A}(3,4)
\end{pmatrix}
$$

A solução do sistema (1) será dado por $\begin{pmatrix} \bar{A}(1,4) \\ \bar{A}(2,4) \\ \bar{A}(3,4) \end{pmatrix}$

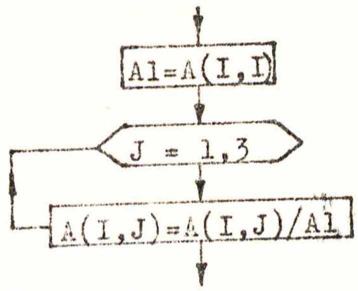
Esquematizamos abaixo as passagens intermediárias que nos levam à situação procurada:

$$
\underset{1}{\begin{pmatrix}
1 & A'(1,2) & A'(1,3) & A'(1,4) \\
0 & A'(2,2) & A'(2,3) & A'(2,4) \\
0 & A'(3,2) & A'(3,3) & A'(3,4)
\end{pmatrix}}
\longrightarrow
\underset{2}{\begin{pmatrix}
1 & 0 & A''(1,3) & A''(1,4) \\
0 & 1 & A''(2,3) & A''(2,4) \\
0 & 0 & A''(3,3) & A''(3,4)
\end{pmatrix}}
\longrightarrow
$$

$$
\longrightarrow
\underset{3}{\begin{pmatrix}
1 & 0 & 0 & \bar{A}(1,4) \\
0 & 1 & 0 & \bar{A}(2,4) \\
0 & 0 & 1 & \bar{A}(3,4)
\end{pmatrix}}
$$

Descrição para obtenção da I-ésima situação a partir da anterior

a) Dividimos os elementos da i-ésima linha por A(I,I)



b) Substituimos os elementos de cada J-ésima linha pela diferença en-- tre os mesmos e os correspondentes da i-ésima linha multiplicados por A(K,I)

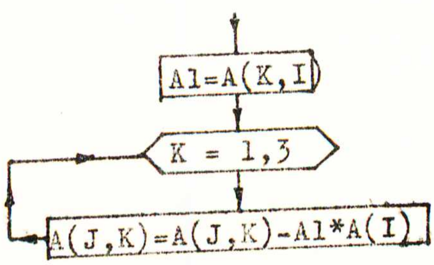
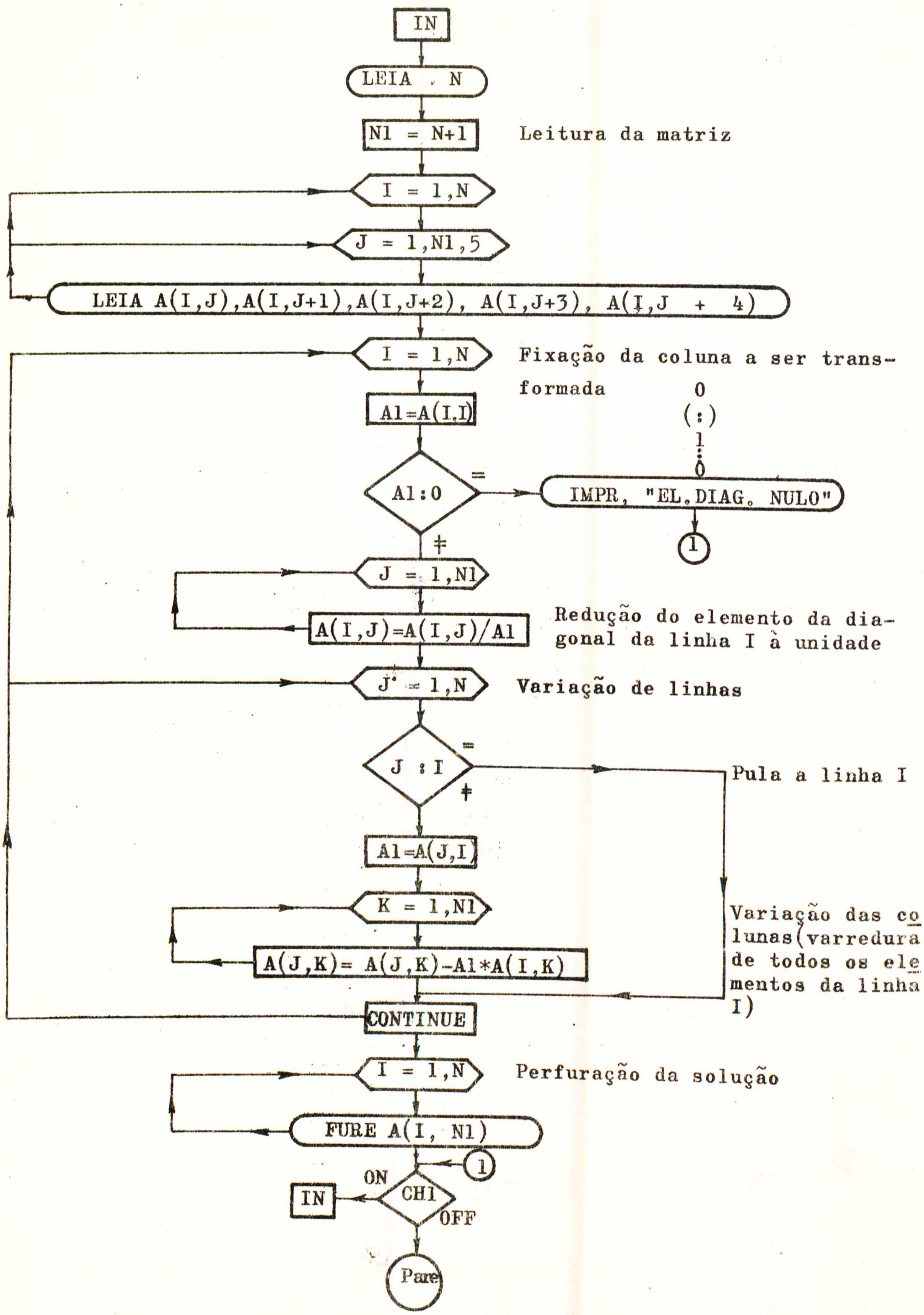


Diagrama de blocos

(v. página seguinte)


IN
LEIA . N
N1 = N+1
Leitura da matriz
I = 1,N
J = 1,N1,5
LEIA A(I,J),A(I,J+1),A(I,J+2), A(I,J+3), A(I,J + 4)
I = 1,N
Fixação da coluna a ser transformada
A1=A(I,I)
A1:0
=
≠
IMPR, "EL.DIAG. NULO"
1
J = 1,N1
A(I,J)=A(I,J)/A1
Redução do elemento da diagonal da linha I à unidade
J = 1,N
Variação de linhas
J : I
Pula a linha I
A1=A(J,I)
K = 1,N1
A(J,K)= A(J,K)-A1*A(I,K)
Variação das colunas (varredura de todos os elementos da linha I)
CONTINUE
I = 1,N
Perfuração da solução
FURE A(I, N1)
ON
CH1
OFF
Pare

Exercício 30

Resolver um sistema triangular de N equações a N incógnitas.

Rotina:

Seja dado o sistema

X(1)	X(2)	...	X(N-1)	X(N)	B(I,N+1)
A(1,1)	A(1,2)	...	A(1,N-1)	A(1,N)	A(1,N+1)
0	A(2,2)	...	A(2,N-1)	A(2,N)	A(2,N+1)
0	0	...	A(3,N-1)	A(3,N)	A(3,N+1)
⋮			⋮	⋮	⋮
0	0		0	A(N,N)	A(N,N+1)

Vamos representar a solução X(I)—por—A(I,N+1). A vantagem desta representação é a simplificação do programa, ocasionando economia em tempo e espaço.

Valor de X(N)

$$A(N,N)\,X(N) = A(N,N+1) \quad \therefore \quad X(N) = A(N,N+1) = A(N,N+1)/A(N,N)$$

Valor de X(N-1)

$$A(N-1,N-1)\,X(N-1) + A(N-1,N)\,X(N) = A(N-1,N+1) \quad \therefore$$

$$X(N-1) = A(N-1,N+1) = [A(N-1,N+1) - A(N-1,N)\,A(N,N+1)] \;/\; A(N-1,N-1)$$

Valor de X(N-2)

$$A(N-2,N-2)\,X(N-2) + A(N-2,N-1)\,X(N-1) + A(N-2,N)\,X(N) = A(N-2,N+1) \quad \therefore$$

$$X(N-2) = A(N-2,N+1) =$$

$$= \left\{A(N-2,N+1) - \left[A(N-2,N-1)A(N-1,N+1) + A(N-2,N)A(N,N+1)\right]\right\} \;/\; A(N-2,N-2)$$

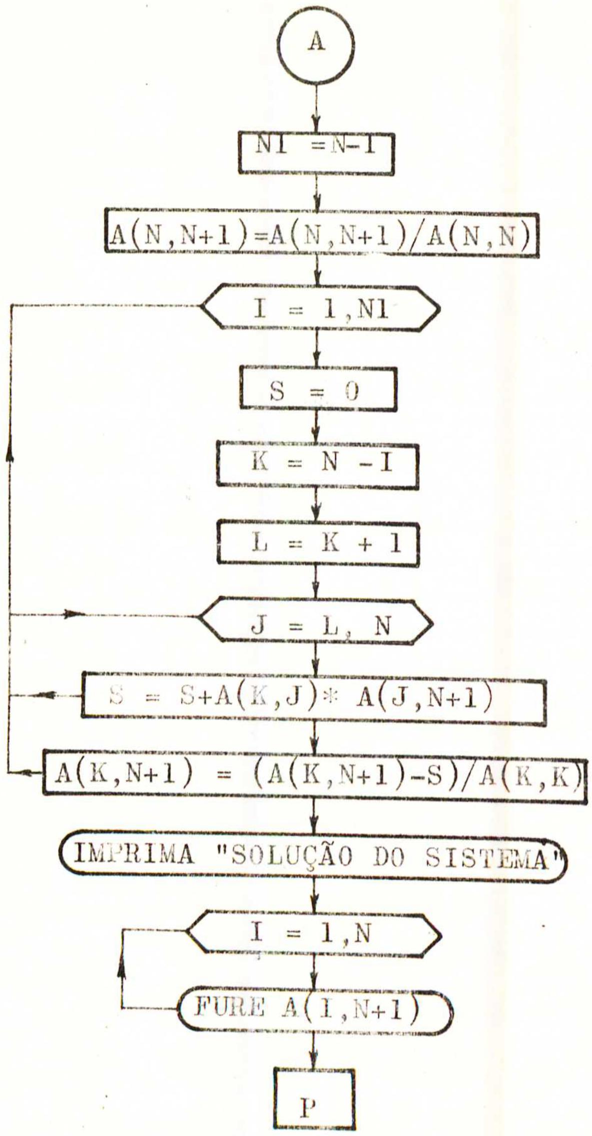
Valor de X(K)

Analogamente,

$$\boxed{X(K) = A(K,N+H) = \left[A(K,N+1) - \sum_{J=K+1}^{N} A(K,J)\,A(J,N+1)\right] \Big/ A(K,K)}$$

Diagrama de blocos

Supondo-se o sistema triangular já na memória,



Considerando-se (A) como ligação entre o final do exercício precedente e o início dêste, obtemos a resolução de um sistema de N equações a N incógnitas pelo método de eliminação sucessiva de Gauss, por triangularização.

<u>Exercício proposto</u>: Resolver novamente êste exercício utilizando sòmente as variáveis indexadas A(I), B(J). Observa-

mos que I varia de 1 a $N(N+1)/2$. Representar os têrmos conhecidos de cada equação por $B(J)$, $1 \leq J \leq N$.

Exercício 31

Calcular a inversa de uma matriz dada.

Rotina:

Seja $A = \left\{a_{ij}\right\}$ a matriz cuja inversa A^{-1} se procura. Por definição $AA^{-1} = A^{-1}A = I$. Exemplifiquemos para uma matriz de ordem 3.

$$\begin{pmatrix} A(1,1) & A(1,2) & A(1,3) \\ A(2,1) & A(2,2) & A(2,3) \\ A(3,1) & A(3,2) & A(3,3) \end{pmatrix} \begin{pmatrix} X(1,1) & X(1,2) & X(1,3) \\ X(2,1) & X(2,2) & X(2,3) \\ X(3,1) & X(3,2) & X(3,3) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Aplicando-se a definição de produto matricial, obtemos:

$$\begin{aligned} A(1,1)\ X(1,1) + A(1,2)\ X(2,1) + A(1,3)\ X(3,1) &= 1 \\ A(2,1)\ X(1,1) + A(2,2)\ X(2,1) + A(2,3)\ X(3,1) &= 0 \\ A(3,1)\ X(1,1) + A(3,2)\ X(2,1) + A(3,3)\ X(3,1) &= 0 \end{aligned}$$ ou

$$(A) \begin{pmatrix} X(1,1) \\ X(1,2) \\ X(1,3) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{aligned} A(1,1)\ X(1,2) + A(1,2)\ X(2,2) + A(1,3)\ X(3,2) &= 0 \\ A(2,1)\ X(1,2) + A(2,2)\ X(2,2) + A(2,3)\ X(3,2) &= 1 \\ A(3,1)\ X(1,2) + A(3,2)\ X(2,2) + A(3,3)\ X(3,2) &= 0 \end{aligned}$$ ou

$$(A) \begin{pmatrix} X(2,1) \\ X(2,2) \\ X(2,3) \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\begin{aligned} A(1,1)\ X(1,3) + A(1,2)\ X(2,3) + A(1,3)\ X(3,3) &= 0 \\ A(2,1)\ X(1,3) + A(2,2)\ X(2,3) + A(2,3)\ X(3,3) &= 0 \\ A(3,1)\ X(1,3) + A(3,2)\ X(2,3) + A(3,3)\ X(3,3) &= 1 \end{aligned}$$ ou

$$(A) \begin{pmatrix} X(3,1) \\ X(3,2) \\ X(3,3) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Generalizando êsse resultado, podemos dizer que a inversão de uma matriz de ordem N requer a resolução de N sistemas de N equações a N incógnitas.

A solução do sistema $A X_i = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} i$

dá os elementos da i-ésima coluna da matriz inversa A^{-1}.

No diagrama de blocos abaixo, após a leitura da matriz A, montaremos o i-ésimo sistema (i = 1, 2, ..., N) acrescentando os respectivos têrmos conhecidos. A resolução dos sistemas far-se-á pelo método de diagonalização de Gauss.

Diagrama de blocos

(v. página seguinte)

Exercício 32

Calcular a inversa de uma matriz A.

Rotina:

É a mesma do exercício precedente. Na prática, objetivando economia de tempo e maior concisão do programa, adotamos o diagrama de blocos abaixo exposto, pelo qual projetamos a resolução simultânea dos N sistemas de N equações em N incógnitas. Explicitaremos a rotina com o seguinte exemplo: Calcular a inversa de $\begin{pmatrix} 2 & 2 \\ 3 & 5 \end{pmatrix}$.

Partimos de $\begin{pmatrix} 2 & 2 \\ 3 & 5 \end{pmatrix}$ e $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

1) Dividimos a 1ª linha por A(1,1), i.e., 2

$$\begin{pmatrix} 1 & 1 \\ 3 & 5 \end{pmatrix} \qquad \begin{pmatrix} 0,5 & 0 \\ 0 & 1 \end{pmatrix}$$

2) Multiplicamos a primeira linha por A(2,1), i.e., 3 e subtraímo-la da segunda

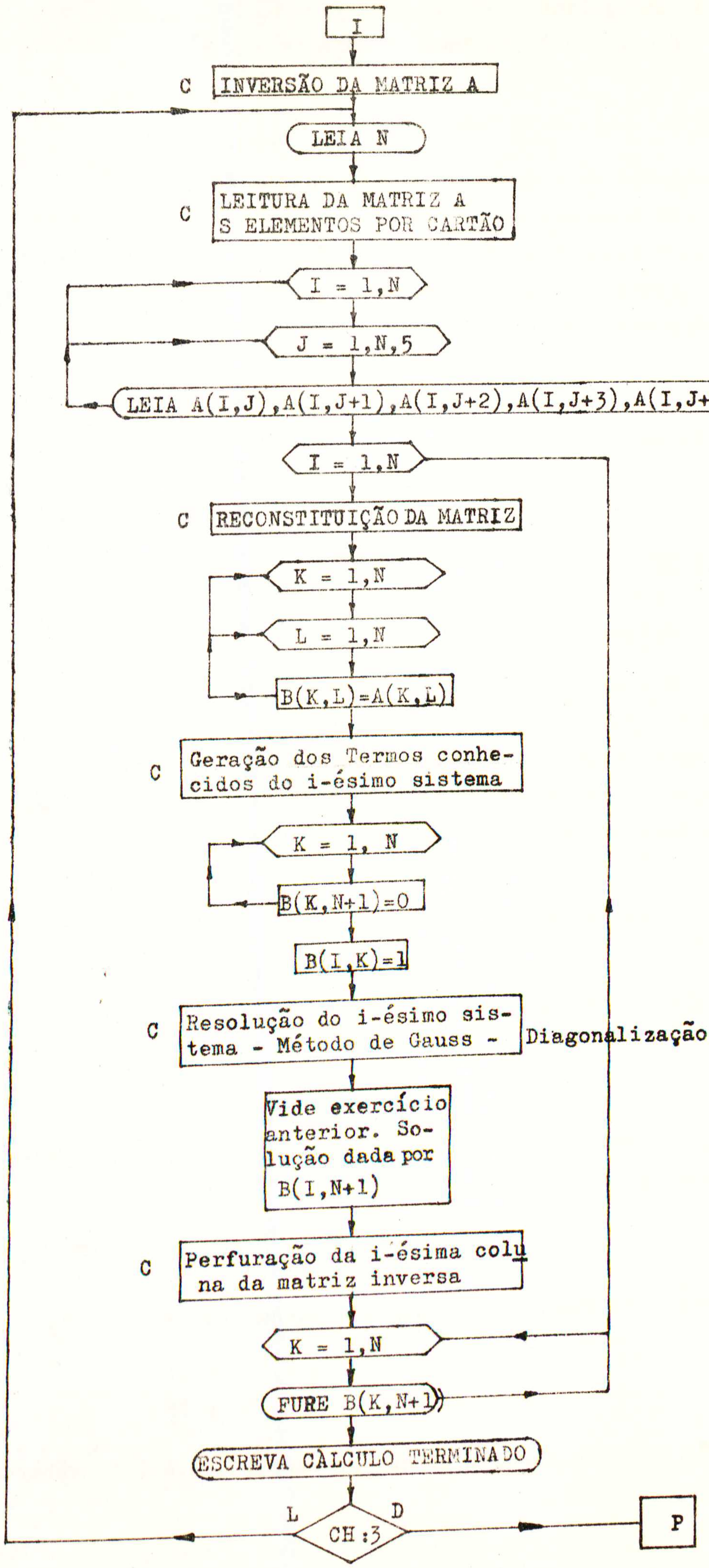
$$\begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix} \qquad \begin{pmatrix} 0,5 & 0 \\ -1,5 & 1 \end{pmatrix}$$

3) Dividimos os elementos da segunda linha por A(2,2), i.e., 2

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 0,5 & 0 \\ -0,75 & 0,5 \end{pmatrix}$$

4) Multiplicamos a segunda linha por A(1,2), i.e., 1, subtraímo-la da primeira

Diagrama de blocos do Exercício 31



$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1{,}25 & -0{,}5 \\ -0{,}75 & 0{,}5 \end{pmatrix}$$

Resulta finalmente

$$\begin{pmatrix} 1{,}25 & -0{,}5 \\ -0{,}75 & 0{,}5 \end{pmatrix}$$

como matriz inversa.

Submetemos os elementos da matriz I às mesmas transformações que aplicamos à matriz A, transformações estas equivalentes às mesmas utilizadas para resolução de um sistema de equações pelo método da diagonalização de Gauss. Conforme vimos em exercício anterior, após essas transformações, a solução do sistema fica dada pelos próprios têrmos conhecidos.

No diagrama de blocos a solução está representada pela matriz I, cujas colunas são os têrmos conhecidos dos sistemas montados com a matriz A. Todavia a matriz I está superposta à matriz A, resultando daí estar a solução, após as transformações, representada pela própria matriz A. Essa superposição está considerada no diagrama de blocos:

1º) Na redução à unidade dos elementos da diagonal principal

2º) na redução a zero dos elementos de cada coluna, exceção feita ao elemento que também pertence à diagonal principal.

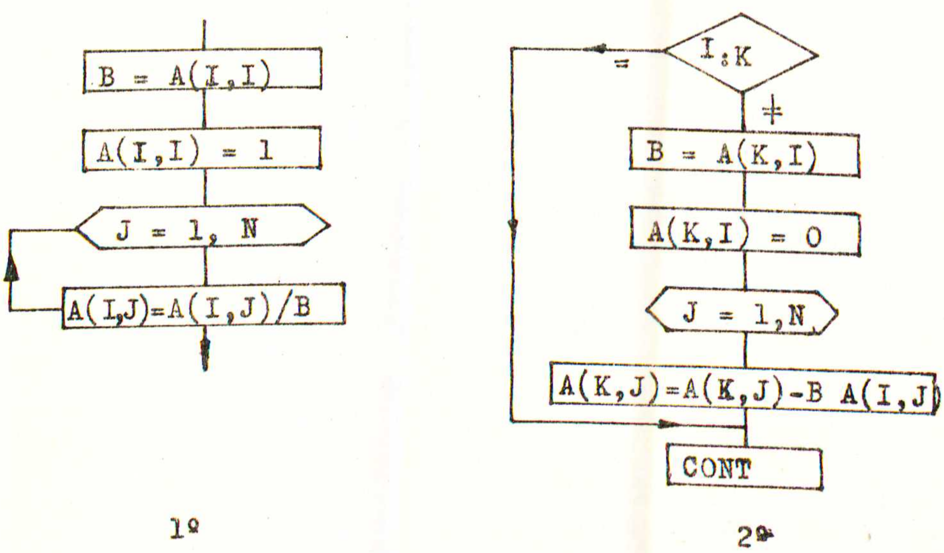
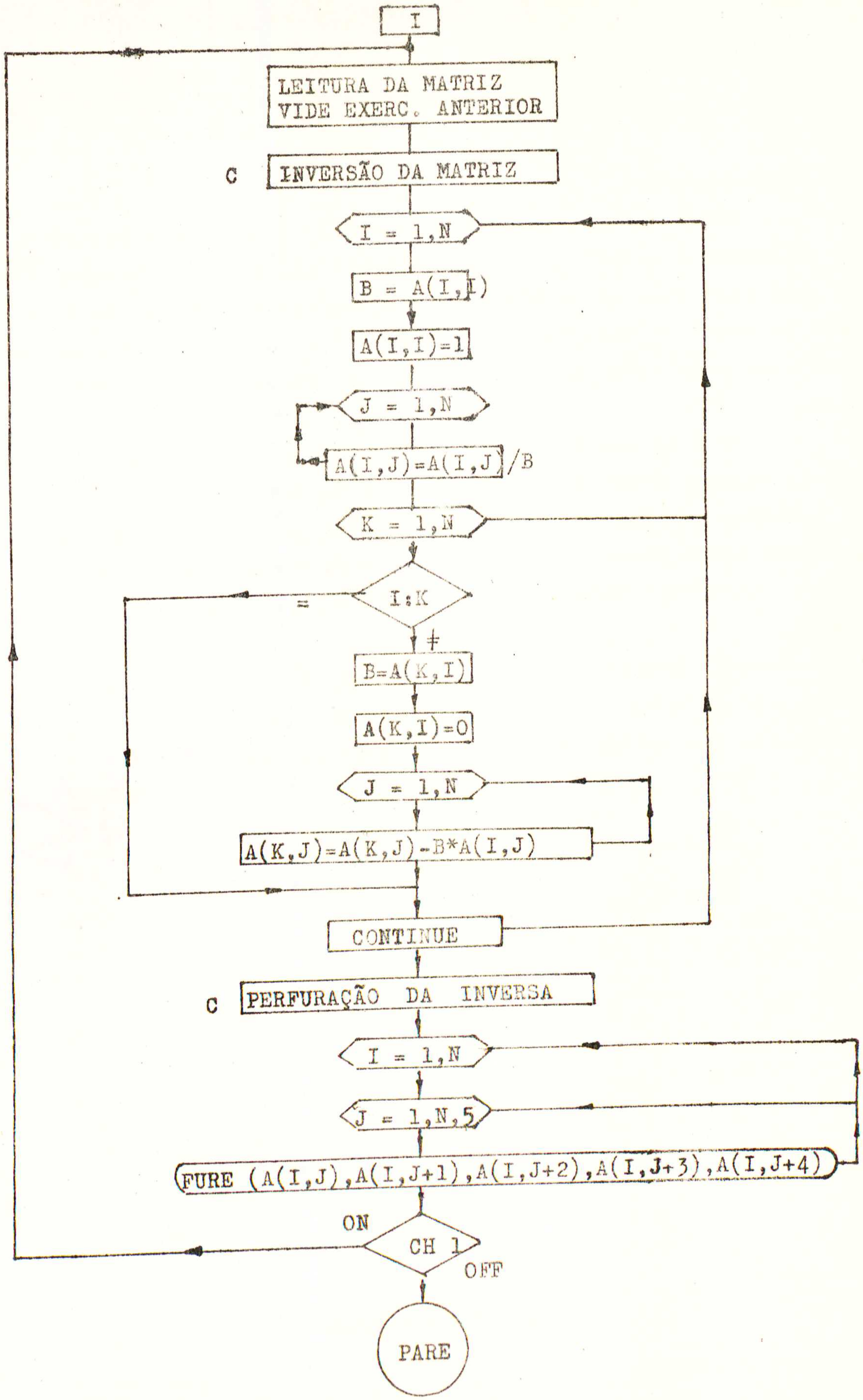


Diagrama de blocos

(v. página seguinte)


I
LEITURA DA MATRIZ
VIDE EXERC. ANTERIOR
C
INVERSÃO DA MATRIZ
I = 1,N
B = A(I,I)
A(I,I)=1
J = 1,N
A(I,J)=A(I,J)/B
K = 1,N
I:K
=
≠
B=A(K,I)
A(K,I)=0
J = 1,N
A(K,J)=A(K,J)-B*A(I,J)
CONTINUE
C
PERFURAÇÃO DA INVERSA
I = 1,N
J = 1,N,5
FURE (A(I,J),A(I,J+1),A(I,J+2),A(I,J+3),A(I,J+4))
ON
CH 1
OFF
PARE