

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

DEPARTAMENTO DE ENGENHARIA QUÍMICA DA

ESCOLA DE QUÍMICA

FORTRAN PARA ENGENHEIROS QUÍMICOS

ABRAHAM ZAKON

## P R E F Á C I O

FORTTRAN PARA ENGENHEIROS QUÍMICOS destina-se a oferecer aos estudantes uma coleção de problemas de fácil compreensão, selecionados dentre os diversos ramos da Engenharia Química.

O Capítulo 1 - Processamento de Dados na Engenharia Química - é meramente introdutório. O Capítulo 2 - Forttran na Engenharia Química - destina-se a proporcionar uma iniciação sólida e simples no uso da linguagem, oferecendo os rudimentos básicos da construção de fluxogramas, redação de programas em folhas de codificação e as respectivas listagens.

O Capítulo 3 - Quantificação de Fenômenos e Materiais - apresenta alguns programas mais elaborados, com os recursos de listagem comumente encontrados na bibliografia.

Alguns dos exemplos contêm os respectivos fluxogramas; outros não os contêm propositadamente. Em muitos casos o leitor não os encontra na literatura, sendo induzido a construí-los para efetuar a adaptação dos mesmos para os nossos computadores ou sistemas.

A presente edição não contém exercícios propostos, os quais poderão ser encontrados nas bibliografias citadas. Os programas foram redigidos de modo a resolverem problemas com as noções mais simples de FORTRAN, e dispostos em ordem de dificuldade crescente. Cada leitor poderá alterar, sofisticar ou adaptar cada um dos exemplos.

Abraham Zakon  
EQUFRJ - 1977

## ÍNDICE GERAL

CAPÍTULO 1 - PROCESSAMENTO DE DADOS NA ENGENHARIA QUÍMICA	pág.
1. Tipos de Dados ou Informações Disponíveis	1
2. Tipos de Quantidades e Variáveis	1
3. Objetivos do Processamento de Dados na Engenharia Química	1
4. Tipos de Computadores Empregados	2
5. Entes Matemáticos Fundamentais	3
6. Modelos Analógicos de Simbolização e Raciocínio	4
7. Dinâmica de Processos	5
8. Hierarquia do Uso de Computadores	6
9. Seleção de Computadores	7
10. Conclusões	8
11. Bibliografia	8
CAPÍTULO 2 - FORTRAN NA ENGENHARIA QUÍMICA	
1. Elaboração do Algoritmo, do Fluxograma e do Programa	9
2. Problemas de Engenharia Química	9
3. Instruções ou Declarações FORTRAN	9
4. Linguagem de Fluxogramas	10
5. Linguagem FORTRAN	11
6. Redação de Programas Pequenos	15
7. Exemplos Simples	16
7.1 Tensão na Corda de uma Roldana	17
7.2 Diferença de Temperaturas Média Logarítmica	19
7.3 Desintegração Radioativa	20
8. Formatação de Entrada e Saída	22
9. Outros Exemplos Simples	45
9.1 Tempo de Enchimento de um Tanque Cilíndrico	45
9.2 Medidor Venturi	47
9.3 Dimensionamento de Tubos - Perda de Carga Desprezível	49
9.4 Números Complexos	54
10. Bibliografia	
CAPÍTULO 3 - QUANTIFICAÇÃO DE FENÔMENOS E MATERIAIS	
1. Transferência de Calor em Parede Plana Composta	60
2. Funções e Representação de Dados - Cálculo do pH	66
3. Erros Numéricos - Pressão de um Gás	72
4. Frações de Solute em Extração Contracorrente	77
5. Umidade de uma substância	80
6. Média e Desvio Padrão	83
7. Obtenção de Gráfico com o emprego do Comando DO	85
8. Sistema Oscilador	93
9. "IF" Lógico e Transferência Controlada - Sistema Oscilador	98
10. Cálculo da Perda de Carga em Hidráulica	100
11. Rotinas e Subrotinas FORTRAN - Fórmulas Empíricas de Compostos	106
12. Dimensionamento de Tubos com Perda de Carga	110
13. Ajustamento de Curvas	116
14. Cinética de Reações Químicas	121
15. Bibliografia.	130

PROCESSAMENTO DE DADOS NA  
ENGENHARIA QUÍMICA

1. TIPOS DE DADOS OU INFORMAÇÕES DISPONÍVEIS.

Usualmente, os engenheiros químicos utilizam dados:

- A - Técnicos: de projeto, operação, simulação, etc.
- B - Econômicos: de contabilidade, viabilidade econômica, otimização, etc.
- C - Estatísticos: médias, percentuais, desvios, etc.

2. TIPOS DE QUANTIDADES E DE VARIÁVEIS.

De um modo geral, a Matemática nos oferece quantidades escalares e vetoriais. E a Estatística nos apresenta elementos quantitativos (variáveis) e qualitativos (atributos). Ambas nos permitem avaliar ou descrever um processo ou sistema e tomar decisões a respeito dos mesmos. Mas, para que compreendamos ainda melhor as alterações que ocorrem nos mesmos, vale a pena definir os diferentes tipos de variáveis:

- A - Aleatórias: só assumem valores após a realização de alguma etapa do processo.
- B - Discretas: (ou descontínuas) - assumem valores bem determinados, geralmente números inteiros.
- C - Contínuas: assumem todos os valores dentro de um intervalo.

3. OBJETIVOS DO PROCESSAMENTO DE DADOS NA ENGENHARIA QUÍMICA.

Destina-se, comumente, a obter informações de:

- A - Gerais: listagens, tabelas, gráficos, etc.
- B - Economia.
- C - Gerência.
- D - Matemática.
- E - Pesquisa e Previsões.
- F - Simulação, otimização e projeto de equipamentos, tubulações, unidades industriais, processo, utilidades, controle de poluição, etc.
- G - Análise de Processos.

#### 4. TIPOS DE COMPUTADORES EMPREGADOS.

Os computadores modernos podem ser classificados em digitais, analógicos e híbridos.

A - Digital: executa operações aritméticas com alto grau de precisão e pode ser equipado com uma grande capacidade de memória que permite estocar e operar um número quase ilimitado de informações. Além disto é usado para obter decisões lógicas, pela comparação de dois números (informações), e decidindo se um é igual, menor que ou maior que o outro.

A programação num computador digital é uma tarefa de muita precisão, onde são fornecidas instruções detalhadas através de linguagens especializadas.

O uso de computadores digitais, através da linguagem FORTRAN é o objetivo principal da disciplina COMPUTAÇÃO

B - Analógico: não possui a capacidade para memória e para um grande número de operações.

A programação de um analógico não requer conhecimentos de uma linguagem altamente especializada, conforme o digital. Os detalhes da programação são geralmente semelhantes aos clássicos métodos de solução com os quais o engenheiro está familiarizado. Os computadores analógicos encontram sua maior aplicação na solução de equações diferenciais, tal como ocorre em análise dinâmica de sistemas em processamento, nos quais a quantidade de operações algébricas e lógicas requeridas é limitada.

No analógico as quantidades físicas são usadas para representar as variáveis do problema. Os números são manipulados através da dimensão de uma variável tal como o Volt ou Psi (pressão). Por exemplo, na régua de cálculo (que é um dos mais simples computadores analógicos), os números são representados por comprimentos proporcionais aos logaritmos dos números. Os comprimentos são, então, manipulados para obtermos os resultados desejados: multiplicação, divisão, e assim por diante. Nos computadores analógicos eletrônicos as voltagens são usadas para representar variáveis de problemas, tais como, velocidade, temperatura e aceleração. Se o circuito de um analógico possuir diversos ramais, diversas operações podem ser executadas em paralelo ou simultaneamente. A precisão é limitada pela precisão das medições do processo e pela conversão destas para o meio físico com os quais o computador opera.

C - Híbrido: é a combinação de um computador digital e de um analógico, no qual a solução de um problema é compartilhada pelos dois, incorporando-se ao sistema as vantagens de ambos.

## 5. ENTES MATEMÁTICOS FUNDAMENTAIS.

- A - ESCALAR: é uma quantidade física caracterizada por uma INTENSIDADE, tal como massa (ou biomassa), tempo e temperatura. É representada por um número (independentemente do sistema de coordenadas). Pode ser, também, uma quantidade econômica ou estatística.
- B - VETOR: é uma quantidade física caracterizada por INTENSIDADE e DIREÇÃO, tal como deslocamento, força, momento linear ou velocidade.
- C - MATRIZ: é um conjunto de escalares dispostos num quadro retangular que nos fornece uma representação matemática útil: é simplesmente um recurso matemático. Uma tabela é uma matriz.
- D - PROPRIEDADE: é uma quantidade mensurável de um sistema, isto é, um valor ou uma série de valores dentro de um sistema, descrevendo o estado do mesmo.
- E - PROPRIEDADE EXTENSIVA: depende do tamanho (extensão) do sistema ou quantidade de matéria envolvida. Exemplos: energia interna e volume.
- F - PROPRIEDADE INTENSIVA: não depende da quantidade de material envolvido e varia de ponto a ponto. Exemplos: temperatura e pressão, volume específico (\*), energia por mol (\*).
- (\* ) Em muitas aplicações é conveniente escrever equações de balanço para quantidades de material ou energia por unidade de massa, ou de volume ou por mol, que no caso podem representar o sistema em consideração.
- G - TAXA: ("RATE"), (VELOCIDADE) - é a variação de uma propriedade qualquer em relação a uma variação ou intervalo de tempo.
- H - NÚMEROS ADIMENSIONAIS: são resultantes do agrupamento de variáveis dimensionais, cujos significados físicos podem ser diferentes ou de mesma natureza. Geralmente, aparecem sob a forma de razão entre variáveis, gerando números "puros". Permitem caracterizar o comportamento de um processo de modo simplificado, onde os números representam diferentes estados do sistema considerado. Exemplos: Número de Reynolds e Densidade Relativa.

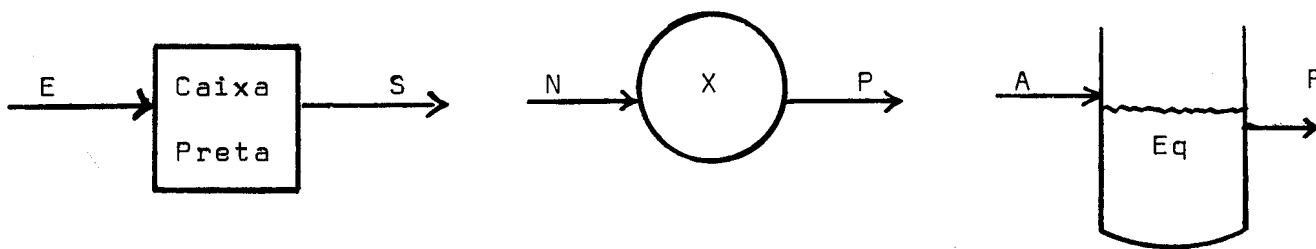
6. MODÉLOS ANALÓGICOS DE SIMBOLIZAÇÃO E RACIOCÍNIO.

Podemos considerar três formas análogas de sistemas unitários:

MATEMÁTICA

MICROBIOLÓGICA

INDUSTRIAL



E - valor de entrada

N - nutriente

A - alimentação

CP - onde ocorre uma transformação

X - célula microbiana

Eq - equipamento

S - valor de saída

P - produto excretado

P - produto

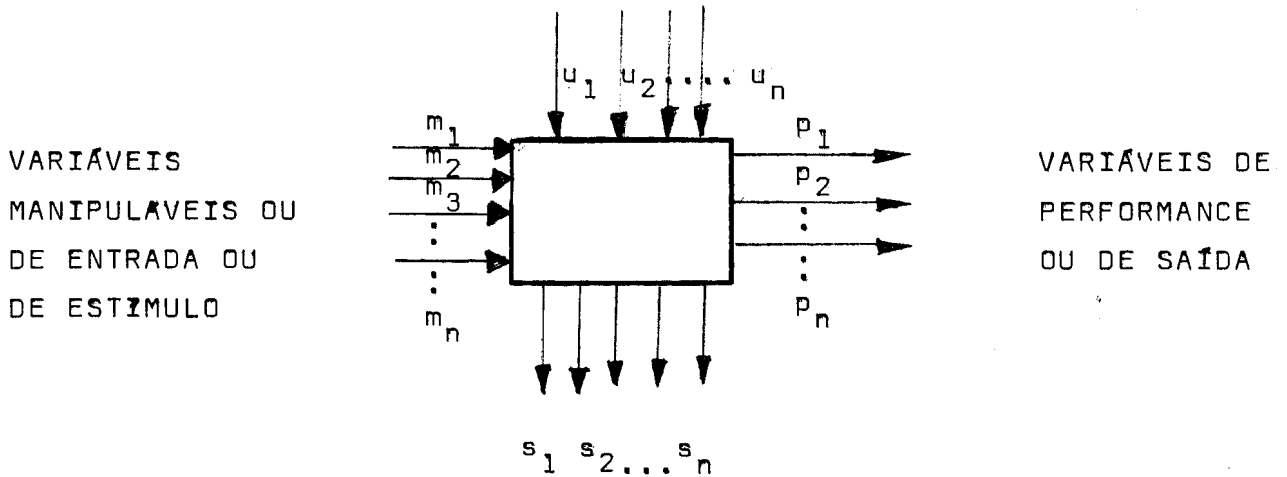
Podemos observar que nos três sistemas ocorre, dentro dos mesmos, a transformação da propriedade absorvida. E que é possível distinguir três etapas em cada processamento: captação, transformação e expulsão de propriedades. Com estes modelos podemos gerar equações diferenciais lineares e sistemas de equações lineares. Teremos modelos matemáticos que nos serão satisfatórios, até que surja outro melhor que os suplante e os substitua.

7. DINÂMICA DE PROCESSOS.

É o resultado da interação de variáveis, perturbações e sistemas de controle biológicos ou físico-químicos. Deveremos considerar basicamente:

A - Uma operação ou etapa unitária qualquer:

VARIÁVEIS DE PERTURBAÇÃO OU CARGA





## B - Variáveis de Operação e Projeto:

FÍSICAS	QUÍMICAS	BIOQUÍMICAS	BIOLÓGICAS
Temperatura	pH da solução	Nível de DNA	Contaminação
Pressão do Vaso	Potencial Redoxi	Nível de RNA	Mutação
Potência de Entrada	O <sub>2</sub> dissolvido	Proteínas to	
Velocidade de agi- tação	O <sub>2</sub> dissolvido	tais	
Taxas de Escoamento de Gás	Nível de carbohi- dratos	Proteínas Espe- cíficas (Ativi- dade Enzimáti- ca)	
Taxas de Alimenta- ção de líquidos	Nível de Nitro- gênio		
Viscosidade	Nível de Ions Mi- nerais		
Volume do líquido			

## C - Aspecto Geral de um Balanço:

Térmo de ACUMULAÇÃO	Térmo de TRANSPORTE	Térmo de TRANSPORTE	Térmo de GERAÇÃO	Térmo de CONSUMO
resultante = no Volume do sistema	resultante - Na Entrada através a Superfície do Sistema	resultante + Na Saída através a Superfície do Sistema	resultante - No Volume do Sistema	resultante No Volume do Sistema

Quando consideramos um só componente cuja composição não varia no sistema podemos excluir o termo de consumo e o de geração.

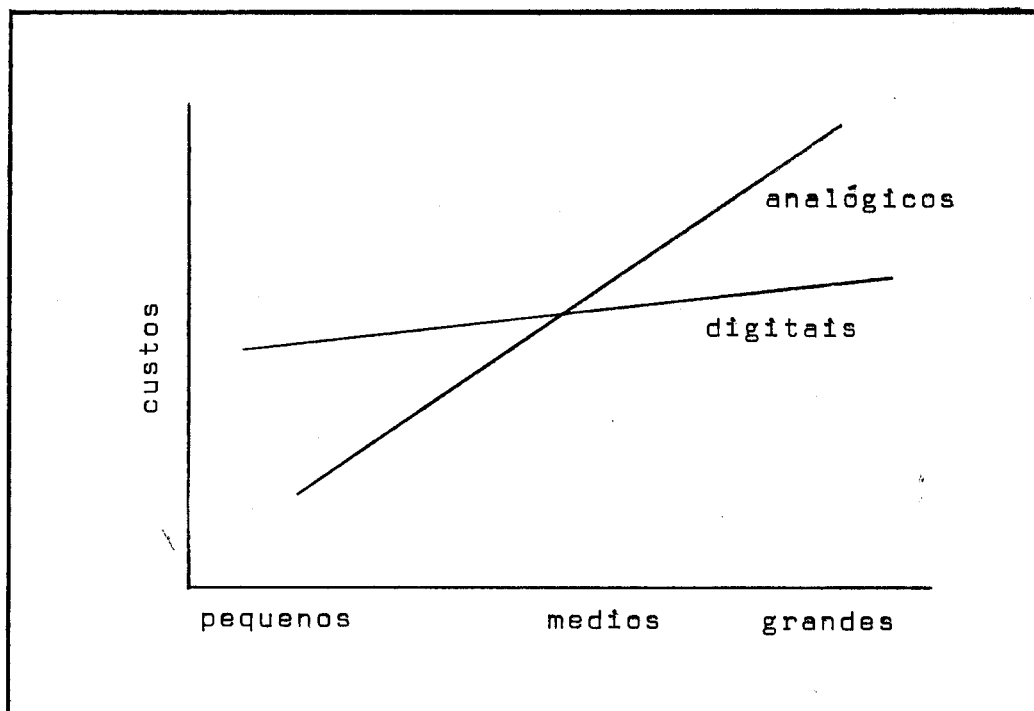
8. HIERARQUIA DO USO DE COMPUTADORES - podemos identificar cinco níveis:
1. CONTRÔLE DA OPERAÇÃO UNITÁRIA - Ex.: permutador, reator, estufa.
  2. CONTRÔLE DA UNIDADE DE PROCESSO - Ex.: reformação, que inclui reator catalítico, fracionador de produto e permutadores.
  3. CONTRÔLE DE FÁBRICAS - Ex.: FAFER, FAVOR, REDUC, etc.
  4. DEPARTAMENTO INDUSTRIAL - Ex.: DIRIND
  5. CORPORAÇÃO - Ex.: PETROBRÁS, FOSTER WHEELER, etc.

Evidentemente, não estamos considerando apenas o uso em controle mas também todos os demais objetivos anteriormente mencionados.

## 9. SELEÇÃO DE COMPUTADORES.

### A - Custos:

A escolha do computador é governada pelo nível de controle a ser executado e o respectivo custo de execução. A figura abaixo ilustra os custos relativos dos sistemas de controle, através de computadores analógicos e digitais.



### B - Vantagens:

#### Computadores Analógicos

Menor Custo para Sistemas Menores  
Mais flexível em Sistemas Menores  
Técnicas empregadas são mais familiares ao pessoal de instrumentação na fábrica  
Procedimentos de resolução dos problemas são mais familiares  
Computação paralela: um defeito em um componente não prejudica o sistema inteiro.  
Mais imune ao barulho  
Entrosamento técnico entre a Instrumentação e as medições analógicas é o menos complexo.

#### Computadores Digitais

Menor Custo para Sistemas Maiores  
Mais flexível em Sistemas Maiores  
Técnicas empregadas mais familiares ao engenheiros de processo  
Maior precisão disponível  
Capaz de operar procedimentos complexos como a otimização  
Armazena dados sem distorcê-los  
Custos de Expansão menores.

## 10. CONCLUSÕES.

Verifica-se, portanto, que o Engenheiro Químico lida com Sistemas Industriais e de Pesquisa, Biológicos e não-Biológicos, Macroscópicos e Microscópicos, Quantitativos e Qualitativos. Os recursos empregados na resolução de seus problemas são decorrentes de Fundamentos Matemáticos, Físicos (Termodinâmicos) e Químicos (Cinéticos).

Deseja-se permitir que o Engenheiro Químico manipule os dados, variáveis e computadores de acordo com objetivos industriais. Ou seja, o computador é um equipamento idealizado para proporcionar a economia de recursos num sistema qualquer e a operação econômica deve ser prioritária em nossos trabalhos.

Um Engenheiro Químico poderá ser um excelente programador de computadores, mesmo que ele não se dedique exclusivamente à programação e análise de sistemas, porque estes equipamentos são ferramentas de trabalho indispensáveis na Engenharia Química.

## 11. BIBLIOGRAFIA.

1. Hsu, H.P., "Análise Vetorial" Livros Técnicos e Científicos Editora Ltda., RJ (1972).
2. Nóbrega et alii, "Introdução aos Processos de Transporte" COPPE-UFRJ, RJ (1973).
3. Perry, J.H. (Ed.), "Chemical Engineers' Handbook/Fourth Edition" Mc Graw - Hill & Kogakusha, Tokyo (1963).
4. Perry and Chilton, "Chemical Engineers' Handbook/Fifth Edition" McGraw - Hill & Kogakusha, Tokyo (1973).
5. Nyiri, L.K., "Application of Computers in Biochemical Engineering" Adv. Bioch. Eng., 2 - Springer Verlag, Berlin (1972).
6. Hydrocarbon Processing, February 1968, 47, nº 2.
7. Chemical Engineering, May 6, 1968.
8. Franks, R.G.E., "Mathematical Modeling in Chemical Engineering" John Wiley & Sons - New York (1967)
9. Henley & Rosen, "Material and Energy Balance Computations" John Wiley & Sons - New York (1969)
10. Di Stefanò III e outros, "Feedback and Control Systems" Schaum's Outline Series, McGraw-Hill - NY (1967)

## FORTRAN NA ENGENHARIA QUÍMICA

### 1. ELABORAÇÃO DO ALGORÍTIMO, FLUXOGRAMA E PROGRAMA.

A elaboração de um programa de computador, para resolver qualquer tipo de problema, exige que a pessoa organize o seu raciocínio. E para isto ela deve conhecer bem o algoritmo do problema e a partir d'êste elaborar um fluxograma, o qual permitirá redigir facilmente o programa mencionado.

Um algoritmo é uma lista de instruções para a execução, passo a passo, de algum processo. Uma técnica de análise química constitui um algoritmo. Um método de resolução de algum problema de engenharia constitui um algoritmo.

Sabemos que Atividade é a execução de uma tarefa, operação ou um trabalho. Gasta TEMPO e RECURSOS (mão-de-obra, material, dinheiro). Evento é o momento de início ou fim de uma atividade, não gasta tempo nem recursos. É um marco dentro da execução do projeto. Um diagrama é um conjunto de atividades e de eventos, no qual vemos as tarefas na ordem em que se sucedem, com a nítida noção de dependência entre elas. Portanto, um FLUXOGRAMA é um diagrama para a representação de um algoritmo.

### 2. PROBLEMAS DE ENGENHARIA QUÍMICA.

Podemos classificá-los em três níveis.

- 1º Nível - modelização e quantificação dos fenômenos.
- 2º Nível - projeto, dimensionamento e seleção de equipamento.
- 3º Nível - engenharia de processos ou de unidades industriais.

### 3. INSTRUÇÕES OU DECLARAÇÕES FORTRAN

São declarações formais que:

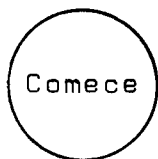
- \* definem operações aritméticas a serem executadas pelo computador,
- \* fornecem informações para o controle do computador durante a execução do programa,
- \* descrevem as operações de entrada e saída necessárias à alimentação de dados no computador e a conseqüente apresentação dos resultados,
- \* especificam fatos adicionais necessários ao programa compilador - por exemplo, as dimensões das variáveis subscriptas que aparecem no programa.

#### 4. LINGUAGEM DE FLUXOGRAMAS.

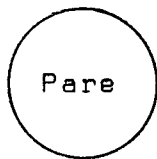
Os nossos fluxogramas retratarão os caminhos, alternativas e condições que o computador oferece à resolução do problema de Engenharia Química. Facilitarão, portanto, ao programador e ao consultor localizar os possíveis erros de programação e lógica, quando houver necessidade de correção ou alteração dos mesmos.

Forsythe et alii<sup>1</sup> e Pacitti apresentam duas linguagens diferentes para a composição dos fluxogramas. Eis, em resumo, os componentes das mesmas:

FORSYTHE ET ALLII:

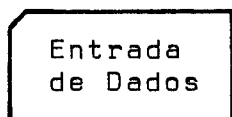


Comece

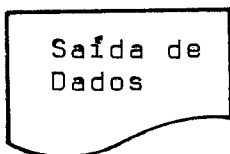


Pare

Botões de Partida e de Parada

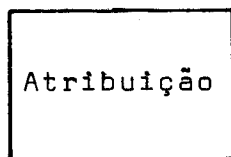


Entrada de Dados

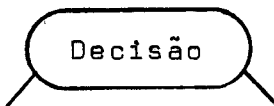


Saída de Dados

Caixas de Entrada e de Saída



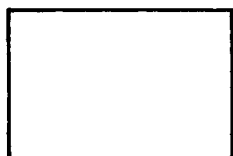
Atribuição



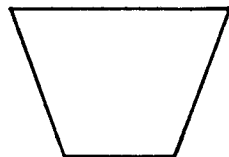
Decisão

Caixas de Atribuição e de Decisão

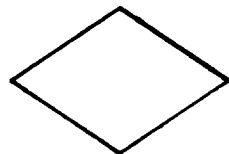
PACITTI:



Indica o processamento de algum cálculo que deve ser escrito dentro do retângulo, onde uma variável vai assumir o valor de uma expressão equivalente.



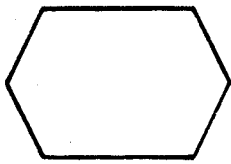
Indica entrada e saída de dados via seus respectivos equipamentos.



Indica decisão a ser tomada.



Indica a conexão entre dois pontos em um programa, onde uma linha não pode ser traçada, ou seja, continuação em outra página, e n é um número

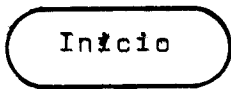


Tem dois significados:

- a) Significa que está sendo chamada um subprograma da biblioteca.
- b) Significa a execução repetida, sobre o controle, de um certo número de declarações.

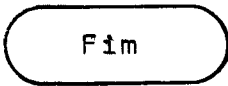


Indica direção do escoamento.



Início

Indica início ou fim de um programa.



Fim



Manipulação com fita magnética.

Pacitti também comenta que não existe uma padronização geral dos blocos significativos, afirmando que cada livro, cada manual, aconselha simbologia toda própria, ou parcialmente coincidente entre si.

## 5. LINGUAGEM FORTRAN

Esta envolve caracteres alfanuméricos, operadores aritméticos e delimitadores, e do ponto de vista matemático, constantes (inteiras ou reais) e variáveis (inteiras ou reais).

Considerando a bibliografia existente, iremos encontrar, por exemplo, em livros e revistas, diversas variáveis na Engenharia Química, respeitando as regras de programação FORTRAN. Em consequência disto, algumas palavras são truncadas para criar nomes esquisitos, ou então, coloca-se um " X " no início dos caracteres que dominam a variável, para tornar uma variável inteira em uma variável real. Por outro lado, muitas das variáveis são resultados de abreviações de nomes originalmente escritos em inglês.

Vale a pena observar aqui neste ponto que não iremos descrever neste texto um curso de FORTRAN, mas apenas algumas de suas aplicações na Engenharia Química.

Na página seguinte apresentamos diversos exemplos de variáveis.



O uso mais poderoso da declaração aritmética FØRTRAN consiste na execução de séries de operações aritméticas, envolvendo variáveis e constantes, cujos resultados são conferidos a um nome específico de uma variável. Qualquer expressão algébrica normal pode ser escrita sob a forma de uma declaração aritmética FØRTRAN, tão longa quanto seja, de modo a conter à esquerda do sinal de igualdade somente uma variável. Alguns exemplos de expressões algébricas e declarações FØRTRAN equivalentes são:

1. A equação de estado de um gás ideal é dada por  $PV = nRT$ . Quando resolvida para  $P$ , obtemos  $P = nRT/V$ . Uma declaração aritmética seria:

$$P = (XMOL*0.0821*TEMP) / VOL$$

2. A equação de estado de Van der Waals para um gás é:

$$(P + (an^2/V^2))(V-nb) = nRT$$

Quando resolvida para  $P$ , obtemos:

$$P = \frac{nRT}{V - nb} - \frac{an^2}{V^2}$$

Uma declaração aritmética FØRTRAN seria:

$$PRESS = ((FN*R*T)/(V - FN*B)) - ((A*FN**2)/(V** 2))$$

3. A energia cinética pode ser expressa como  $KE = \frac{1}{2}mV^2$ . Abaixo temos exemplos cabíveis de declarações aritméticas FØRTRAN.

$$EKIN = (XM/2.0)* (V**2)$$

$$\text{ou } EKIN = .5*XM*V*V$$

$$\text{ou } EKIN = (1./2.)* XM* (V**2)$$

4. A Lei de Dalton das expressões parciais envolvendo três componentes pode ser escrita sob a forma  $P_t = P_a + P_b + P_c$ . Teríamos então:

$$PT = PA + PB + PC$$

5. A expressão para a energia de um eletrón em um átomo, de acordo com o modelo de Bohr é:

$$E = \frac{-2\pi^2 me^4}{n^2 h^2}$$



Em FORTRAN teríamos:

$$EN = [(-2.)*(3.14**2)*XM*(E**4)]/[(FN**2)*(H**2)]$$

6. Em Termodinâmica, uma expressão para variações de entalpia, em determinadas condições é:

$$H = E + P V$$

Uma declaração FORTRAN poderia ser:

$$DELH = DELA + P*DELV$$

7. A massa específica de uma substância, numa dada temperatura, é definida como massa por volume. Podemos expressá-la assim:

$$D = FAMSS/VOL$$

8. O algoritmo para a solução de uma equação quadrática é dado por:

$$X = \frac{-b \pm (b^2 - 4ac)^{1/2}}{2a}$$

Uma declaração FORTRAN equivalente não é possível porque o "mais ou menos" não pode ser expressado. Entretanto, duas declarações FORTRAN podem ser escritas para representar ambos os casos. Por exemplo:

$$\begin{aligned} X &= [(-B) + (B**2 - 4.*A*C)**.5]/(2.*A) \\ e \quad X &= [(-B) - (B**2 - 4.*A*C)**.5]/(2.*A) \end{aligned}$$

9. O desvio padrão pode ser calculado por:

$$S = \left[ \frac{\sum x^2 - (\sum x)^2 / n}{n - 1} \right]^{1/2}$$

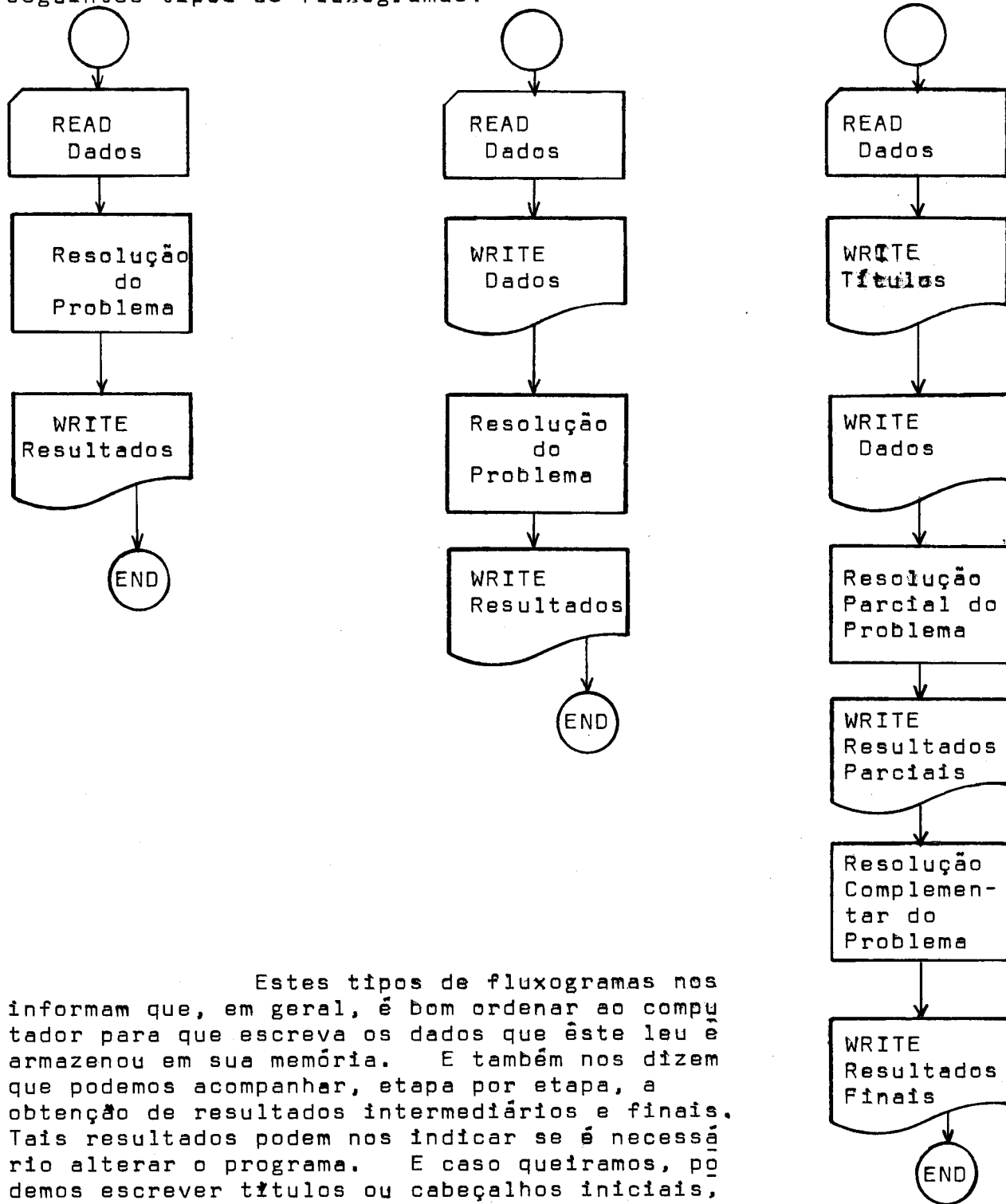
Uma declaração FORTRAN análoga seria:

$$S = [((SUMX2) - (SUMX**2)/FN)/(FN-1.0)]**.5$$

Os nomes de variáveis SUMX2 e SUMX são usadas para  $\sum x^2$  e  $\sum x$  e devem ser calculadas antes de aparecer nesta declaração.

## 6. REDAÇÃO DE PROGRAMAS PEQUENOS.

Os problemas mais simples podem ser resolvidos pelos seguintes tipos de fluxogramas:



Estes tipos de fluxogramas nos informam que, em geral, é bom ordenar ao computador para que escreva os dados que este leu e armazenou em sua memória. E também nos dizem que podemos acompanhar, etapa por etapa, a obtenção de resultados intermediários e finais. Tais resultados podem nos indicar se é necessário alterar o programa. E caso queiramos, podemos escrever títulos ou cabeçalhos iniciais, intermediários e finais. Apresentamos a seguir diversos exemplos ilustrativos.

## 7. EXEMPLOS SIMPLES

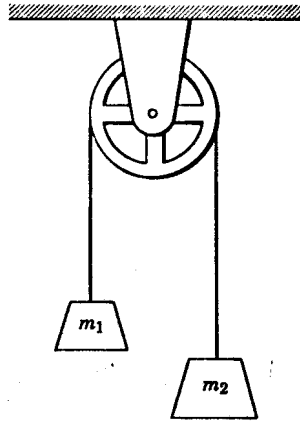
Os problemas aqui apresentados foram escolhidos devido à simplicidade dos mesmos, que permitem ao estudante organizar seu raciocínio de modo gradativo.

O aprendizado da linguagem FORTRAN requer uma prática constante da mesma. Ao final de um certo tempo, o estudante torna-se habilitado a adaptar programas e subrotinas da bibliografia existente. Com esta base, torna-se mais fácil criar novos programas. Daí para o emprego do SSP e do IMSL, com vistas à resolução de projetos de engenharia química, o uso da linguagem assume uma conotação objetiva, prática e rentável.

Após esta série de exemplos simples, serão apresentados outros mais elaborados, pertinentes à quantificação de fenômenos e materiais. Alguns deles abrangem subrotinas e funções embutidas entre si.

A etapa posterior será a de projeto (que ainda não consta deste texto) e que compreende o verdadeiro objetivo desta exposição: permitir que os engenheiros químicos utilizem a linguagem FORTRAN para resolver seus projetos.

PROBLEMA 7.1 TENSÃO NA CORDA DE UMA ROLDANA



A figura acima descreve duas massas desiguais ligadas por uma corda, de peso desprezível, passando por uma roldana, cujo atrito e massa são também desprezíveis. A fórmula para a tensão na corda é:

$$T = \frac{2 m_1 m_2}{m_1 + m_2} \cdot g$$

onde  $m_1, m_2$  = massa, slugs  
 $g$  = constante gravitacional,  $32.2 \text{ ft}^2/\text{sec}^2$   
 $T$  = tensão, libra força

Desejamos calcular o valor da tensão e escrever os valores desta e das massas lidas nos cartões.

PROGRAMA

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34			
C																																				
C																																				
C																																				
C																																				



C  
C  
C  
C  
C  
C

CALCULO DA TENSÃO NA CORDA DE  
UMA ROLDANA QUE SUPORTA DUAS  
MASSAS DESIGUAIS

```
REAL M1,M2  
READ(8,1)M1,M2  
1 FORMAT(2F10,0)  
T=64,4*M1*M2/(M1+M2)  
WRITE(5,2)M1,M2,T  
2 FORMAT(1H ,3(F13,5,2X))  
CALL EXIT  
END
```

2,00000

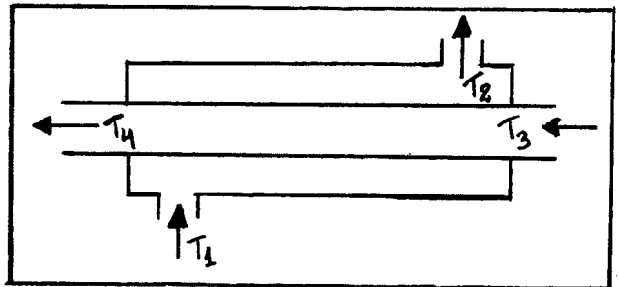
1,00000

42,93333

## 7.24 DIFERENÇA DE TEMPERATURAS MÉDIA LOGARÍTMICA

Escrever o fluxograma FORTRAN para o cálculo da Diferença de Temperaturas Média Logarítmica para o trocador de calor de cano duplo esquematizado abaixo. A DTML é a força motriz da transferência de calor entre as duas correntes fluidas no escoamento contracorrente através do permutador e é definida por:

$$DTML = \left\{ \frac{(T_1 - T_4) - (T_2 - T_3)}{\ln \left( \frac{T_1 - T_4}{T_2 - T_3} \right)} \right\}$$



### SOLUÇÃO

- A - FLUXOGRAMA: Conforme a página anterior.
- B - Programa FORTRAN

```

*LIST SOURCE PROGRAM
*CNE WORD INTEGERS
*ICCS(2501READER,1403PRINTER)
C   CALCULO DA DIFERENÇA DE TEMPERATURAS MEDIA LOGARITMICA
1   READ(8,1)T1,T2,T3,T4
2   FORMAT(4F10.2)
3   DTML=(((T1-T4)-(T2-T3))/ALCG((T1-T4)/(T2-T3)))
4   WRITE(5,2)DTML
5   2   FORMAT(1X,6HDTML =,F10.2,7HGRAUS F)
6   CALL EXIT
7   END
    
```

C - Aspecto da Listagem de um resultado:

```

PAG. 2 * COPPE-FORTRAN * 09/04/76
// XEQ
DTML * 79.58GRAUS F
FIM DA EXECUCAO
    
```

ou alterando levemente o FORMAT:

```

4   WRITE(5,2)DTML
5   2   FORMAT(1X,6HDTML =,F10.2,2X,7HGRAUS F)
6   CALL EXIT
7   END
    
```

```

DTML * 79.58 GRAUS F
    
```

PROBLEMA 7.3 DESINTEGRAÇÃO RADIOATIVA

A taxa de emissão de partículas por uma substância que sofra uma desintegração radioativa é dada por:

$$R = R_0 e^{-\lambda t} \quad \text{na qual} \quad \lambda = 0.693/T$$

onde R = taxa de emissão no tempo t, em unidades convenientes  
 R<sub>0</sub> = taxa de emissão no tempo zero, idem -1  
 $\lambda$  = constante de desintegração, segundos<sup>-1</sup>  
 t = tempo, em unidades convenientes  
 T = tempo de meia-vida, idem

Desejamos elaborar um programa que leia valores de R<sub>0</sub>, t, e T, calcule  $\lambda$  e R e escreva todos os cinco valores.

PROGRAMA

```

CCCCC
1      REAL LAMB
2      READ(8,1)RO, TMV, T
3      1  FORMAT(3F10.0)
4      LAMB=0.693/TMV
5      R=RO*EXP(-LAMB*T)
6      WRITE(5,2)RO, TMV, T, LAMB, R
7      2  FORMAT(1H ,5(F13.5,2X))
8      CALL EXIT
9      END
    
```

Ou poderíamos ter escrito:

$$R = R_0 * EXP [(-0.693/TMV) * T]$$

E não precisaríamos imprimir valores de LAMB.

Os parênteses para - 0.693/TMV não são realmente necessários, mas foram inseridos para garantir que a fração, ao invés do denominador, seria multiplicada pelo que se segue. Neste caso estamos seguindo uma regra do bom-senso: quando em dúvida, utilizar os parênteses.

RESULTADOS

437.00000	14.50000	39.00000	0.04779	67.76208
-----------	----------	----------	---------	----------





## 8. FORMATAÇÃO DE ENTRADA E SAÍDA DE INFORMAÇÕES

Uma declaração de entrada (ou leitura) ou de saída (ou listagem) de informações deve referir-se sempre a alguma declaração FORMAT correspondente, que define a natureza das variáveis em uso. A declaração FORMAT deve especificar o tamanho exato, o modo e a localização da variável, tanto na leitura de cartões como na listagem de resultados (casos mais frequentes).

As declarações de entrada e saída mais utilizadas no NCE da UFRJ são:

READ (8,n) variáveis - leitura de cartões

WRITE(5,n) variáveis - impressão de resultados  
n FORMAT (especificações das variáveis, nomes etc)

A declaração n FORMAT utiliza caracteres de controle, tipos de especificação e códigos de conversão para introduzir, manipular e obter dados alfanuméricos. Os mais comuns são:

Caracteres de Contrô: lHC, /

Especificações: lW, Fw.d, Ew.d, Aw

Conversões: nH, nX, apóstrofes

O controle do carro das impressoras é feito via declaração FORMAT. As impressoras do sistema IBM-1130 não imprimem o primeiro caracter, especificado pela declaração FORMAT. Este é interpretado como um caracter de controle e poderá ser:

b (vazio)      determinará espaço simples

0 (zero)                      espaço duplo

l                              ejeção de uma folha de relatório

+                              supressão do espaço antes da impressão.

Usa-se comumente a especificação lHC, no início do FORMAT de cada linha, onde C é um dos caracteres acima.

O caracter / (barra) indica que um novo cartão ou uma linha será usado para as variáveis descritas pelas especificações que lhe seguem, tanto na leitura como na impressão da listagem. Também pode ser usado para fazer saltar linhas.

ESPECIFICAÇÃO	VARIÁVEL	SIGNIFICADOS
CONVERSÃO	NÚMERO TÍTULO	
Iw	inteira	w é uma constante inteira sem sinal, que indica o número total de espaços (campo) ocupados pelos dígitos e o sinal.
Fw.d	real	w tem mesmo significado. d indica o número de decimais à direita do ponto decimal.
Ew.d	exponencial	w deverá conter neste caso: 1 posição para o sinal, a parte inteira, o ponto e a parte decimal, e 4 posições para o expoente E.

IMPORTANTE: O expoente é uma constante inteira de um ou dois dígitos, com ou sem sinal, porém nunca maior que 38, e precedido pela letra E. Para entrada de dados, podemos simplificar os critérios da especificação Ew.d, conforme os exemplos apresentados nas páginas seguintes.

Aw	alfabética	w indica o número de letras que compõem a variável, que pode ser manipulada do mesmo modo que as variáveis numéricas.
nH	alfanumérica	aparece dentro da declaração FORMAT seguida somente por n caracteres alfanuméricos.
nX		corresponde aos espaços em branco (ou vazios) na leitura e na listagem, obrigando o computador a ignorar n posições no cartão lido, e a fornecer n espaços vazios na listagem.
apóstrofes	alfanumérica	são semelhantes, em sua aplicação, ao nH; a informação alfanumérica deve estar contida entre apóstrofes.

FOLHA DE CODIFICAÇÃO

C EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA - PRIMEIRO

```

1 READ(8,1)I,J,K,L,M,N
1 FORMAT(6I2)
2 WRITE(5,2)I,J,K,L,M,N
2 FORMAT(4M,1I2)
CALL EXIT
END
    
```

C VÓCE PODERA PERFORAR UM CARTÃO DE  
 C DADOS CONTENDO, POR EXEMPLO:

```

1 2 3 4 5 6
W
    
```

(cada 2 espaços é um campo I2)

```

// FCR
*LIST SOURCE PROGRAM
*CAE MCRC INTEGERS
*ICCS1250IREADER,14C3PRINTER)
C
1 READ(8,1)I,J,K,L,M,N
2 FORMAT(6I2)
3 WRITE(5,2)I,J,K,L,M,N
4 CALL EXIT
5 END
6
    
```

0 ERRC(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO  
 FIM DA COMPILAÇÃO \* ESTAO OCUPADAS 57 DAS 8042 PALAVRAS DISPONIVEIS  
 // XEC  
 1 2 3 4 5 6  
 FIM DA EXECUCAO  
 FCRAP EXECUTADAS 3 INSTRUcoes  
 // \*

FOLHA DE CODIFICAÇÃO

EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA - SEGUNDA	
3	<pre> READ(8,3) A, LING, UAGEM FORMAT(F4.2, I1, I2, 2X, F3.1) E=A ISSO=LING AI=UAGEM WRITE(5,4)E, ISSO, AI FORMAT(1H, F4.2, 2X, I2, 2X, F3.1) CALL EXIT END                     </pre>
4	<pre> READ(8,3)A, LING, UAGEM FORMAT(F4.2, I1, I2, 2X, F3.1) E=A ISSO=LING AI=UAGEM WRITE(5,4)E, ISSO, AI FORMAT(1H, F4.2, 2X, I2, 2X, F3.1) CALL EXIT END                     </pre>
<p>0 ERRC(S) E C ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO</p> <p>FIP CA CCPILACAO * ESTAO OCUPATAS 75 DAS 8042 PALAVRAS DISPONIVEIS</p> <p>// XEC 1.25 -3 0.14</p> <p>VERIFIQUE O FORMAT DESTA ESPACO.</p>	
<p>SEU CARTÃO DE DADOS PODERIA CONTER:</p> <p>1.25 -3 0.14</p> <p>F4.2 I2 F3.1</p> <p>1X 2X</p>	

W deve sempre incluir pelo menos espacos para o digito significativo e para o ponto decimal.  
 A perfuracao do numero real, incluindo o ponto, deve ocorrer sempre dentro de W. Se este for menor que d + 2 ocorrerá truncamento.

FOLHA DE CODIFICAÇÃO

CONT	DEVT
<p>EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA - TERCEIRA</p>	
1	READ (8,5) PO, DES, CRER
2	FORMAT (F40.5, 3X, F4.0, 3X, F8.6)
3	AMI=PO
4	ZA=DES
5	DE=CRER
6	WRITE (5,6) AMI, ZA, DE
7	FORMAT (AH, F10.3, 3X, F4.0, 3X, F8.6)
8	CALL EXIT
9	END
<p>ERR(S) E C ADVERTENCIAIS ASSINALADOS NESTA COMPILAÇÃO</p>	
<p>FIM DA COMPILAÇÃO * ESTAO OCPATAS 77 DAS 8042 PALAVRAS DISPONIVEIS</p>	
<p>// XEC 333.3333C 123. C.123456</p>	
<p>FIM DA EXECUCAO 6 INSTRUcoes</p>	
<p>// *</p>	
<p>SEU CARTAO DE DADOS PODERA CONTER</p>	
333.33333	123. 123456
W	W=0
W	W=d+2
<p>SE O SISTEMA COPIE FORTRAN NA ACEITA W=d OU W=d+1</p>	
<p>QUANDOS VALORES ACIMA SAO PERFUZADOS SEM A ESPECIFICACAO</p>	
<p>CORRETA PARA W, OU SEM O PONTO, O SISTEMA COPIE FORTRAN</p>	
<p>APRESENTA O QUE ESTA ERRADO. VALE A PENALIDADE DE TARDAR NA FOR-</p>	
<p>MAÇAO E NA PERFUZACAO, O RIGOR DA ESPECIFICACAO FOM. 1</p>	

EXEMPLOS	DE	FORMATAÇÃO	DE	ENTRADA	E	SAIADA	-	QUARTO
7	READ (8,7) EXER, CITE, SEM, PRE							
	FORMAT (F7.5, 3X, F4.3, 3X, F5.2, 3X, F6.4)							
	PRA=EXER							
	TI=CITE							
	QUE=SEM							
	A=PRE							
	WRITE (5,8) PRA, TI, QUE, A							
8	FORMAT (A1H, F7.5, 3X, F5.3, 3X, F6.2, 3X, F7.4)							
	CALL EXIT							
	END							
OS NÚMEROS PERIFURADOS E LIDOS SÃO:								
8.765432	A.681	531.35	975.31					
F7.5	3X	F4.3	3X	F5.2	3X	F6.4		
OS RESULTADOS APRESENTADOS NA LISTAGEM, REPRESENTADA NA PÁGINA SEQUINTE								
FORMATAÇÃO DE ENTRADA E SAÍDA - QUARTO								
FORMATAÇÃO DE SAÍDA								
PERIFURACÃO NA								
FORMATAÇÃO								

```

1      READ(7,7)EXER,CITB,SEM,PRE
2      FORMAT(F7.5,3X,F4.3,3X,F5.2,3X,F6.4)
3      PR=EXER
4      TI=CITE
5      QUE=SFH
6      A=PRE
7      WRITE(5,8)PRA,TI,QUE,A
8      FORMAT(IH,F7.5,3X,F5.3,3X,F6.2,3X,F7.4)
9      CALL EXIT
10     END

```

```

0  ERRC(S) E      C ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO
FIM DA COMPILACAO * ESTAO OCUPADAS      95 DAS  8042 PALAVRAS DISPONIVEIS

```

```
// XEB
```

```

X 50 * FOI LIDO UM DADO INTEIRO PARA UMA VARIÁVEL REAL
X 01 * LINHA =      1 * ENCBRECO = 7356
X 40 * CAMPO = 531

```

```

X 50 * FOI LIDO UM DADO INTEIRO PARA UMA VARIÁVEL REAL
X 01 * LINHA =      1 * ENDERECO = 7354
X 40 * CAMPO = 975

```

```

X 42 * NUMERO REAL NAO COUBE EM ESPECIFICACAO *F*
X 01 * LINHA =      7 * ENCBRECO = 7342
X 43 * VALCR = 9.750000E+02

```

```
8.76543  4.600  531.00  *****
```

```
FIM DA EXECUCAO
```

```
FCRAM EXECUTAEAS      7 INSTRUCCOES
```

```
// *
```

FOLHA DE CODIFICAÇÃO

C A FICHATACAB ADEQUADA SERIA

8.765432 4.681 531.36 978.34

F8.6 3X F8.3 3X F6.2 3X F6.2 ou F8.4

C ASSIM TERIAMOS

C 7 FFORMAT(F8.6, 3X, F5.3, 3X, F6.2, 3X, F6.2)

C 8 FORMATCAH, F8.6, 3X, F5.3, 3X, F6.2, 3X, F8.4)

C IMPORTANTE: NÃO MISTURE CARTÕES COM CARTÕES DE DADOS

C VÓCE OBSERVOU NA LISTAGEM QUE A ARMAZENAGEM INTERNA DOS VALORES

C E FOI NA FORMA EXPERIMENTAL?

// XEC 8.765432 4.681 531.35 978.3100

FIP CA EXECUCAC

FCFAM EXECUTACAS 7 INSTRUCOES

// \*

1 REAC(,TI,EXER,CITE,SEM,PRE  
2 7 FFORMAT(F8.6,3X,F5.3,3X,F6.2,3X,F6.2)  
3 PRATEYER  
4 TI=CITE  
5 QUE=SEM  
6 A=PRE  
7 WRITE(5,8)PRA,TI,QUE,A  
8 FFORMAT(H,PF6.6,3X,PF5.3,3X,PF6.2,3X,PF8.4)  
9 CALL FEXIT  
10 ENC

0 ERRC(S) E C ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO

C



FOLHA DE CODIFICAÇÃO

EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA - QUINTA	Lembre-se que o valor do expoente é a potência que deve ser elevado 10, e que multiplicado pela mantissa dá o valor do número.
<pre> 9  READ(8,9) EX, P0MEN, CIAIS    FORMAT(E12.6, 2X, E10.3, 3X, E9.2) EN=EX ROL=P0MEN ADAS=CIAIS WRITE(5,10) EN, ROL, ADAS 10  FORMAT(1H, E12.6, 2X, E10.3, 3X, E9.2) CALL EXIT END </pre>	<p>Compare estes exemplos com as significações das especificações para eliminar suas dúvidas.</p>
<pre> C  OS NUMEROS SAO EX = -12.1234    P0MEN = 123.    CIAIS = -1.23 </pre>	
<pre> C  SEU CARTAO DE DADOS PODERA CONTER -1.21234E+02 </pre>	
<p> <math>d=6</math>  <math>d=3</math>  <math>d=2</math>  <math>d=2</math>  <math>w=10</math>  <math>w=9</math> </p>	

```

*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*10CS(25C IREADER,14C3PRINTER)
C
C

```

```

1 READ(8,9)EX,PONEN,CIAIS
2 9 FORMAT(E12.6,2X,E10.3,3X,E9.2)
3 EN=EX
4 ROL=PONEN
5 ADAS=CIAIS
6 WRITE(5,10)EN,ROL,ADAS
7 10 FORMAT(1H ,E12.6,2X,E10.3,3X,E9.2)
8 CALL EXIT
9 END

```

```

0 ERR(C) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO
FIM DA COMPILACAO * ESTAO OCUPADAS 77 DAS 8042 PALAVRAS DISPONIVEIS
// XEQ

```

```

X 41 * NUMERC REAL NAO COUBE EM ESPECIFICACAO *E*
X 01 * LINHA = 6 * ENDERECO = 734C
X 43 * VALOR = -1.212240E+01

```

```

***** 1.230E+02 -1.27E+00

```

```

FIM DA EXECUCAO

```

```

FCRAM EXECUTADAS 6 INSTRUcoes

```

```

// *

```

O sistema COPPE-FORTRAN armazena os números lidos na forma exponencial. Caso o número lido não contenha dígitos significativos à esquerda do ponto decimal, o sistema retira o primeiro dígito da parte decimal e faz a transformação indicada.

Ele mantém o mesmo número de casas decimais e exige um espaço a mais para o dígito significativo.

Vale a pena, portanto, sempre especificar valores de W maiores que (d + 6). Veja a página seguinte.

// FCF

\*LIST SOURCE PROGRAM  
\*ONE WORD INTEGERS  
\*IOCSI250IRFADER,1403PRINTER)

C  
C

```
1 READ(7,9)BX,PONEN,CIAIS
2 FORMAT(E12.6,2X,E10.3,3X,E9.2)
3 EN=EX
4 ROL=PONEN
5 ACAS=CIAIS
6 WRITE(5,10)EN,ROL,ADAS
7 10 FORMAT(IH ,E14.6,2X,E10.3,2X,E9.2)
8 CALL EXIT
9 ENC
```

0 ERRE(S) E C ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

FIM DA COMPILACAO \* ESTAO OCUPADAS 77 DAS 8042 PALAVRAS DISPONIVEIS

// XEG  
-1.212340E+01 1.230E+02 -1.23E+00

FIM DA EXECUCAO

FCRAM EXECUTACAS 6 INSTRUcoes

// \*

Podemos observar aqui o efeito da formatação E 14.6 na saída. Sobrou um espaço e o valor lido sofreu apenas uma transformação interna, que não prejudicaria qualquer cálculo que fosse efetuado.

FOLHA DE CODIFICAÇÃO

C	EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA - SEXTA								
C	READ (8, 11) P9, R8, FUNA, AMEN, TALS 11 FORMAT(E10.3, 2X, E14.8, 2X, E14.7, 2X, E10.2, 2X, E12.3) V9=PD CE=REM DU=FUNA VI=AMEN JA=TALS WRITE(5, 12) V9, CE, DU, VI, JA 12 FORMAT(1H, E10.3, 2X, E14.8, 2X, E14.7, 2X, E10.2, 2X, E12.3) // XEQ 6.023E+23 * 12345678E+01 1.2345678E-01 -12.34E-00 67.887-7 // *								
C	A LISTAGEM RESULTOU EM:								
	6.023E+23	1.2345678E+00	1.2345670E-01	-1.23E+01	0.000E+00				
	E10.3	E14.8	E14.7	E10.2	E12.3				

FOLHA DE CODIFICAÇÃO

E CONTINUAÇÃO DO SEITO EXEMPLO

E FORAM TESTADOS OS SEQUINTES DADOS:

6.023E+23     1.2345678E+01     1.2345678E-01  
 6.023E+23     1.2345678E+01     1.2345678E-01

E10.3

E14.8

E11.7

OS RESULTADOS FORAM

6.023E+23     1.2345670CE4CC     1.7345670CE-01     1.23E+01     7.887E-07

FIP DA EXECUCAC

FCRAM EXECUTADAS     E INSTRUÇÕES

// 9

6.023E+23     1.2345670CE4CC     1.7345670CE-01     1.23E+01     6.788E+01

FIP DA EXECUCAC

E 10.4

-12.34E-00  
 -12.34E-00

E 10.2

67.887E-7  
 67.887E-7

E 10.11

Aqui o número 67.887E-7 foi perfurado incorretamente.  
 O número 6 foi truncado pois deveria estar perfurado  
 na segunda posição (primeira em branco)

Aqui o número foi perfurado corretamente e deu uma  
 casa decimal truncada (faltou o número 7). Portan-  
 to, a especificação adequada na saída seria E10.4)

Obel foi o outro dígito que desperateou?

Para alguns tipos de computadores, ou sistemas de linguagem, as formas E+01, E 01, E01, +01, E+1, E1, +1 são válidas. As formas de truncar são diferentes de sistema para sistema. O sistema COPPE-FORTRAN só aceita as formas que incluem a letra E.

No exemplo anterior, podemos observar que o Campo W de onde partiu o erro não foi mencionado. Foram truncados três números e o último não foi lido, pois faltava a letra E.

O dígito 8 que constava do segundo e terceiro números desapareceu, devido a lógica interna do sistema de computação. Estes procedimentos foram apresentados pelo sistema COPPE-FORTRAN (listagem reproduzida na página seguinte). Os resultados apresentados pelo computador B 6700 do NCE da UFRJ foram:

```

86700/87700  F O R T R A N  C O M P I L A T I O N  M A R K  2.8.001  WEDNESDAY, 11/10/76
FILE 8=CARTOES,UNIT=READER
FILE 5=SAIDA,UNIT=PRINTER
C
11  REACC(8,11)PO,REM,FUND,AMEN,TAIS          0001000  C
    FORMA T(E10.3,2X,E14.8,2X,E14.7,2X,E10.2,2X,E10.4)  0002000  C
    VO=PO                                         00003000  C
    CE=REM                                         00004000  C
    DU=FUND                                        00005000  C
    VI=AMEN                                       00006000  C
    DA=TAIS                                       00007000  C
    WRITE(5,12)VO,CE,DU,VI,DA                 00008000  C
    FORMA T(CH,E10.3,2X,E14.8,2X,E14.7,2X,E10.2,2X,E9.3)  00009000  C
    CALL EXIT                                   00010000  C
    END                                          00011000  C
                                           00012000  C
                                           00013000  C
                                           00014000  C
                                           00015000  C
002:0000:0
002:000F:2
002:000F:2
002:0010:1
002:0011:0
002:0011:5
002:0012:4
002:0013:3
002:0020:2
002:0020:2
002:0021:1

```

.602E+24 .12345678E+01 .1234568E+00 -.12E+02 0.

Neste exemplo, a formatação foi alterada. Verificou-se que o dígito 8 permaneceu, que este computador lista os expoenciais sem os dígitos significativos e que é necessário, também, perfurar a letra E.

// FCR

\*LIST SOURCE PROGRAM  
\*CNE WORD INTEGERS  
\*ICCSJ2501READER,1403PRINTER)

C  
C

```

1      READ(7,11)PO,REP,FUND,AMEN,TAIS
2      FORMAT(E10.3,2X,E14.8,2X,E14.7,2X,E10.2,2X,E12.3)
3      VO=PO
4      CE=REM
5      CU=FUND
6      VI=AMFN
7      DA=DTATS
8      WRITE(5,12)VO,CE,DU,VI,DA
9      FORMAT(IH ,E10.3,2X,E14.8,2X,E14.7,2X,E10.2,2X,E12.3)
10     CALL EXIT
11     END

```

O ERRE(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTAO OCUPADAS 113 DAS 8042 PALAVRAS DISPONIVEIS

// XEG

X 52 \* CAMPC NUMERICO INVALIDO  
 X 01 \* LINHA = 1 \* ENCERECO = 7352  
 X 40 \* CAMPC = 67.787-7

6.023E+23 1.2345670CE400 1.7345670E-01 -1.23E+01 0.000E+00

FIM DA EXECUCAO

FCRAM EXECUTAEAS 8 INSTRUcoes

// \*

NOE NUCLEO DE COMPUTAÇÃO ELETRÔNICA  
 SISTEMA COPPE - FORTRAN  
 S/D R SISTEMA

FOLHA DE CODIFICAÇÃO

PROGRAMA: **FORMATA CAD**  
 PROPOSTA Nº: **84.864**  
 DATA: **14/01/76**  
 PÁG: **9**  
 CONTA: **0101**

EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA	SETIM
REAO (8,13) FORMATA, CAD 13 FORMAT(13.7,2X,18.2,2X,19.1) ER=FORM RA=ATA CA=CAD WRITE(5,14) ER, RA, DA 14 FORMAT(14,13.7,2X,18.2,2X,19.1) CALL EXIT ENA	1 REAO(8,13)FORMATA,CAD 2 13 FORMAT(13.7,2X,18.2,2X,19.1) 3 ER=FORM 4 RA=ATA 5 CA=CAD 6 WRITE(5,14)ER,RA,DA 7 14 FORMAT(14,13.7,2X,18.2,2X,19.1) 8 CALL EXIT 9 ENA
// XER	0 ERRC(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO
572434E+12	FIM DA COMPILAÇÃO * ESTADO OCUPADAS 77 DAS 8042 PALAVRAS DISPONIVEIS
// *	X 53 * CAPFC NUMERIC INVALIDO X 01 * LINHA = 1 * ENCERECO = 7350 X 40 * CAPFC = 53.865-7
e A LISTAGEM RESULTOU EM	5.724340E+17 0.00E+00 -1.2E+01
5.724340E+17	FIM DA EXECUÇÃO 6 INSTRUCOES
D sistema COPPE/FORTRAN manteve o valor do número lido e o seu as será desobede e mais previstas.	
O sistema COPPE FORTRAN exigiu que a letra E seja perfurada.	
Houve truncamen to parqu o sis tema COPPE/FORTRAN apresenta os números exponenciais em apenas um dígitb significativo.	



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

CPPE - FORTRAN

PROGRAMA FORMATAÇAO  
 PROFESSOR EAKM  
 DATA 21.10.60  
 CONTA

FOLHA DE CODIFICACAO

EXEMPLOS DE	FORMATACAO DE ENTRADA E SAIDA	SETIM (CICLO INICIAL)	NOVA C.A.O.
13	READ (8,13)FORM,ATA,CAO FORMAT(E13.7,2X,E9.3,2X,E10.2) COR=FORM RE=ATA TA=CAO WRITE(5,14)COR,RE,TA FORMAT(1H,E13.7,2X,E10.4,2X,E10.3) CALL EXIT END		
14	53.865E-7 -12.34E-00		
13	FORMATE13.7,2X,E9.3,2X,E10.2)		
14	FORMAT(1H,E13.7,2X,E10.4,2X,E10.3)		
1	READ(F,13)FORM,ATA,CAO		
2	FORMAT(E13.7,2X,E9.3,2X,E10.2)		
3	COR=FORM		
4	RE=ATA		
5	TA=CAO		
6	WRITE(5,14)COR,RE,TA		
7	FORMAT(1H,E13.7,2X,E10.4,2X,E10.3)		
8	CALL EXIT		
9	END		
0	ERRE(S) E	C	ACVERTENCIA(S) ASSINALADOS NESTA COMPILACAO
	FIP CA	COMPILACAO	* ESTAD OCUPATAS 77 DAS 8042 PALAVRAS DISPONIVEIS
	// XEC		5.724340E+10 5.3865E-06 -1.274E+01
	FIP CA	EXECUCAO	
	FCPAM	EXECUTACAO	6 INSTRICOES
	// *		



Os formatos 15 e 16 originais resultam na listagem abaixo:

```
1 READ(P,15)OU,TRAS,POS,SI,BILI,DA,DES
2 15 FORMAT(E9.2,1X,E14.7,1X,E9.3,1X,E14.8,1X,E10.3,1X,E14.0,1X,E7.1)
3 WRITE(5,16)OU,TRAS,POS,SI,BILI,DA,DES
4 16 FORMAT(1H ,E9.2,1X,E14.7,1X,E9.3,1X,E14.8,1X,E10.3,1X,E14.8,1X,E7.
  11)
5 CALL EXIT
6 END
```

0 ERRO(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO


FIM DA COMPILACAO \* ESTAO OCUPADAS 93 DAS 8042 PALAVRAS DISPONIVEIS

// XEQ

```
X 53 * CAMPO NUMERICO INVALIDO
X 01 * LINHA = 1 * ENDERECO = 7352
X 40 * CAMPO = 9.876E+5 1
```

```
X 53 * CAMPO NUMERICO INVALIDO
X 01 * LINHA = 1 * ENDERECO = 7350
X 40 * CAMPO = 345678.E-20 0.
```

```
X 38 * FOI EXCEDIDO O TAMANHO DA AREA DE LEITURA - 80 CARACTERES
X 01 * LINHA = 1
```



```
X 49 * NAO HA MAIS CARTOES DE DADOS DISPONIVEIS
X 01 * LINHA = 1
```

X 03 \* EXECUCAO INTERROMPIDA

FCRAM EXECUTADAS 1 INSTRICOES

Respeitando a formatação do cartão de dados, obtemos:

```
C
1 READ(9,15)OU,TRAS,POS,SI,BILI,DA,DES
2 15 FORMAT(E9.2,1X,E14.7,1X,E9.3,1X,E14.7,1X,E8.2,1X,E13.0,1X,E7.1)
3 WRITE(5,16)OU,TRAS,POS,SI,BILI,DA,DES
4 16 FORMAT(IH ,E9.2,1X,E14.7,1X,E9.3,1X,E14.7,1X,E8.2,1X,E13.0,1X,E7.1
1)
5 CALL EXIT
6 END

0 ERRO(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO
FIM DA COMPILACAO * ESTAO OCUPADAS 93 DAS 8042 PALAVRAS DISPONIVEIS
// XEG
-1.23E+02 1.2345670E-02 1.235E+14 -8.7654330E+15 9.87E+05 1.E-13 0.0E+00
FIM DA EXECUCAO
FCRAM EXECUTADAS 3 INSTRUICOES
// *
```



FOLHA DE CODIFICAÇÃO

PRINCIPAL *FORMATACAO*  
 SECUNDARIA *EXAMEN*  
 DATA *13.03.76*  
 C.N.

EXEMPLOS DE FORMATAÇÃO DE ENTRADA E SAÍDA - DECIMAL
20 READ(8,20) 20 FORMAT(80H 1 WRITE(5,20)
21 READ(8,21) 21 FORMAT(AH,79H 1 WRITE(5,21)
22 READ(8,22)A,K 22 FORMAT(28H WRITE(5,22)A,K ,F5.2,2X,I4)
23 WRITE(5,23)A,K 23 FORMAT(AH,27H ,F5.2,2X,I4)
// XEE QUALQUER EXPRESSÃO QUALQUER EXPRESSÃO QUE USE OS 79 ESPAÇOS DIVIRTA-SE CONTANDO ESPAÇOS 27.27 4444
// *

```

// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*IOCS(250IRFADER,1403PRINTER)
C
C
1 READ(5,20)
2 FORMAT(80H
3 WRITE(5,20)
C
C
4 READ(8,21)
5 FORMAT(1H ,79H
6 WRITE(5,21)
C
C
7 READ(9,22)A,K
8 FORMAT(28H
9 WRITE(5,22)A,K
C
C
10 WRITE(5,23)A,K
11 FORMAT(1H ,27H
C
C
12 WRITE(5,24)
13 FORMAT(1H ,E MAIS FACIL USAR APOSTROFOS,1X,PODEMOS ATE SEPARAR.
I O C E S P A COS SEM COLOCAR X ENTRE VIRGULAS.)
C
C
14 CALL EXIT
15 END
C
C
0 ERR(C) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO
FIM DA COMPILACAO * ESTAO OCUPADAS 259 DAS 8042 PALAVRAS DISPONIVEIS
// XEC
QUALQUER EXPRESSAO QUE NAO ULTRAPASSE OS 79 ESPACOS
QUALQUER EXPRESSAO QUE USE OS 79 ESPACOS
DIVIRTA-SE CONTANDO ESPACOS 27.27 4444
27.27 4444
E MAIS FACIL USAR APOSTROFOS PODEMOS ATE SEPARAR O S E S P A COS SEM COLOCAR X ENTRE VIRGULAS
FIM DA EXECUCAO
FCRAM EXECUTADAS 9 INSTRUICOES
// *

```

## 9. Outros Exemplos Simples.

### 9.1 TEMPO DE ENCHIMENTO DE UM TANQUE CILÍNDRICO

Faça um programa FORTRAN para o cálculo do tempo de enchimento de um tanque cilíndrico. São dados o raio do tanque ( $R=8.00\text{ft}$ ), a altura ( $H=20.00\text{ft}$ ) e a vazão de alimentação ( $Q=50.000\text{ gpm}$ ).

A - Primeira Solução do Programa:

$$1 \text{ ft}^3 = 7,48 \text{ galões}, V = \pi R^2 H \quad \text{Tempo} = V/Q$$

```
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*ICCS(2501READER,1403PRINTER)
C CALCULO DO TEMPO DE ENCHIMENTO DE UM TANQUE CILINDRICO
C
1  READ(8,10)R,H,Q
2  10  FORMAT(2F7.2,F10.2)
3     PI=3.1416
4     V=PI*R*R*H
5     TIME=V/((Q*60.)/7.48)
6     WRITE(5,11)TIME
7  11  FORMAT(1X,'TIME=',2X,F7.2)
8     CALL EXIT
9     END
```

B - Segunda Solução:

Podemos eliminar os cartões de PI e V, inserindo suas informações em apenas um único cartão, oriundo da seguinte expressão:

$$\text{TIME} = \left( \frac{3,1416 \times 7,48 \times R^2 \times H}{Q \times 60,0} \right)$$

Portanto:

```
1  READ(8,10)R,H,Q
2  10  FORMAT(2F7.2,F10.2)
3     TIME = (0.3916*R*R*H)/Q
```

Com esta expressão de TIME economizamos tanto no espaço da memória como na operação do computador, além do tempo e cartões consumidos na perfuração. Logicamente, para problemas pequenos esta economia é aparentemente irrisória, mas para problemas maiores as simplificações acumuladas geram lucros maiores.



C - Aspecto da Listagem de um resultado:

TIME= 0.01

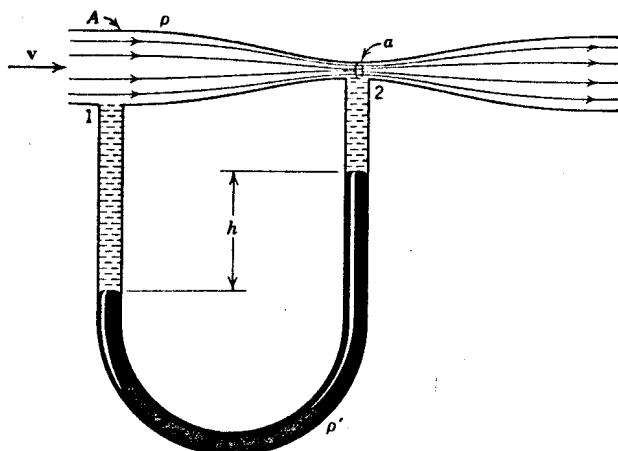
Faltou indicar na formatação do WRITE qual seria a unidade de tempo, mas um cartão C (comentário), colocado no início do programa, conforme o problema, poderia conter esta indicação. Isto fica a critério do programador. Ou então, conforme já exemplificamos, o próprio cartão de FORMAT poderia incluir a especificação da unidade de tempo, conforme o aspecto abaixo:

```
4  
5      11 WRITE(5,11)TIME  
6      11 FORMAT(1X,'TIME =',2X,F7.2,2X,'HORAS')  
7      CALL EXIT  
      END
```

```
PAG. 2 * COPPE-FORTRAN * 12/04/76  
FIM DA COMPILACAO * ESTAO OCUPADAS 69 DAS 7782 PALAVRAS DISPONIVEIS  
// XEC  
TIME = 0.01 HORAS  
FIM DA EXECUCAO  
FORAM EXECUTADAS 4 INSTRUCCES  
// *
```

PROBLEMA 9 . 2 MEDIDOR VENTURI

O medidor Venturi é usado para quantificar o escoamento de um líquido através de um tubo, conforme a figura abaixo.



Um líquido de densidade  $\rho$  escoa através de um tubo cuja área de seção reta é  $A$ . No ponto de estrangulamento a área é reduzida para  $a$ , onde um tubo manométrico está ligado ao tubo. O líquido manométrico, por exemplo, mercúrio, tem uma densidade  $\rho'$ . A taxa de escoamento, ou vazão volumétrica é dada por:

$$Q = aA \sqrt{\frac{2 (\rho' - \rho) gh}{\rho (A^2 - a^2)}}$$

onde  $Q$  = vazão volumétrica, pés cúbicos por segundo

$a, A$  = áreas, pés quadrados

$\rho, \rho'$  = massas específicas, em unidades consistentes

$g$  = aceleração da gravidade, pés por segundo

$h$  = diferença de níveis, pés.

Neste caso desejamos elaborar um programa que irá preparar uma tabela de vazões volumétricas, em função das diferenças de níveis, para valores fixos de áreas e densidades. O programa lerá um cartão que contém as áreas e as massas específicas, escreverá estes valores, então, lerá diversos cartões adicionais, cada um contendo uma altura (diferença de níveis). Para cada cartão de altura serão impressos a altura e a vazão volumétrica correspondente, que nos permitirão obter um gráfico ou uma tabela para uso junto ao medidor Venturi.

Este programa conterà dois comandos de leitura e um comando que provocará a repetição de parte do programa tantas vezes quantas existem cartões de altura a serem lidos.

O que acontecerá quando não existirem mais cartões de dados a serem lidos ? O sistema interno detectará esta condição e emitirá a seguinte advertência:

PROGRAMA

```

C          CALCULO DA VAZAO VOLUMETRICA PELO MEDIDOR VENTURI
C          ESTA PRIMEIRA LEITURA E EXECUTADA APENAS UMA VEZ
C          READ(P,1)A1,A2,RHO,RHOLI
C          FORMAT(8F10.0)
1          1 IMPRESSAO DE UMA LINHA CONTENDO OS PARAMETROS CONSTANTES
C          WRITE(5,2)A1,A2,RHO,RHOLI
2          2 FORMA T(1H,4(E13.5,2X))
C          ESTA SEGUNDA LEITURA E EXECUTADA UMA VEZ POR CADA CARTAO DE ALTURA
3          3 READ(Q,1)H
C          CALCULO DA VAZAO VOLUMETRICA POR ALTURA
C          Q=A1*A2*SQRT(64.4*(RHOLI-RHO)*H/(RHO*(A1**2-A2**2)))
C          IMPRESSAO DOS VALORES DE ALTURA E VAZAO VOLUMETRICA
7          7 WRITE(5,2)H,Q
C          LEIA UM NOVO VALOR DE ALTURA E REPITA O CALCULO
8          8 GO TO 3

```

\* CPPE-FORTRAN \* 26/10/76

END

PAG. 2

9

O ERRO(S) E O ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

FIM DA COMPILACAO \* ESTAO OCUPADAS 131 DAS 8042 PALAVRAS DISPONIVEIS

```

// XEC
6.00000E-02
1.00000E-01
2.00000E-01
3.00000E-01
4.00000E-01
5.00000E-01
6.00000E-01
7.00000E-01
8.00000E-01
9.00000E-01
1.00000E+00
1.10000E+00
3.00000E-02
3.11426E-01
4.40423E-01
5.39406E-01
6.22852E-01
6.96370E-01
7.62835E-01
8.23956E-01
8.80846E-01
9.34278E-01
9.84816E-01
1.03288E+00
1.00000E+00
1.35500E+01

```

X 49 \* NAO HA MAIS CARTOES DE DADOS DISPONIVEIS  
X 01 \* LINHA = 5

X 03 \* EXECUCAO INTERROMPIDA

FORAM EXECUTADAS 47 INSTRUICOES

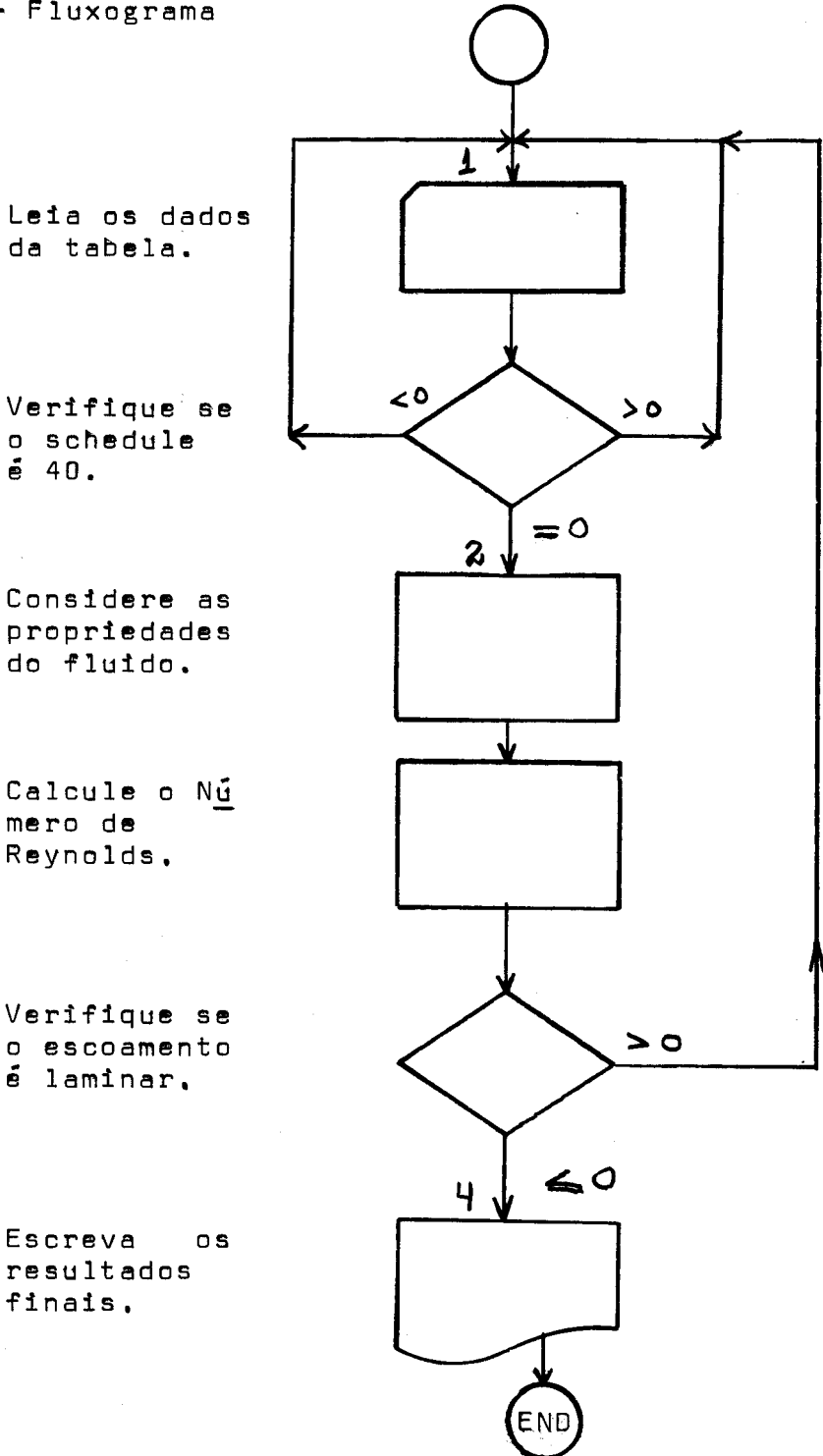
### 9.3 DIMENSIONAMENTO DE TUBOS

#### QUANDO A PERDA DE CARGA É DESPREZÍVEL

Escolher um diâmetro nominal de um tubo Sch. 40 para que o escoamento seja efetuado ao longo de um trecho reto em regime laminar, considerando que o fluido é um óleo com viscosidade dinâmica de 5 cp e massa específica de 60 lb/ft<sup>3</sup>. A vazão do produto é de 50 litros por minuto e a perda de carga é desprezível. A tabela a ser utilizada para a escolha do tubo é a mencionada no "Appendix C-6" do livro de Foust et alii, e deverá ser incluída no programa tanto para o Sch. 40 como para o Sch. 80.

#### SOLUÇÃO:

#### A - Fluxograma



B - Programa

// FCR

```
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*ICCS(2501READER,1403PRINTER)
C
C
C CALCULO DO DIAMETRO DE UM TUBO
C QUANDO A PERDA DE CARGA E DESPREZIVEL
C
C CONVERSOES DE UNIDADES DO PROBLEMA
C LB/(FT*SEC) = CP*6.72E-4
C (FT**3)/SEC = (L/MIN)/(28.316*60.)
C FT = 12.*INCH
C
C NOMENCLATURA EMPREGADA
C DNOM = DIAMETRO NOMINAL DO TUBO
C ISCHE = SCHEDULE DO TUBO
C DIN = DIAMETRO INTERNO DO TUBO
C ARIN = AREA INTERNA DO TUBO
```

PAG. 2

\* CCPPE-FORTRAN \*

19/04/76

```
C
1 1 READ(8,10)DNOM,ISCHE,DIN,ARIN
2 10 FORMAT(F10.3,I10,F10.3,F10.5)
3 IF(ISCHE-40)1,2,1
4 2 Q=50.
5 RHO = 60.
6 VISCO = 5.
7 RE=(DIN*RHO*Q/(VISCO*ARIN))*0.073
8 IF(RE-2100.)4,4,1
9 4 WRITE(5,20)RE, DNOM
10 20 FORMAT(2F10.3)
11 CALL EXIT
12 END
```

0 ERRO(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
FIM DA COMPILACAO \* ESTAO OCUPADAS 125 DAS 7782 PALAVRAS DISPONI

// XEC  
19941782 4.000

FIM DA EXECUCAO

FCRAM EXECUTADAS 117 INSTRUcoes

// \*

C - Comentários

- a) Esta listagem mostra que existem duas formas de entrada de dados para o problema; primeira - sob a forma do cartão READ; segunda - sob a forma de declarações simples (2<sup>o</sup> Q=50., RHO = 60., VISCO = 5,1).

- b) Este programa envolve tentativas e erros, portanto envolve ITERAÇÃO. Em certos casos, como iremos ver adiante, o próprio programa gera valores tentativos. Neste aqui os valores tentativos são fornecidos pela tabela, lançada nos cartões de entrada.
- c) Este programa não nos informa, na listagem, quais os valores lidos que o computador testou. Só apresenta o resultado final.
- d) Para  $Q = 100$ .  $Re = 2012,479$   $DNOM = 8.000$ .

PAG. 2

\* CCPPE-FORTRAN \*

14/04/76

```

C
1 1 READ(8,10)DNOM,ISCHE,DIN,ARIN
2 10 FORMAT(F10.3,I10,F10.3,F10.5)
3 WRITE(5,10)DNOM,ISCHE,DIN,ARIN
4 IF(ISCHE-4C)1,2,1
5 2 Q=10.
6 RHO = 6C.
7 VISCO = 5.
8 RE=(DIN*RHO*Q/(VISCO*ARIN))*0.073
9 IF(RE-21CC.)4,4,1
10 4 WRITE(5,20)RE, DNOM
11 20 FORMAT(2F10.3)
12 CALL EXIT
13 END

```

0 ERRO(S) E C ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO  
 FIM DA COMPILAÇÃO \* ESTÃO OCUPADAS 139 DAS 7782 PALAVRAS DISPONÍVEIS

// XEG			
0.125	40	0.269	C.CCC40
0.125	80	0.215	C.00025
0.250	40	0.364	C.00072
0.250	80	0.302	C.CCC50
0.375	40	0.493	C.CC133
0.375	80	0.423	C.CC098
0.500	40	0.622	C.CC211
0.500	80	0.546	C.CC163
0.750	40	0.824	C.CC371
1945.617	0.750		

FIM DA EXECUÇÃO  
 FORAM EXECUTADAS 54 INSTRUÇÕES  
 // \*

- e) Aqui foi incluída a declaração WRITE (5,10) DNOM, ISCHE, DIN, ARIN para fornecer um "espelho" dos valores lidos pelo computador até a obtenção do resultado final. Observe o valor de Q e a listagem correspondente.

PAG. 2

\* COPPE-FORTRAN \*

14/04/76

```

1      1 READ(8,10)DNOM,ISCHE,DIN,ARIN
2
3      10 FORMAT(F10.3,I10,F10.3,F10.5)
4      WRITE(5,10)DNOM,ISCHE,DIN,ARIN
5      IF(ISCHE-40)1,2,1
6
7      2  Q=2CC.
8      RFO = 60.
9      VISCO = 5.
10     RE=(DIN*RFO*Q/(VISCO*ARIN))*0.073
11     IF(RE-21CC.)4,4,1
12     4  WRITE(5,20)RE, DNOM
13     20 FORMAT(2F10.3)
14     CALL EXIT
15     ENC

```

O ERRC(S) E C ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTAO OCUPADAS 139 DAS 7782 PALAVRAS DISPON

// XEQ

0.125	40	0.269	0.00040
0.125	80	0.215	0.00025
0.250	40	0.364	0.00072
0.250	80	0.302	0.00050
0.375	40	0.493	0.00133
0.375	80	0.423	0.00098
0.500	40	0.622	0.00211
0.500	80	0.546	0.00163
0.750	40	0.824	0.00371
0.750	80	0.742	0.00300
1.000	40	1.049	0.00600
1.000	80	0.957	0.00499
1.250	40	1.380	0.01040
1.250	80	1.278	0.00891
1.500	40	1.610	0.01414
1.500	80	1.500	0.01225
2.000	40	2.067	0.02330
2.000	80	1.939	0.02050

PAG. 3

\* COPPE-FORTRAN \*

14/04/76

2.500	40	2.469	0.03322
2.500	80	2.323	0.02942
3.000	40	3.068	0.05130
3.000	80	2.900	0.04587
3.500	40	3.548	0.06870
3.500	80	3.364	0.06170
4.000	40	4.026	0.08840
4.000	80	3.826	0.07986
5.000	40	5.047	0.13900
5.000	80	4.812	0.12630
6.000	40	6.065	0.20060
6.000	80	5.761	0.18100
8.000	40	7.981	0.34740
8.000	80	7.625	0.31710
10.000	40	10.020	0.54750
10.000	80	9.564	0.49890
12.000	40	11.938	0.77730
12.000	80	11.376	0.70580

X 49 \* NAC HA MAIS CARTOES DE DADOS DISPONIVEIS  
 X 01 \* LINHA = 1

X 03 \* EXECUCAO INTERROMPIDA

FCRAM EXECUTADAS 199 INSTRUcoes

// \*

Nesta "corrida do computador" o valor de Q era alto demais e os cartões de dados não apresentavam valores capazes de fornecer um resultado satisfatório ao problema. O diagnóstico final indica "X 03 \* EXECUÇÃO INTERROMPIDA" ao invés de "FIM DA EXECUÇÃO". Portanto o cartão WRITE mencionado foi útil para fornecer uma idéia ao engenheiro dos erros de programação em que êle estava incorrendo. Pois o programa estava certo, mas os dados fornecidos estavam mal-dimensionados.



## PROBLEMA 9.4 OPERAÇÕES BÁSICAS COM NÚMEROS COMPLEXOS

A manipulação de números complexos na UFRJ é efetuada pelos computadores IBM/360 e B-6700. Portanto o texto abaixo é apresentado para complementar as informações que os leitores, presumivelmente estudantes de Engenharia Química, estão recebendo.

Uma constante complexa FORTRAN é composta de duas constantes reais separadas por vírgulas, dentro de parênteses. Portanto, (2.0,3.0) seria uma representação aceitável daquilo que escreveríamos em notação matemática como  $2+3i$  ou em notação de engenharia elétrica  $2+j3$ . Um número real puro escrito em notação complexa possui simplesmente uma parte imaginária nula, tal como (4.67,00). Ao contrário, um número imaginário puro possui uma parte real nula, tal como (0.0,91.0).

Qualquer forma para uma constante real pode ser usada na redação de números complexos, de modo que os exemplos abaixo, são constantes complexas corretas.

```
(0., 659)
(1.57, 4.5 E-10)
(.1905, 7.62 E2)
```

Uma variável complexa é qualquer uma que tenha sido definida através de uma declaração COMPLEX. Não existem restrições à nomenclatura. Uma variável complexa FORTRAN consiste de dois números representando as partes real e imaginária; cada um destes números pode assumir qualquer valor permitido às variáveis reais. Poderíamos ter, por exemplo, um programa contendo as seguintes declarações:

```
COMPLEX Z, IMPED
Z = (2.0, 87.344)
IMPED = (0.0,0.0056)
```

As quatro operações aritméticas e a exponenciação são todas representadas pelos mesmos símbolos aplicáveis aos casos real e inteiro. É claro que as diversas operações requerem ações separadas nas partes real e imaginárias. Convém relembrar as seguintes operações:

$$(a+bi) + (c+di) = (a+c) + (b+d) i$$

$$(a+bi) - (c+di) = (a-c) + (b-d) i$$

$$(a+bi) * (c+di) = (ac-bd) + (ad+bc) i$$

$$(a+bi) / (c+di) = \frac{(ac+bd)}{c^2+d^2} + \frac{(bc-ad)}{c^2+d^2} i$$

Uma expressão para exponenciação não é apresentada porque o método de elevar um número complexo para uma potência, através de computador, depende da magnitude do expoente. Para pequenas potências uma multiplicação é empregada para obter o mesmo efeito. Para grandes potências o número pode ser convertido para a forma polar, empregando-se a fórmula de De Moivre, da seguinte maneira:

$$a + bi = \rho (\cos \theta + i \sin \theta)$$

sendo  $\rho = \sqrt{a^2 + b^2}$  = valor absoluto ou módulo ou magnitude

$$\theta = \text{arc tg } (b/a) \text{ amplitude}$$

Portanto  $(a + bi)^n = \rho^n (\cos n\theta + i \sin n\theta)$

Uma quantidade complexa pode ser elevada apenas a uma potência inteira; não pode ser elevada a uma potência real ou complexa (usando o operador FORTRAN\*\*). A operação de exponenciação no caso pode ser definida matematicamente e executada através de funções.

As quatro operações aritméticas são definidas para pares de quantidades complexas, baseadas nas fórmulas anteriores. Elas também podem ser usadas quando um dos números é complexo e o outro real. Portanto, se Z é complexo e R é real, podemos admitir expressões dos tipos:

$$\begin{aligned} &2.0 + (3.0,4.0) \\ &6.5 * Z \\ &Z / R \end{aligned}$$

O programa exemplificado a seguir nos apresenta os seguintes aspectos.

- a) três variáveis foram declaradas complexas;
- b) uma das variáveis assume a posição de função das operações aritméticas efetuadas com algumas constantes complexas;
- c) cada resultado é impresso individualmente, devido ao comando WRITE especificado para cada uma das operações efetuadas;
- d) a linha 6 demonstra que, se dois números complexos são reais puros escritos em notação complexa, o resultado é correto;
- e) aplicado a números imaginários puros, entretanto, o resultado é diferente (linha 7);
- f) a linha 8 demonstra que um dos números complexos pode ser real puro;
- g) as linhas 9 e 10 demonstram que um dos números pode ser redigido em notação FORTRAN real, e o resultado fornecido será em notação complexa;
- h) a linha 11 envolve <sup>variáveis</sup> ~~variáveis~~ complexas ao invés de constantes e demonstra que o processamento admite e fornece os resultados esperados;

- i) a linha 12 apresenta o primeiro exemplo de uma função: valor absoluto, que aceita um argumento complexo e oferece um número real simples como resultado da função. Neste caso

$$\text{CABS} (a + bi) = \sqrt{a^2 + b^2}$$

- j) as linhas 13 e 14 ilustram a aplicação de duas funções frequentemente usadas. REAL aceita um argumento complexo e apresenta a parte real como um número real FORTRAN simples. AIMAG aceita um argumento complexo e apresenta a parte imaginária como um número real FORTRAN simples;

- k) a linha 15 é um exemplo de uma função de uma constante: a raiz quadrada complexa. Observe os parênteses duplos, um para função argumento e outro para a constante complexa. Note também que esta função oferece apenas uma das duas raízes quadradas, justamente como sua correspondente real faz;  $-3 + 2i$  é também uma raiz quadrada de  $5 - 12i$ . Semelhantemente a linha 16 mostra que a função raiz quadrada complexa oferece somente uma das raízes quadradas de  $-1$ . Aliás a expressão para raízes quadradas complexas é dada por

$$\text{CSQRT} (a + bi) = \frac{-b}{2a} + \frac{\{(-1)(b^2 - 4ac)\}^{1/2}}{2a}$$

- l) A exponencial complexa é demonstrada nas linhas 17 e 18. Esta última é uma verificação da famosa Equação de Euler

$$e^{ni} + 1 = 0$$

que combina em uma simples equação os cinco números mais importantes da Matemática

- m) Para informação complementar, eis as definições de algumas funções complexas comuns, em termos das operações sobre suas partes real e imaginária.

$$\text{CEXP} (a + bi) = e^a (\cos b + i \sin b)$$

$$\text{CLOG} (a + bi) = (1/2) \log (a^2 + b^2) + i \arctg (b/a)$$

$$\text{CSEN} (a + bi) = \sin (a(e^b + e^{-b})/2) + i \cos (a(e^b - e^{-b})/2)$$

$$\text{CCOS} (a + bi) = \cos (a(e^b + e^{-b})/2) - i \sin (a(e^b - e^{-b})/2)$$

PROGRAMA

```

FILE 8=CARIOES,UNIT=READER
FILE 5=SAIDA,UNIT=PRINTER
C      EXEMPLOS DE OPERACOES COM NUMEROS COMPLEXOS
C
      COMPLEX X,Y,Z
      Z=(1.0,2.0) + (3.0,4.0)
      WRITE(5,1) Z
1  FCRMAT(1H,' LINHA 1 (1.0,2.0) + (3.0,4.0) = ',2F8.2)
      Z=(2.0,6.0) - (3.0,3.0)
      WRITE(5,2) Z
2  FCRMAT(1H,' LINHA 2 (2.0,6.0) - (3.0,3.0) = ',2F8.2)
      Z=(1.0,3.0) * (2.0,4.0)
      WRITE(5,3) Z
3  FCRMAT(1H,' LINHA 3 (1.0,3.0) * (2.0,4.0) = ',2F8.2)
      Z=(5.0,10.0) / (2.0,1.0)
      WRITE(5,4) Z
4  FCRMAT(1H,' LINHA 4 (5.0,10.0) / (2.0,1.0) = ',2F8.2)
      Z=(3.0,-2.0)**2
      WRITE(5,5) Z
5  FCRMAT(1H,' LINHA 5 (3.0,-2.0)**2 = ',2F8.2)
      Z=(4.0,0.0) * (5.0,0.0)
      WRITE(5,6) Z
6  FCRMAT(1H,' LINHA 6 (4.0,0.0) * (5.0,0.0) = ',2F8.2)
      Z=(0.0,4.0) * (0.0,5.0)
      WRITE(5,7) Z
7  FCRMAT(1H,' LINHA 7 (0.0,4.0) * (0.0,5.0) = ',2F8.2)
      Z=(6.0,0.0) + (-4.0,-2.0)
      WRITE(5,8) Z
8  FCRMAT(1H,' LINHA 8 (6.0,0.0) + (-4.0,-2.0) = ',2F8.2)
      Z=6.0 + (-4.0,-2.0)
      WRITE(5,9) Z
9  FCRMAT(1H,' LINHA 9 6.0 + (-4.0,-2.0) = ',2F8.2)
      Z=(5.0,10.0) / 2.0
      WRITE(5,10) Z
10 FCRMAT(1H,' LINHA 10 (5.0,10.0) / 2.0 = ',2F8.2)
      X=(1.0,2.0)
      Y=(3.0,4.0)
      Z=X * Y
      WRITE(5,11) Z
11 FCRMAT(1H,' LINHA 11 Z = ',2F8.2)
      R=CAES(Y)
      WRITE(5,12) R
12 FCRMAT(1H,' LINHA 12 ABS(Y) = ',2F8.2)
      R=REAL(X)
      WRITE(5,13) R
13 FCRMAT(1H,' LINHA 13 REAL(X) = ',F8.2)
      R=AIMAG(X)
      WRITE(5,14) R
14 FCRMAT(1H,' LINHA 14 AIMAG(X) = ',F8.2)
      Z=CSQRT((5.0,-12.0))
      WRITE(5,15) Z
15 FCRMAT(1H,' LINHA 15 CSQRT((5.0,-12.0)) = ',2F8.2)
      Z=CSQRT((-1.0,0.0))
      WRITE(5,16) Z
16 FCRMAT(1H,' LINHA 16 CSQRT((-1.0,0.0)) = ',2F8.2)
      Z=CEXP((1.0,2.0))
      WRITE(5,17) Z
17 FCRMAT(1H,' LINHA 17 CEXP((1.0,2.0)) = ',2F12.6)
      Z=CEXP((0.0,3.14159265))
      WRITE(5,18) Z
18 FCRMAT(1H,' LINHA 18 CEXP((0.0,3.14159265)) = ',2F12.6)
      CALL EXIT
      END

```

```

LINHA 1 (1.0,2.0) + (3.0,4.0) = 4.00 6.00
LINHA 2 (2.0,6.0) - (3.0,3.0) = -1.00 3.00
LINHA 3 (1.0,3.0) * (2.0,4.0) = -10.00 10.00
LINHA 4 (5.0,10.0) / (2.0,1.0) = 4.00 3.00
LINHA 5 (3.0,-2.0)**2 = 5.00 -12.00
LINHA 6 (4.0,0.0) * (5.0,0) = 20.00 0.00
LINHA 7 (0.0,4.0) * (0.0,5.0) = -20.00 0.00
LINHA 8 (6.0,0.0) + (-4.0,-2.0) = 2.00 -2.00
LINHA 9 6.0 + (-4.0,-2.0) = 2.00 -2.00
LINHA 10 (5.0,10.0) / 2.0 = 2.50 5.00
LINHA 11 Z = -5.00 10.00
LINHA 12 ABS(Y) = 5.00
LINHA 13 REAL(X) = 1.00
LINHA 14 AIMAG(X) = 2.00
LINHA 15 CSQRT((5.0,-12.0)) = 3.00 -2.00
LINHA 16 CSQRT((-1.0,0.0)) = 0.00 1.00
LINHA 17 CEXP((1.0,2.0)) = -1.131204 2.471727
LINHA 18 CEXP((0.0,3.14159265)) = -1.000000 0.000000

```

#### 10. BIBLIOGRAFIA

1. Forsythe, A.I. e outros, "Ciência dos Computadores", Vol.1,2  
Ao Livro Técnico S.A. - RJ (1972).
2. Pacitti, T., "FORTRAN - MONITOR, Princípios"  
Livros Técnicos e Científicos Editora S.A. - RJ (1974).
3. McCracken, D.D., "FORTRAN with Engineering Applications"  
John Wiley & Sons, Inc. - NY (1967).
4. Moyle, M.P., "Introduction to Computers for Engineers"  
John Wiley & Sons, Inc. - NY (1967).

## QUANTIFICAÇÃO DE FENOMENOS E MATERIAIS

Muitos engenheiros e empresários consideram-se indivíduos essencialmente " práticos e objetivos " e rejeitam aqueles profissionais com uma base teórica sólida, ou que se dedicam às pesquisas de caráter científico. O computador digital é um instrumento importantíssimo para servir de ligação entre as teorias que envolvem um fenômeno, um material e um processo e suas aplicações de caráter industrial, comercial ou doméstico. A simulação de situações ou a previsão de resultados pode ser efetuada em segundos por um computador, oferecendo inúmeras listagens de resultados hipotéticos que podem ocorrer na realidade que nos cerca.

Os próximos exemplos nos darão uma visão de como é possível utilizar métodos matemáticos para projetos ou pesquisas de Engenharia Química.

## 1.1. TRANSFERENCIA DE CALOR EM PAREDE PLANA COMPOSTA.

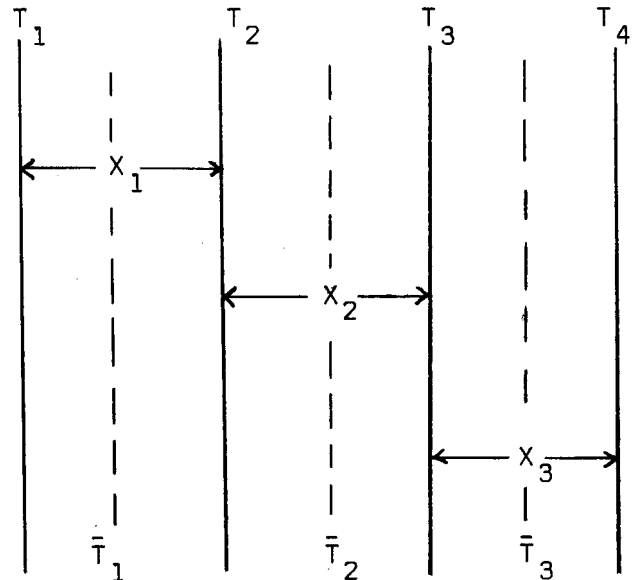
É necessário determinar o fluxo térmico em uma parede plana composta de materiais isolantes diferentes. As espessuras das camadas,  $X_1$ ,  $X_2$ ,  $X_3$  e as temperaturas externas,  $T_1$ ,  $T_4$ , são conhecidas. A condutividade térmica  $K_i$  de cada camada, que é uma função linear de temperatura média  $\bar{T}_i$ , não é conhecida. As equações abaixo descrevem o fluxo de calor  $Q$ , que é a taxa de transferência por unidade de área as condutividades térmicas  $K_i$ , onde os valores dos A's e B's são constantes respectivas aos materiais das camadas.

$$Q = \frac{T_1 - T_4}{\frac{X_1}{K_1} + \frac{X_2}{K_2} + \frac{X_3}{K_3}}$$

$$\bar{K}_1 = A_1 \bar{T}_1 + B_1$$

$$\bar{K}_2 = A_2 \bar{T}_2 + B_2$$

$$\bar{K}_3 = A_3 \bar{T}_3 + B_3$$



As temperaturas médias são obtidas pelas seguintes equações:

$$\bar{T}_1 = (T_1 + T_2) / 2$$

$$\bar{T}_2 = (T_2 + T_3) / 2$$

$$\bar{T}_3 = (T_3 + T_4) / 2$$

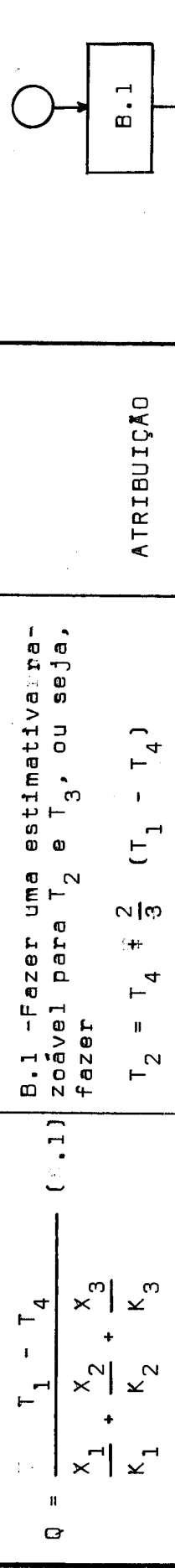
Sendo  $T_2$  e  $T_3$  desconhecidos, temos de empregar algumas considerações e artifícios matemáticos que nos permitam, através de iterações e aproximações sucessivas, obter os valores das temperaturas, condutividades e do fluxo térmico  $Q$  (que será neste caso uma perda de calor).

Considerando que o fluxo térmico apresenta regime permanente através das camadas, então, em cada unidade de tempo esta quantidade será a mesma nos três materiais; portanto:

$$\frac{K_1}{X_1} (T_1 - T_2) = \frac{K_2}{X_2} (T_2 - T_3) = \frac{K_3}{X_3} (T_3 - T_4)$$

onde obtemos:

$$T_2 = T_1 - Q \frac{X_1}{K_1} \quad \text{e} \quad T_3 = T_4 + Q \frac{X_3}{K_3}$$

A - EQUAÇÕES	B - ALGORITMO	C - TAREFAS	D - FLUXOGRAMA RESULTANTE
$Q = \frac{T_1 - T_4}{\frac{X_1}{K_1} + \frac{X_2}{K_2} + \frac{X_3}{K_3}}$	<p>B.1 - Fazer uma estimativa razoável para <math>T_2</math> e <math>T_3</math>, ou seja, fazer</p> $T_2 = T_4 + \frac{2}{3} (T_1 - T_4)$ $T_3 = T_4 + \frac{1}{3} (T_1 - T_4)$	<p>ATRIBUIÇÃO</p>	 <pre> graph TD     Start(( )) --&gt; B1[B.1]     B1 --&gt; B2[B.2]     B2 --&gt; B3[B.3]     B3 --&gt; B4[B.4]     B4 --&gt; B5[B.5]     B5 --&gt; B6{B.6}     B6 --&gt; Exit1[ ]     B6 --&gt; Exit2[ ]           </pre>
$K_1 = A_1 \bar{T}_1 + B_1$ $K_2 = A_2 \bar{T}_2 + B_2$ $K_3 = A_3 \bar{T}_3 + B_3$	<p>B.2 - Usar a eq. (0.3) para determinar valores de <math>T_i</math>.</p>	<p>ATRIBUIÇÃO</p>	
$\bar{T}_1 = (T_1 + T_2) / 2$ $\bar{T}_2 = (T_2 + T_3) / 2$ $\bar{T}_3 = (T_3 + T_4) / 2$	<p>B.3 - Usar a eq. (0.2) para determinar valores de <math>K_i</math>.</p>	<p>ATRIBUIÇÃO</p>	
$T_2 = T_1 - Q \frac{X_1}{K_1}$ $T_3 = T_1 + Q \frac{X_3}{K_3}$	<p>B.4 - Usar a eq. (0.1) para determinar o valor de <math>Q</math>.</p>	<p>ATRIBUIÇÃO</p>	
$T_3 = T_4 + Q \frac{X_3}{K_3}$	<p>B.5 - Considerando que os valores aproximados para <math>K_i</math> e <math>Q</math> já são conhecidos, recomputar <math>T_2</math> e <math>T_3</math>, ou seja: usar as equações (0.4) e (0.5) para calcular <math>T_2</math> e <math>T_3</math>.</p>	<p>ATRIBUIÇÃO</p>	
$T_3 = T_4 + Q \frac{X_3}{K_3}$	<p>B.6 - Repetir as etapas B.2 até a etapa B.5 até que dois valores sucessivos de <math>Q</math> sejam iguais.</p>	<p>ATRIBUIÇÃO + DECISÃO</p>	

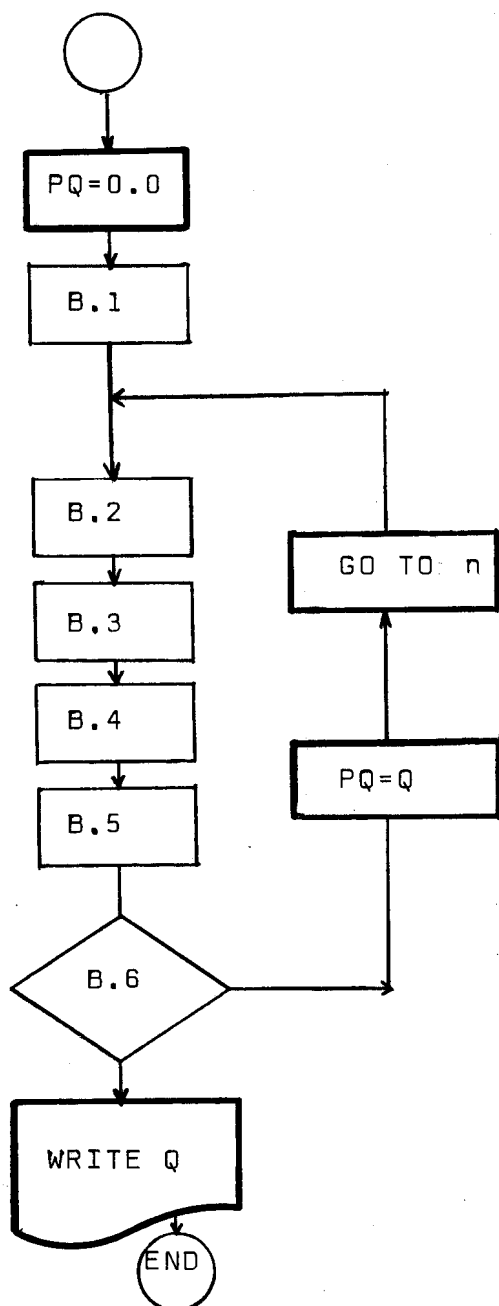


## D - COMENTÁRIOS

Observando o fluxograma gerado verificamos que falta muito ainda para que um programa possa ser codificado. Este programa apresenta inicialmente um critério de decisão baseado no valor de  $Q$ . Portanto, temos de admitir um valor prévio desta variável, que será  $PQ = 0.0$ .

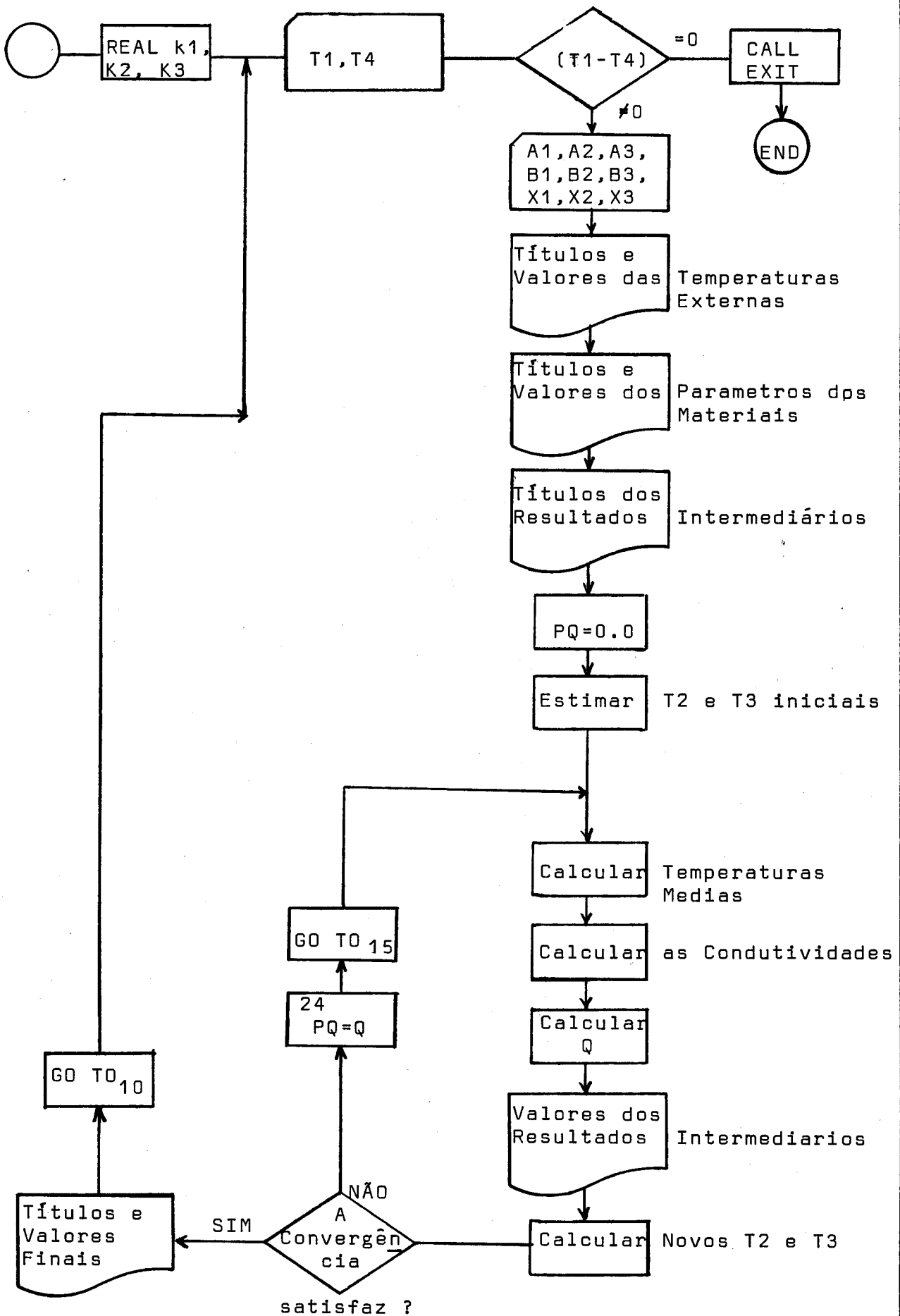
O critério de decisão nos sugere que façamos um teste de convergência. Ou seja, a diferença entre  $PQ$  e o valor instantâneo de  $Q$  é comparada com um valor bem pequeno, o qual indica a precisão desejada do resultado. Por exemplo:  $(PQ - Q) \leq 0.0005$ .

Se a diferença for menor ou igual que  $0.0005$ , o programa pode mandar imprimir os resultados finais. Porém, se tal diferença é maior que o valor de referência, o programa segue a lógica da etapa B.6, ordenando ao computador para que armazene em  $PQ$  o último valor de  $Q$ . Ou seja, temos uma atribuição nova:  $PQ = Q$  e processa-se uma nova iteração. Para tanto, uma declaração  $GO TO n$  favorece a localização da declaração que iniciará a nova iteração. E teremos um fluxograma novo, o qual apresenta o seguinte aspecto:



Aqui já podemos observar que um simples fluxograma pode ser alterado a partir da lógica do algoritmo e da lógica de programação. A introdução de novas declarações de atribuição permite introduzir novos critérios de decisão ou vice-versa.

Nas páginas seguintes apresentamos o fluxograma final e a Listagem original do programa elaborado pelos seus autores.



• THREE LAYER HEAT CONDUCTION

C

C

REAL K1,K2,K3

C

C

C

C

INPUT

10 READ (2,900) T1,T4

900 FORMAT (2E12.5)

IF (T1-T4) 12,30,12

12 READ (2,901) A1,B1,X1,A2,B2,X2,A3,B3,X3

901 FORMAT (2E12.5/E12.5)

WRITE (3,902) T1,T4

902 FORMAT ('EXTERIOR TEMPERATURES'/3X,'T1 =',F8.2,4X,'T4 =',F8.2)

WRITE (3,903) A1,B1,X1,A2,B2,X2,A3,B3,X3

903 FORMAT ('MATERIAL PARAMETERS'/3X,'A1 =',F7.4,4X,'B1 =',F8.5,

1 4X,'X1 =',F5.2/3X,'A2 =',F7.4,4X,'B2 =',F8.5,4X,'X2 =',

2 F5.2/3X,'A3 =',F7.4,4X,'B3 =',F8.5,4X,'X3 =',F5.2)

WRITE (3,904)

904 FORMAT ('INTERMEDIATE RESULTS'/6X,'Q',17X,'T1/M',14X,'T2/M',

1 14X,'T3/M')

C

C

INITIALIZE 'PREVIOUS' Q

C

PQ=0.0

C

C

INITIAL ESTIMATE OF T2 AND T3

C

T2=2.0\*(T1-T4)/3.0+T4

T3=(T1-T4)/3.0+T4

C

C

MEAN TEMPERATURES

C

15 T1M=0.5\*(T1+T2)

T2M=0.5\*(T2+T3)

T3M=0.5\*(T3+T4)

C

C

CONDUCTIVITY

C

K1=A1\*T1M+B1

K2=A2\*T2M+B2

K3=A3\*T3M+B3

C

C

HEAT QUANTITY

C

Q=(T1-T4) / (X1/K1+X2/K2+X3/K3)

C

C

PRINT INTERMEDIATE RESULTS

C

WRITE (3,905) Q,T1M,T2M,T3M

905 FORMAT (4E18.7)

C

C

NEW ESTIMATE OF T2 AND T3

C

T2=T1-Q\*X1/K1

T3=T4+Q\*X3/K3

C

C

TEST FOR CONVERGENCE

C

IF (ABS(PQ-Q)-0.0005) 26,26,24

C

C

SAVE 'PREVIOUS' Q

C

24 PQ=Q

GO TO 15

C

C

PRINT FINAL RESULTS

C

26 WRITE (3,906) Q,T2,T3

906 FORMAT ('FINAL RESULTS'/3X,'Q =',F9.2/3X,'T2 =',F8.2,

1 4X,'T3 =',F8.2)

GO TO 10

C

30 STOP

END

PROGRAM to be used in solving problem of Fig. 4 (three-layer heat conduction)—Fig. 5

EXTERIOR TEMPERATURES

T1 = 1400.00 T4 = 200.00

MATERIAL PARAMETERS

A1 = 0.0623 B1 = 0.00010 X1 = 2.15  
 A2 = 0.0255 B2 = 0.00005 X2 = 2.00  
 A3 = 2.4395 B3 = 0.00050 X3 = 6.00

INTERMEDIATE RESULTS

Q	T1/M	T2/M	T3/M
0.9026195E 04	0.1200000E 04	0.9000001E 03	0.4000000E 03
0.7931891E 04	0.1270209E 04	0.5979594E 03	0.2277501E 03
0.8355880E 04	0.1291569E 04	0.7345583E 03	0.2430990E 03
0.8316826E 04	0.1288353E 04	0.7306279E 03	0.2422747E 03
0.8318571E 04	0.1288611E 04	0.7303251E 03	0.2422152E 03
0.8318863E 04	0.1288608E 04	0.7303436E 03	0.2422350E 03
0.8318908E 04	0.1288605E 04	0.7303392E 03	0.2422325E 03
0.8318316E 04	0.1288606E 04	0.7308339E 03	0.2422325E 03
0.8318914E 04	0.1288606E 04	0.7308338E 03	0.2422325E 03
0.8318314E 04	0.1288606E 04	0.7308338E 03	0.2422325E 03

FINAL RESULTS

Q = 8318.81  
 T2 = 1177.21 T3 = 284.46

PRINTOUT for program of Fig. 5 (three-layer conduction)—Fig. 6

EXTERIOR TEMPERATURES

T1 = 1400.00 T4 = 200.00

MATERIAL PARAMETERS

A1 = 0.0623 B1 = 0.00010 X1 = 2.15  
 A2 = 0.0255 B2 = 0.00005 X2 = 2.00  
 A3 = 2.4395 B3 = 0.00060 X3 = 4.00

INTERMEDIATE RESULTS

Q	T1/M	T2/M	T3/M
0.9167527E 04	0.1200000E 04	0.9000001E 03	0.4000000E 03
0.8060339E 04	0.1269177E 04	0.6869559E 03	0.2187997E 03
0.8408079E 04	0.1290328E 04	0.7205319E 03	0.2302033E 03
0.8377233E 04	0.1287561E 04	0.7175054E 03	0.2293443E 03
0.8378197E 04	0.1287732E 04	0.7175004E 03	0.2293532E 03
0.8378357E 04	0.1287735E 04	0.7175163E 03	0.2293313E 03
0.8378334E 04	0.1287732E 04	0.7175131E 03	0.2293902E 03
0.8378335E 04	0.1287733E 04	0.7175134E 03	0.2293903E 03
0.8378335E 04	0.1287733E 04	0.7175134E 03	0.2293903E 03

FINAL RESULTS

Q = 8378.33  
 T2 = 1175.46 T3 = 259.76

PRINTOUT for program of Fig. 5, using a different set of parameters—Fig. 7

EXTERIOR TEMPERATURES

T1 = 1400.00 T4 = 200.00

MATERIAL PARAMETERS

A1 = 0.0623 B1 = 0.00010 X1 = 4.30  
 A2 = 0.0255 B2 = 0.00005 X2 = 0.00  
 A3 = 2.4395 B3 = 0.00060 X3 = 6.00

INTERMEDIATE RESULTS

Q	T1/M	T2/M	T3/M
0.1884833E 05	0.1200000E 04	0.8000001E 03	0.4000000E 03
0.1333575E 05	0.8579473E 03	0.3158945E 03	0.2579473E 03
0.1344456E 05	0.8635790E 03	0.3271559E 03	0.2635780E 03
0.1342816E 05	0.8627274E 03	0.3254549E 03	0.2627274E 03
0.1343059E 05	0.8628536E 03	0.3257073E 03	0.2628537E 03
0.1343023E 05	0.8628349E 03	0.3256698E 03	0.2628349E 03
0.1343029E 05	0.8628377E 03	0.3256755E 03	0.2628377E 03
0.1343028E 05	0.8628372E 03	0.3256746E 03	0.2628373E 03
0.1343028E 05	0.8628374E 03	0.3256747E 03	0.2628374E 03
0.1343028E 05	0.8628374E 03	0.3256747E 03	0.2628374E 03

FINAL RESULTS

Q = 13430.28  
 T2 = 325.67 T3 = 325.67

ANOTHER PRINTOUT for program of Fig. 5, using other input values—Fig. 8

## 2. FUNÇÕES E REPRESENTAÇÃO DE DADOS

Muitas funções são usadas na Engenharia Química, podendo ser baseadas tanto em definições fundamentais como em teorias. Por exemplo: a massa específica,  $D=M/V$ , é uma função baseada numa definição fundamental, enquanto que a equação dos gases perfeitos,  $PV=nRT$ , é oriunda de uma teoria.

Em certos casos, quantidades negativas, como volume negativo ou massa negativa, não possuem significado físico. Torna-se necessário conhecer, sempre que possível, os intervalos de significado físico das variáveis envolvidas pela função para que não sejam obtidos resultados incompatíveis com a realidade.

As funções são representadas por três modos: relações algébricas, forma gráfica e forma tabular. A maneira particular na qual a função é expressada depende do processo pelo qual ela foi desenvolvida e de como será empregada. Frequentemente torna-se necessário mudar de uma representação funcional para outra. Por exemplo: da forma tabelada de dados / experimentais para uma forma gráfica. Um computador pode ser usado nestas transformações através do equipamento traçador de gráficos (plotador), mais conhecido como PLOTTER.

Podemos transformar os dados da forma tabelada para a gráfica, ou da tabelada para a algébrica.

A transformação de dados tabelados numa expressão algébrica é denominada AJUSTAMENTO DE CURVAS.

O processo de conversão de valores da forma algébrica de uma função para uma forma tabelada envolve a procura da função algébrica desejada, a determinação dos valores das constantes, e, então, a determinação dos valores da variável dependente que correspondem aos valores das variáveis independentes do programa.

O sub-programa FUNCTION é conveniente quando necessitamos de rotinas genéricas, destinadas a produzir apenas um resultado cada vez que são utilizadas. Ou seja, calcular apenas um resultado da variável dependente para um valor da variável independente, ou um conjunto de valores das variáveis independentes do processo.

## CÁLCULO DO pH

Escrever um programa para ilustrar a relação entre o pH de uma solução aquosa de um ácido forte e as propriedades ácido-base da água. Isto pode ser efetuado calculando-se os valores de pH de soluções de ácido cuja diluição é crescente. Para uma solução de HCl, a concentração de ion hidrônio é determinada pelas seguintes considerações.

$$\text{Balanço das Cargas: } (H_3O^+) = (Cl^-) + (OH^-)$$

$$\text{Produto Iônico da Água: } K_w = (OH^-) (H_3O^+)$$

Podemos usar estas relações para estabelecer uma equação quadrática em termos de  $(H_3O^+)$

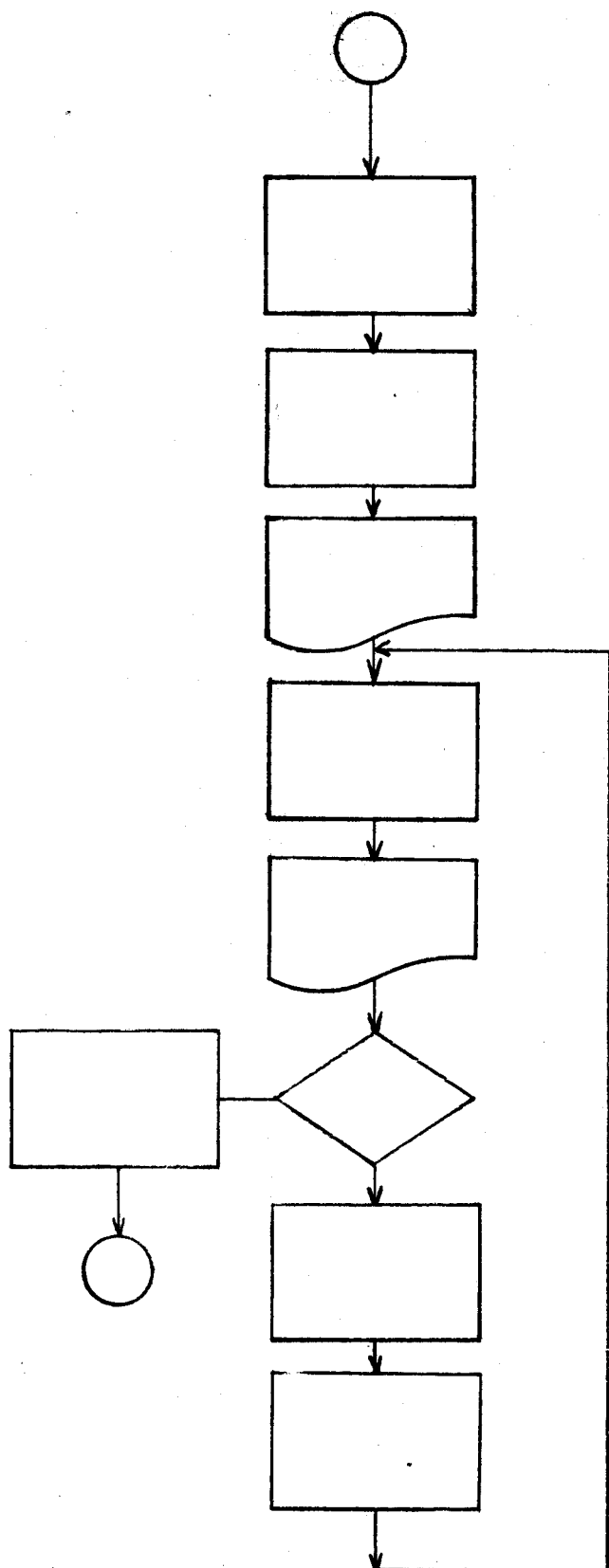
$$(H_3O^+) = (Cl^-) + \frac{K_w}{(H_3O^+)} \text{ ou } (H_3O^+)^2 - (Cl^-) (H_3O^+) - K_w = 0$$

O valor de  $(Cl^-)$  é igual ao da molaridade do ácido e  $K_w$  é uma constante conhecida. Considerando-se a equação do segundo grau obtida, temos que o valor de  $(H_3O^+)$  pode ser calculado pela expressão.

$$x = \frac{-b + (b^2 - 4ac)^{1/2}}{2a}$$

Para ilustrar a variação do pH com a concentração de ácido, esta concentração deverá sofrer incrementos a partir de 1 M até a molaridade na qual o pH é constante para oito dígitos, isto é, precisão de oito casas decimais. Na realidade, o pH variará continuamente à medida que a concentração varie e atingirá o valor 7 à medida que a mesma diminua. É claro que soluções de ácido extremamente diluídas não representam, em termos práticos soluções reais, mas nós as consideraremos para mostrar que o pH nunca excederá o valor 7.

1. FLUXOGRAMA



Definir a função aritmética para  $(H_3O^+)$

Atribua valores iniciais para as variáveis

Escrever títulos para os resultados

Calcular  $(H_3O^+)$  e o pH usando a concentração de ácido gerada pelo programa

Escrever as concentrações  $(H_3O^+)$  e o pH

Verificar se a diferença entre os valores iniciais e calculado é menor que a precisão

Trocar o valor do pH pelo atual

Diminuir a concentração do ácido

Repetir os cálculos com os mesmos valores.

## 2. PROGRAMA

B6700/B7700 F O R T R A N C O M P I L A T I O N M A R K 2.

```

FILE 8=CARTOES,UNIT=READER
FILE 5=SAIDA,UNIT=PRINTER
C
C
C   CALCULO DO PH DE SOLUCOES DE HCL
C
C
C   FUNCAO PARA O CALCULO DA CONCENTRACAO DE ION HIDRONIO
C
FUNCTION AZ(A,FKW)
AZ=(A+(A**2+4.*FKW)**.5)/2.0
RETURN
END

C
C
C   INICIACAO DE VARIAVEIS
CA=1.0
PHS=0.9
FKW=1.0E-14
C   IMPRESSAO DE TITULOS
WRITE(5,1)
1  FORMAT(1H,'PH DA SOLUCAO DE HCL',//,1H,'MCA',20X,'H',20X,'PH')
C   LOOP DE CALCULOS (CIRCUITO DE REPETICAO DE CALCULOS)
5  H=AZ(CA,FKW)
   PH=-ALOG(H)/2.303
C   IMPRESSAO DOS RESULTADOS CALCULADOS
   WRITE(5,2) CA,H,PH
2  FORMAT(1H,'E14.7,6X,E14.7,6X,E14.7)
C   COMPARACAO ENTRE OS VALORES ATUAL E ANTERIOR
   IF(ABS(PHS-PH)-1.0E-8)20,20,10
C   ARMAZENAGEM DO VALOR ATUAL DO PH
10 PHS=PH
C   DILUICAO DA CONCENTRACAO DE ACIDO
   CA=CA/10.0
C   REINICIO DOS CALCULOS
   GO TO 5
C   INDICACAO DO TERMINO DOS CALCULOS
20 WRITE(5,30)
30 FORMAT(1H,'FIM DOS CALCULOS')
   CALL EXIT
END

```

## 3. COMENTÁRIOS

Estas diversas declarações "comentário" foram empregadas com propósitos de esclarecimento. Normalmente não se empregam tantas.

Os valores iniciais de CA e KW são definidos. FKW é usado para flutuar a variável, transformando-a em real. O valor inicial de PHS é arbitrário e foi empregado apenas para respeitar o funcionamento do computador. O fator 2.303 é necessário para converter o logaritmo neperiano para decimal.

O valor absoluto da diferença entre os valores de pH é necessário para evitar uma diferença negativa, que não teria nenhum sentido prático.

Este método de tentativas e erros é conhecido como "IF loop" ou "calculation loop". Nele não são necessários cartões de dados.



PH DA SOLUCAO DE HCL

HCA	H	PH
.1000000E+01	.1000000E+01	0,
.1000000E+00	.1000000E+00	.9998198E+00
.1000000E-01	.1000000E-01	.1999640E+01
.1000000E-02	.1000000E-02	.2999460E+01
.1000000E-03	.1000001E-03	.3999279E+01
.1000000E-04	.1000100E-04	.4999056E+01
.1000000E-05	.1009902E-05	.5994641E+01
.1000000E-06	.1618034E-06	.6789789E+01
.1000000E-07	.1051249E-06	.6977037E+01
.1000000E-08	.1005012E-06	.6996568E+01
.1000000E-09	.1000500E-06	.6998522E+01
.1000000E-10	.1000050E-06	.6998717E+01
.1000000E-11	.1000005E-06	.6998737E+01
.1000000E-12	.1000001E-06	.6998739E+01
.1000000E-13	.1000000E-06	.6998739E+01
.1000000E-14	.1000000E-06	.6998739E+01
.1000000E-15	.1000000E-06	.6998739E+01

FIM DOS CALCULOS

LEITURA DE FUNÇÕES E SUBROTINAS

A disposição dos cartões perfurados, que envolvem funções e subrotinas, para leitura nos sistemas dos computadores IBM-1130 e B-6700, apresenta apenas uma diferença: no primeiro caso são necessários cartões de controle para cada função ou subrotina; no segundo tais cartões são desnecessários.

As sequências de cartões são apresentadas nas páginas seguintes.



### 3. ERROS NUMÉRICOS

Vários tipos de erros podem ocorrer durante os cálculos executados por um computador. O alcance ou repercussão dos erros pode ser grande se muitos cálculos repetidos estiverem envolvidos. Os dois tipos mais comuns de erros são os de arredondamento e os de truncamento.

Os erros de arredondamento aparecem como resultado do número fixo de dígitos que pode constituir um número armazenado, isto é, uma mantissa de oito dígitos. A manipulação aritmética repetida pode provocar uma perda de dígitos significativos. Isto ocorre porque nenhum dos resultados intermediários é arredondado.

Frequentemente os cálculos envolvem o cálculo de funções, tais como logaritmos, nos quais os resultados são apenas aproximações finitas de uma série. De fato, as séries infinitas são usadas para calcular muitas das funções na biblioteca de sub-rotinas. Desde que o número de dígitos é fixado, as séries podem ser desenvolvidas somente até esta quantidade de dígitos e, então, serem truncadas. Se várias aproximações desta espécie são feitas, a acumulação de erros pode ocorrer.

Muitos dos métodos numéricos empregados para propósitos de cálculos utilizam aproximações sucessivas, e dão, portanto, somente resultados aproximados. Por exemplo, a determinação de raízes por iteração, ou o cálculo de uma integral definida pela Regra de Simpson, oferecem resultados desta natureza.

Ocasionalmente, a perda de elementos significativos pode ser evitada por uma programação adequada das etapas de cálculo.

Uma evidência sobre erros numéricos causados pelo cálculo de funções, como as exponenciais, é apresentada no próximo exemplo. Neste, os dados foram introduzidos, em princípio, no formato real. A listagem indicou determinados valores. Então, foram introduzidos os mesmos dados no formato exponencial (novos cartões de dados foram perfurados). Os resultados foram diferentes e mais próximos da realidade. Explicação: a conversão de dados lidos no formato real para a formatação exponencial gera erros decorrentes da precisão interna do computador. Estes erros se propagaram no cálculo das exponenciais e das demais operações de programa, gerando resultados distorcidos.

Portanto, uma regra de bom-senso, para a manipulação de dados científicos, é introduzir os valores no formato exponencial. Complementando esta regra, a listagem dos resultados pode também ser apresentada na formatação exponencial.

# PRESSÃO DE UM GÁS

Considere a variação da pressão de um número fixo de moles de um gás sob temperatura constante quando o volume varia. Como função algébrica adequada, podemos usar a equação de estado de um gás de Dieterici.

$$\left(\frac{an}{VRT}\right)$$

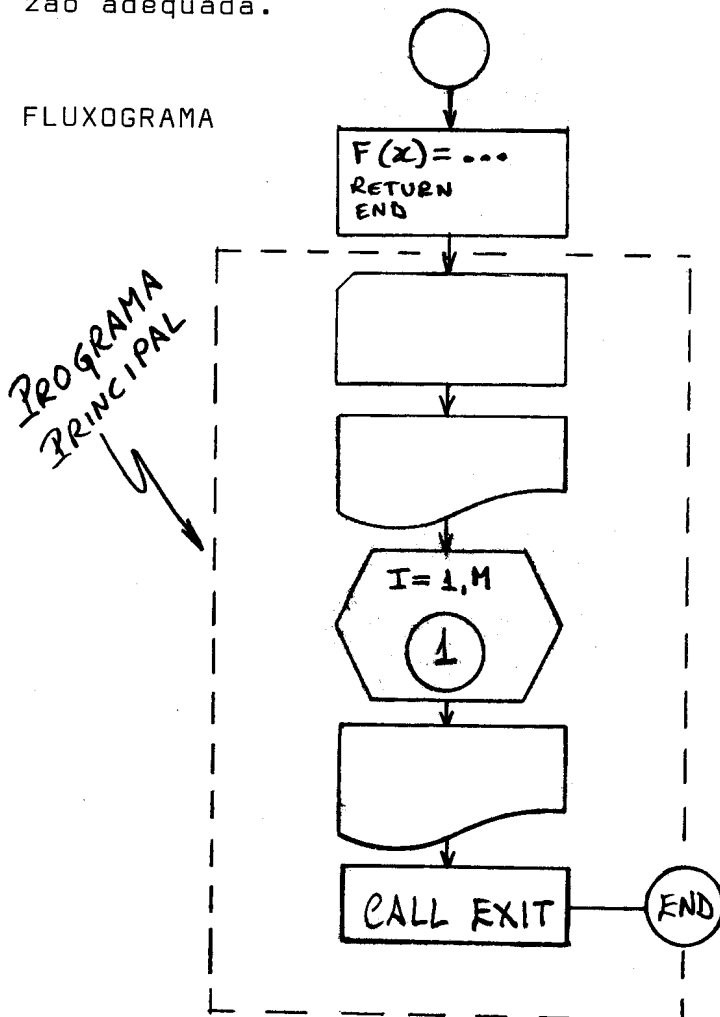
$$P. e \quad (V - nb) = nRT$$

Aqui P é a pressão em atmosferas, V é o volume em litros, T é a temperatura em graus Kelvin, R é a constante dos gases com unidades adequadas, n é o número de moles do gás, e a e b são constantes empíricas que dependem do gás específico em consideração.

$$P = \frac{nRT e^{(-an / VRT)}}{V - nb}$$

Podemos, então, calcular P como uma função de V para um número específico de moles, numa temperatura constante, de um dado gás. Os valores da pressão podem ser calculados para vários volumes, iniciando-se de um valor desejado, incrementando-se com a razão adequada.

FLUXOGRAMA



Defina a função sub programa

Leia o nome do gás, os valores das constantes, os valores iniciais e finais, e o parâmetro de incremento da variável.

Escreva títulos e nomes.

Utilize o comando DO para obter os diferentes valores de P.

Escreva os resultados.



// FOR

```
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
C CALCULO DE 'P' EM FUNCAO DE 'V'
  FUNCTION P(X,R,B,A,T, FN)
  P=(FN*R*T*EXP((-A*FN)/(X*R*T)))/(X-FN*B)
  RETURN
END
```

1  
2  
3  
4

0 ERR(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTAO OCUPADAS 85 DAS 8042 PALAVRAS DISPONIVEIS

// CUP

\*STORE WS UA P  
 K 21 \* OPERACAO COMPLETADA

PAG. 2

\* COPPE-FORTRAN \* 23/11/76

// FOR

```
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*ICCS(2501RADER,1403PRINTER)
C
C PROGRAMA PRINCIPAL
DIMENSION PRESS(100)
READ(P,5)
5 FORMAT(1H,25H )
READ(P,10)FN,R,T,A,B
10 FORMAT(5(1X,F14.7))
READ(P,20)M,DELV,V
20 FORMAT(13,2(1X,F14.7))
WRITE(5,10)FN,R,T,A,B
WRITE(5,20)M,DELV,V
WRITE(5,5)
WRITE(5,6)FN,T
6 FORMAT(1H,'NUMERO DE MOLES = ',F14.7,3X,'TEMPERATURA = ',F14.7)
CO 1 T=1,M
PRESS(1)=P(V,R,B,A,T, FN)
WRITE(5,30)PRESS(1),V
1 V=V+DELV
30 FORMAT(1H,' PRESS = ',E11.4,2X,' V = ',E11.4)
CALL EXIT
END
```

5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23

0 ERR(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTAO OCUPADAS 523 DAS 8042 PALAVRAS DISPONIVEIS

// XEC

1.0000000	0.0820300	289.0000000	3.6000000	0.0428000
30	1.0000000	1.0000000		
BICXICC DE CARBONO				
NUMERO DE MOLES =	1.0000000	TEMPERATURA =	289.0000000	
PRESS =	2.1277E+01	V =	1.0000E+00	
PRESS =	1.1226E+01	V =	2.0000E+00	
PRESS =	7.6205E+00	V =	3.0000E+00	

PAG. 3

\* COPPE-FORTRAN \* 23/11/76

PRESS	=	5.7675E+00	V	=	4.0000E+00
PRESS	=	4.6252E+00	V	=	5.0000E+00
PRESS	=	3.8000E+00	V	=	6.0000E+00
PRESS	=	3.3443E+00	V	=	7.0000E+00
PRESS	=	2.9232E+00	V	=	8.0000E+00
PRESS	=	2.6223E+00	V	=	9.0000E+00
PRESS	=	2.3449E+00	V	=	1.0000E+01
PRESS	=	2.1133E+00	V	=	1.1000E+01
PRESS	=	1.9176E+00	V	=	1.2000E+01
PRESS	=	1.8083E+00	V	=	1.3000E+01
PRESS	=	1.6802E+00	V	=	1.4000E+01
PRESS	=	1.5690E+00	V	=	1.5000E+01
PRESS	=	1.4716E+00	V	=	1.6000E+01
PRESS	=	1.3855E+00	V	=	1.7000E+01
PRESS	=	1.3000E+00	V	=	1.8000E+01
PRESS	=	1.2405E+00	V	=	1.9000E+01
PRESS	=	1.1768E+00	V	=	2.0000E+01
PRESS	=	1.1230E+00	V	=	2.1000E+01
PRESS	=	1.0000E+00	V	=	2.2000E+01
PRESS	=	9.0000E-01	V	=	2.3000E+01
PRESS	=	8.0000E-01	V	=	2.4000E+01
PRESS	=	7.0000E-01	V	=	2.5000E+01
PRESS	=	6.0000E-01	V	=	2.6000E+01
PRESS	=	5.0000E-01	V	=	2.7000E+01
PRESS	=	4.2377E-01	V	=	2.8000E+01
PRESS	=	3.1440E-01	V	=	2.9000E+01
PRESS	=	7.8735E-01	V	=	3.0000E+01

FIM DA EXECUCAO  
 FCRAV EXECUTACAS 159 INSTRUCOES

Observe que este é o mesmo programa, porém, testado em outro computador, com precisão interna diferente.

// FCR

```

*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
C
C
C
C
C
C
CALCULO DE 'P' EM FUNCAO DE 'V'
C
C
FUNCTION P(X,R,B,A,T,FN)
P=(FN*R*T*EXP((-A*FN)/(X*R*T)))/(X-FN*B)
RETURN
END

```

0 ERR(C) E C ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTADO OCUPADAS 85 DAS 8042 PALAVRAS DISPONIVEIS

PAG. 2 \* COPPE-FORTRAN \* 24/11/76

// DUP

\*SICRE WS UA P  
 K 21 \* OPERACAO COMPLETADA

// FCR

```

5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*ICCS(25CIR*ADER,1403PRINTER)
C PROGRAMA PRINCIPAL
DIMENSION PRESS(100)
REAC(5,5)
5 FORMAT(1H,25H
REAC(5,10)FN,R,T,A,B
10 FORMAT(5(1X,E14.7))
REAC(5,20)M,DELV,V
20 FORMAT(13,2(1X,E14.7))
WRITE(5,10)FN,R,T,A,B
WRITE(5,20)M,DELV,V
WRITE(5,5)
WRITE(5,6)FN,T
6 FORMAT(1H,'NUMERO DE MOLES = ',F14.7,3X,'TEMPERATURA = ',F14.7)
DO 1 I=1,M
PRESS(I)=P(V,R,B,A,T,FN)
WRITE(5,30)PRESS(I),V
1 V=V*DELV
30 FORMAT(1H,' PRESS = ',E11.4,2X,' V = ',E11.4)
CALL EXIT
END

```

0 ERR(C) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTADO OCUPADAS 523 DAS 8042 PALAVRAS DISPONIVEIS

// XEC  
 1.000000E+00 8.203000E-02 7.980000E+02 3.600000E+00 4.280000E-02  
 3C 1.000000E+00 1.000000E+00  
 EICXIC DE CARENCO

PAG. 3 \* COPPE-FORTRAN \* 24/11/76

NUMERO DE	MOLES =	TEMPERATURA =
FRESSS	2.2040E+01	1.0000E+00
FRESSS	1.1603E+01	2.0000E+00
FRESSS	7.8702E+00	3.0000E+00
FRESSS	5.9540E+00	4.0000E+00
FRESSS	4.7880E+00	5.0000E+00
FRESSS	4.0035E+00	6.0000E+00
FRESSS	3.4404E+00	7.0000E+00
FRESSS	3.0160E+00	8.0000E+00
FRESSS	2.6647E+00	9.0000E+00
FRESSS	2.4151E+00	1.0000E+01
FRESSS	2.2012E+00	1.1000E+01
FRESSS	2.0194E+00	1.2000E+01
FRESSS	1.8653E+00	1.3000E+01
FRESSS	1.7330E+00	1.4000E+01
FRESSS	1.6183E+00	1.5000E+01
FRESSS	1.5178E+00	1.6000E+01
FRESSS	1.4261E+00	1.7000E+01
FRESSS	1.3350E+00	1.8000E+01
FRESSS	1.2395E+00	1.9000E+01
FRESSS	1.1258E+00	2.0000E+01
FRESSS	1.1158E+00	2.1000E+01
FRESSS	1.1058E+00	2.2000E+01
FRESSS	1.0958E+00	2.3000E+01
FRESSS	1.0858E+00	2.4000E+01
FRESSS	9.7272E-01	2.5000E+01
FRESSS	9.3342E-01	2.6000E+01
FRESSS	9.0018E-01	2.7000E+01
FRESSS	8.6578E-01	2.8000E+01
FRESSS	8.2989E-01	2.9000E+01
FRESSS	8.1155E-01	3.0000E+01

Observe que a formatacao usada para a leitura de dados foi a exponencial.

FIM DA EXECUCAO  
 FCRAM EXECUTADAS 159 INSTRUCOES

#### 4. FRAÇÕES DE SOLUTO EM EXTRAÇÃO CONTRACORRENTE

O método de extração contracorrente é um processo que isola eficientemente um soluto através de extrações líquido-líquido repetidas. O método é usado para separação e purificação de solutos numa solução. Dois solventes imiscíveis, nos quais os solutos se solubilizam, são envolvidos. São utilizados dois conjuntos de tanques especialmente projetados para este método.

Podemos considerar um dos conjuntos de tanques contendo volumes iguais de líquido O e outro conjunto de volumes iguais de líquido W. Estas fases líquidas podem ser denominadas de

$$O_0, O_1, O_2, \dots, O_i$$

$$W_0, W_1, W_2, \dots, W_i$$

O soluto a ser extraído está em solução na fase  $W_0$ .

O processo de transferência é repetido n vezes. Em cada etapa, as fases O são misturadas com a próxima fase W e uma nova fase O é introduzida. O procedimento pode ser ilustrado da seguinte forma:

$$O_i \dots O_3 \quad O_2 \quad O_1 \quad O_0$$

$$W_0 \quad W_1 \quad W_2 \quad W_3 \dots W_i$$

Nesta primeira etapa as fases  $O_0$  e  $W_0$  são misturadas.

A primeira extração conduz a:

$$O_i \dots O_3 \quad O_2 \quad O_1 \quad O_0$$

$$W_0 \quad W_1 \quad W_2 \dots W_i$$

Após o soluto atingir o equilíbrio entre as fases,  $O_0$  é então misturada com a fase  $W_1$  e a fase  $O_1$  é misturada com a fase  $W_0$ .

A segunda extração conduz a:

$$O_i \dots O_3 \quad O_2 \quad O_1 \quad O_0$$

$$W_0 \quad W_1 \quad W_2 \dots W_i$$

Cada vez que as fases O são misturadas com a próxima fase W, uma nova fase O é introduzida no sistema.



A medida que se desenvolve o processo de extração contracorrente, o soluto irá tornar-se disperso em todas as fases. Na maioria dos casos, o soluto estará presente em concentrações maiores nas fases centrais. A distribuição do soluto depende das solubilidades relativas nos solventes. Estas são expressadas em termos de coeficiente de Partição D, que é definido para o soluto x nos solventes W e O:

$$D = \frac{\text{Concentração de x em O}}{\text{Concentração de x em W}} = \frac{(X_o)}{(X_w)}$$

As fases do soluto no equipamento após  $n$  extrações são dadas pelos termos da expansão binomial:

$$\left( \frac{1}{1+D} + \frac{D}{1+D} \right)^n$$

Cada termo pode ser expressado como a fração  $f$  do soluto no tanque  $i$ , após  $n$  extrações e pode ser calculado por:

$$f_{i,n} = \frac{(n!) (D^i)}{i! (n-i)! (1+D)^n}$$

O termo  $n!$  e outros seguidos por  $!$  referem-se aos fatoriais destes termos. O fatorial de um número é dado pela expressão:

$$n! = (n) (n-1) (n-2) \dots$$

A fim de calcularmos as frações de soluto em cada tanque, após  $n$  extrações, um programa utilizando um subprograma FUNCTION, para obter fatoriais de números, ou expressões, pode ser escrito. Assim, teríamos:

// FCR

```

*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
1      FUNCTION FACTO(NUM)
2      IF(NUM)1,2,3
3      FACTO=0.0
4      RETURN
5      FACTO=1.0
6      RETURN
7      FACTO=1.0
8      DO 4 I=1,NUM
9      F=FLOAT(I)
10     FACTO=FACTO*F
11     RETURN
12     END

```

0 ERRC(S) E

0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO

C CALCULO DAS FRACOES DE SOLUTO

```

13      READ(*,10)N,D
14      WRITE(14,2X,F6.3)
15      WRITE(5,2C)D,N
16      FORMAT(1H,10)A DISTRIBUTUICAC DE SOLUTO COM COEFICIENTE
17      DE PARTICACAO , F7.3, / , APOS ',I4,
18      CONST=FACTO(N)/(1.+D)*N
19      DO 30 I=1,N
20      FIN=(TONST#D*I)/FACTO(I)/FACTO(N-I)
21      WRITE(5,40)I,FIN
22      FORMAT(1H,10)TANQUE ',I4,5X,E14.7)
23      CALL EXIT
      END

```

O ERRC(S) E O ADVERTENCIA(S) ASSINALADOS NESTA CCMPIACAO  
 FIM DA CCMPIACAO \* ESTAO OCUPADAS 286 DAS 8042 PALAVRAS DISPONIVEIS  
 // XEC

A DISTRIBUICAC DE SOLUTO EXTRA	SOLUTO	COEFFICIENTE	DE PARTICACAO	1.000
TANQUE	1	7.450600E-06		
TANQUE	2	8.940600E-05		
TANQUE	3	8.545320E-04		
TANQUE	4	3.7699920E-03		
TANQUE	5	1.5877980E-02		
TANQUE	6	1.22323430E-02		
TANQUE	7	3.07416560E-02		
TANQUE	8	1.32285350E-01		
TANQUE	9	6.74184070E-01		
TANQUE	10	1.5458080E-01		
TANQUE	11	1.32284070E-01		
TANQUE	12	1.97416560E-02		
TANQUE	13	6.09833420E-02		
TANQUE	14	1.42779880E-03		
TANQUE	15	1.5833560E-03		
TANQUE	16	1.7699910E-04		
TANQUE	17	8.85406940E-06		
TANQUE	18	7.450600E-06		
TANQUE	19	8.940600E-05		
TANQUE	20	8.545320E-04		
TANQUE	21	3.7699920E-03		
TANQUE	22	1.5877980E-02		
TANQUE	23	1.22323430E-02		
TANQUE	24	3.07416560E-02		
TANQUE	25	1.32285350E-01		

O sub-programa SUBROUTINE é semelhante ao sub-programa FUNCTION, porém é mais útil porque pode fornecer um ou mais resultados ao programa principal. A lista de variáveis operadas pela SUBROUTINE conterá as de entrada de dados e de saídas de resultados, permitindo que possam ser tanto reais como inteiras. No caso de FUNCTION o próprio nome indica a variável calculada e no caso de SUBROUTINE o nome pode ser qualquer um.

5. UMIDADE DE UMA SUBSTÂNCIA

O comando DO é muito útil para efetuar certos tipos de cálculos repetidos e certas operações de entrada e saída de valores. Por exemplo, consideremos a rotina para o cálculo da percentagem de massa de água num hidrato. Tal programa poderia inicialmente ser escrito da seguinte forma:

```

C
C
C          CALCULO DA UMIDADE DE UMA SUBSTANCIA
1          READ (8,50)HYD1,ANH1,HYD2,ANH2,HYD3,ANH3
2          50  FORMAT(6F8.4)
3          PH2C1=((HYD1-ANH1)/HYD1)*100.
4          PH2C2=((HYD2-ANH2)/HYD2)*100.
5          PH2C3=((HYD3-ANH3)/HYD3)*100.
6          AVEPC=(PH2C1+PH2C2+PH2C3)/3.0
7          WRITE(5,6C)PH2C1,PH2C2,PH2C3,AVEPC
8          60  FORMAT(1H,3(2X,F10.3),/,F10.3)
9          CALL EXIT
10         END

```

PAG. 2 \* COPPE-FORTRAN \* 25/11/76

O ERRO(S) E O ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

FIM DA COMPILACAO \* ESTAO OCUPADAS 123 DAS 8042 PALAVRAS DISPONI

// XEC

10.841	16.395	16.676
14.637		

FIM DA EXECUCAO

```

C
C          CALCULO DA UMIDADE DE UMA SUBSTANCIA - EMPREGO DO COMANDO DO
C
C
C          TODA VEZ QUE FOREM EMPREGADAS VARIÁVEIS SUBSCRITAS,
C          DEVEMOS DIMENSIONÁ-LAS.
C

```

```

10  DIMENSION PH2C(20),HYD(20),ANH(20)
10  READ (8,10)N,(HYD(I),ANH(I),I=1,N)
10  FORMAT(1Z,/,2(1X,F6.4))
    SUM=0.0
    FN=FLOAT(N)
    DO 1 I=1,N
    PH2C(I)=((HYD(I)-ANH(I))/HYD(I))*100.
    SUM=SUM+PH2C(I)
1   CONTINUE
    AVEPC=SUM/FN
    WRITE(5,20)
    WRITE(5,30)(I,PH2C(I),I=1,N)
    WRITE(5,40)AVEPC
20  FORMAT(1H,*,PERCENTAGEM DE AGUA NA AMOSTRA *)
30  FORMAT(1H,*,I3,2X,F10.3,3X,*,POR CENTO*)
40  FORMAT(1H,*,VALOR MEDIO = *,2X,F10.3,*,POR CENTO*)
    CALL EXIT
    END

```

// XEC

PERCENTAGEM DE	AGUA NA AMOSTRA
1	10.841 POR CENTO
2	16.395 POR CENTO
3	16.676 POR CENTO
VALOR MEDIO =	14.637 POR CENTO

Para compreendermos melhor o funcionamento desta rotina, consideremos cada etapa em separado:

1. A declaração DIMENSION define as variáveis subscritas e estabelece os valores máximos para os subscritos.
2. A declaração READ é um exemplo de declaração de entrada que incorpora um comando DO (implícito). Este comando é equivalente a:

```
      READ(8,5)N
      5 FORMAT(I3)
      DO 6 I = 1,N
      6 READ(8,9) HYD(I),ANH(I)
      9 FORMAT(F6.4,2X,F6.4)
```

3. A variável SUM é usada no circuito do comando DO para referir-se à soma das percentagens. Deve, pois, iniciar-se com o valor zero.
4. O valor final de AVEPC dará o valor médio das percentagens. A declaração FN = FLOAT(N) foi introduzida para compatibilizar os dados gerados pelo comando DO e aqueles executados pelas declarações posteriores.
5. São calculados N valores de PH20(I), SUM e AVEPC. Em nosso caso, apenas os últimos valores interessam.
6. A listagem dos resultados pode oferecer os aspectos sugeridos pelas tres declarações WRITE. As declarações FORMAT podem estar localizadas fora do circuito do comando DO, pois estão devidamente referenciadas.
7. O dimensionamento de uma variável não requer necessariamente que todas as posições da memória sejam usadas. No exemplo abaixo, o dimensionamento está exagerado.
8. Ainda no exemplo abaixo, poderíamos ter incluído, na listagem, a impressão do valor de I. Tal informação é geralmente desnecessária ao proprio programador.

### CALCULO DA MASSA ESPECIFICA

```
DIMENSION XMASS(50),VOL(50),DEN(50),TEMP(50)
      READ(8,10)(XMASS(I),VOL(I),TEMP(I),I=1,10)
10  FORMAT(1X,3E14.7)
      DO 30 I=1,10
20  DEN(I)=XMASS(I)/VOL(I)
      WRITE(5,20)(DEN(I),TEMP(I),I=1,10)
20  FORMAT(1H, ' MASSA ESPECIFICA = ',1X,E14.7, ' NA TEMPERATURA DE ',1
1X,F5.0,1X, ' GRAUS CENTIGRADOS ')
      CALL EXIT
      END
```

CALCULO DA UNIDADE DE UMA SUBSTANCIA - EMPREGO DO COMANDO DO

TODA VEZ QUE FOREM EMPREGADAS VARIÁVEIS SUBSCRITAS, DEVEMOS DIMENSIONÁ-LAS.

```

COMANDO DO EXPLÍCITO
10 DIMENSION PH20(20),HYD(20),ANH(20)
11 READ(P,10)N,(HYD(I),ANH(I),I=1,N)
12 FORMAT(I2,/, (1X,F6.4,1X,F6.4))
13 SUM=0.0
14 FN=FLTAI(N)
15 DO 1 I=1,N
16 PH20(I)=((HYD(I))-ANH(I))/HYD(I))*100.
17 SUM=SUM+PH20(I)
18 AVEPC=SUM/FN
19 WRITE(5,20)
20 WRITE(5,30)AVEPC
21 FORMAT(IH,*, PERCENTAGEM DE AGUA NA AMOSTRA *)
22 FORMAT(IH,*,13,2X,F10.3,*,POR CENTO*)
23 FORMAT(IH,*,VALOR MEDIO =,2X,F10.3,*,POR CENTO *)
24 CONTINUE
25 CALL EXIT
26 END
COMANDO DO IMPLÍCITO

```

0 ERRC(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO  
 FIM DA COMPILAÇÃO \* ESTÃO OCUPADAS 370 DAS 8042 PALAVRAS DISPONÍVEIS

// XEQ  
 PERCENTAGEM DE AGUA NA AMOSTRA  
 1 10.841 POR CENTO

X 10 \* VARIÁVEL REAL INDEFINIDA  
 X 01 \* LINHA = 11 \* ENDECERCO = 780E

2

X 03 \* EXECUÇÃO INTERROMPIDA 9 INSTRUÇÕES  
 FCRAM EXECUTADAS

A troca de posição da declaração 1 CONTINUE gerou um comando DO embutido que conduziu a erro, porque a declaração WRITE que o comando tinha mandou o computador escrever valores ainda não lidos e calculados.

CCCCCCCC

6. MEDIA E DESVIO PADRÃO

A obtenção de informações estatísticas envolve, frequentemente, o calculo da média e do desvio padrão. Para um conjunto de dados experimentais podemos utilizar o programa abaixo, que emprega, para o calculo do desvio padrão, a expressão apresentada nas páginas iniciais deste texto. Existe outra expressão que nos fornece, evidentemente, valores levemente alterados:

$$S = \frac{1}{n} \sum_{i=1}^n [(x_i - x_m)^2]^{1/2}$$

onde  $x_i$  é cada um dos valores lidos e  $x_m$  sua média aritmética.

```

1234567890111213141516171819
CCCCC
CALCULO DA MEDIA E DO DESVIO PADRAO
REAL MEDIA
SUMX2=C
SUMX2=C,I0)N
READ(I,13)
FORMAT(I,13)
DO I=1,N
  READ(I,20)X
  FORMAT(I,E14.7)
WRITE(5,20)X
SUMX=SUMX+X
SUMX2=SUMX2+X**2
CONTINUE
FN=FLOAT(N)
MEDIA=SUMX/FN
S=((SUMX2-(SUMX**2)/FN))/(FN-1.0)**.5
WRITE(5,3C)MEDIA,S
FORMAT(IH,*,*,MEDIA = ,2X,E14.7,/,*, DESVIO PADRAO = ,E14.7)
CALL EXIT
END

// XE0000E+00
5.300000E+00
5.720000E+00
6.770000E+00
5.260000E+00
4.330000E+00
5.450000E+00
6.090000E+00
5.640000E+00
5.810000E+00
5.750000E+00
MEDIA = 5.6129980E+00
DESVIO PADRAO = 6.2657750E-01
FIM DA EXECUCAC
FCRAM EXECUTADAS 59 INSTRUcoes

```

Podemos observar que as duas primeiras declarações, após o REAL MEDIA, definem os valores iniciais de SUMX e SUMX2. A primeira leitura irá informar quantos dados experimentais serão lidos. O comando DO executará o restante do programa de acordo com este READ.

Se quisermos simplificar este programa, suprimindo a leitura do valor de N, poderemos perfurar diretamente o cartão contendo a declaração DO 1 I=1,N com o valor de N desejado. Por exemplo: DO 1 I=1,15.

Se quisermos transformar este programa numa subrotina e incluir o cálculo da variância (quadrado do desvio - padrão), poderemos fazê-lo com relativa facilidade, introduzindo as variáveis N e X no conjunto de argumentos.

Aiba et al. publicaram na pág. 135, da segunda edição do livro "Biochemical Engineering" um gráfico aplicável a processos de cultivo de microorganismos em regime permanente. As seguintes equações foram as responsáveis pela obtenção do diagrama abaixo:

$$\mu = D \left\{ 1 + \omega \left( 1 - \frac{1 + \omega - \frac{F_0}{F} \cdot \frac{X_0}{X}}{1 + \omega - \frac{F_0}{F}} \right) \right\} \quad (5.20)$$

If  $X_0/X \doteq 0$ , Eq. (5.20) is reduced to:

$$\mu = D \left\{ 1 + \omega \left( 1 - \frac{1 + \omega}{1 + \omega - \frac{F_0}{F}} \right) \right\} \quad (5.21)$$

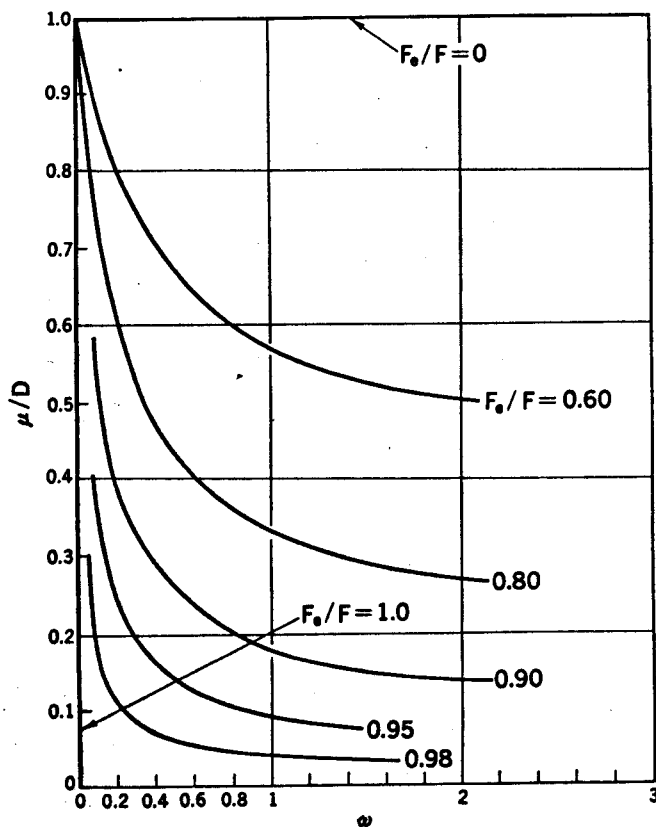


Fig. 5.4.  $\mu/D$  vs.  $\omega$  for recycling in steady state.



A - NOMENCLATURA:

$\mu$  = taxa específica de crescimento celular,  $h^{-1}$

D = taxa de diluição,  $h^{-1}$

$F_e$  = vazão volumétrica do produto desejado,  $m^3/h$

F = vazão volumétrica da alimentação do processo,  $m^3/h$

$\omega$  = razão de reciclo hidráulico no processo

$X_e$  = concentração de células no produto desejado, g/l

X = concentração de células no fermentador do processo, g/l

$(\mu/D)$ ,  $(F_e/F)$ ,  $(X_e/X)$  e  $\omega$  são números adimensionais.

B - Nossa intenção não é a de discutir como os autores obtiveram este gráfico e para que serve. Desejamos, em nosso caso, obter tabelados numa listagem do computador IBM1130, alguns valores que nos permitam reproduzir, numa folha de gráfico, as curvas originais.

C - Faremos os grupos adimensionais variarem nos seguintes intervalos:

$(F_e/F)$  -----> (0.0, 1.0)

$\omega$  -----> (0.0, 3.0)

$(\mu/D)$  -----> (0.0, 1.0)

$(X_e/X)$  -----> (0.0, 1.0)

D - Temos algumas formas de informar ao computador de quanto em quanto irá variar cada variável acima: a precisão desejada é escolha nossa.

E - Temos de adaptar as variáveis à linguagem FORTRAN:

$(\mu/D)$  -----> MUD       $\omega$  -----> RRH

$(F_e/F)$  -----> FEF       $X_e/X$  -----> XEX

F - As equações assumirão os seguintes aspectos:

$$\text{MUD} = 1 + \text{RRH} \left( 1 - \frac{1 + \text{RRH} - (\text{FEF} \cdot \text{XEX})}{1 + \text{RRH} - \text{FEF}} \right) \quad (\text{A})$$

Para XEX desprezível, isto é, praticamente nulo, teremos:

$$\text{MUD} = 1 + \text{RRH} \left( 1 - \frac{1 + \text{RRH}}{1 + \text{RRH} - \text{FEF}} \right) \quad (\text{B})$$

G - Em nossos cálculos, empregaremos inicialmente a eq. (B).

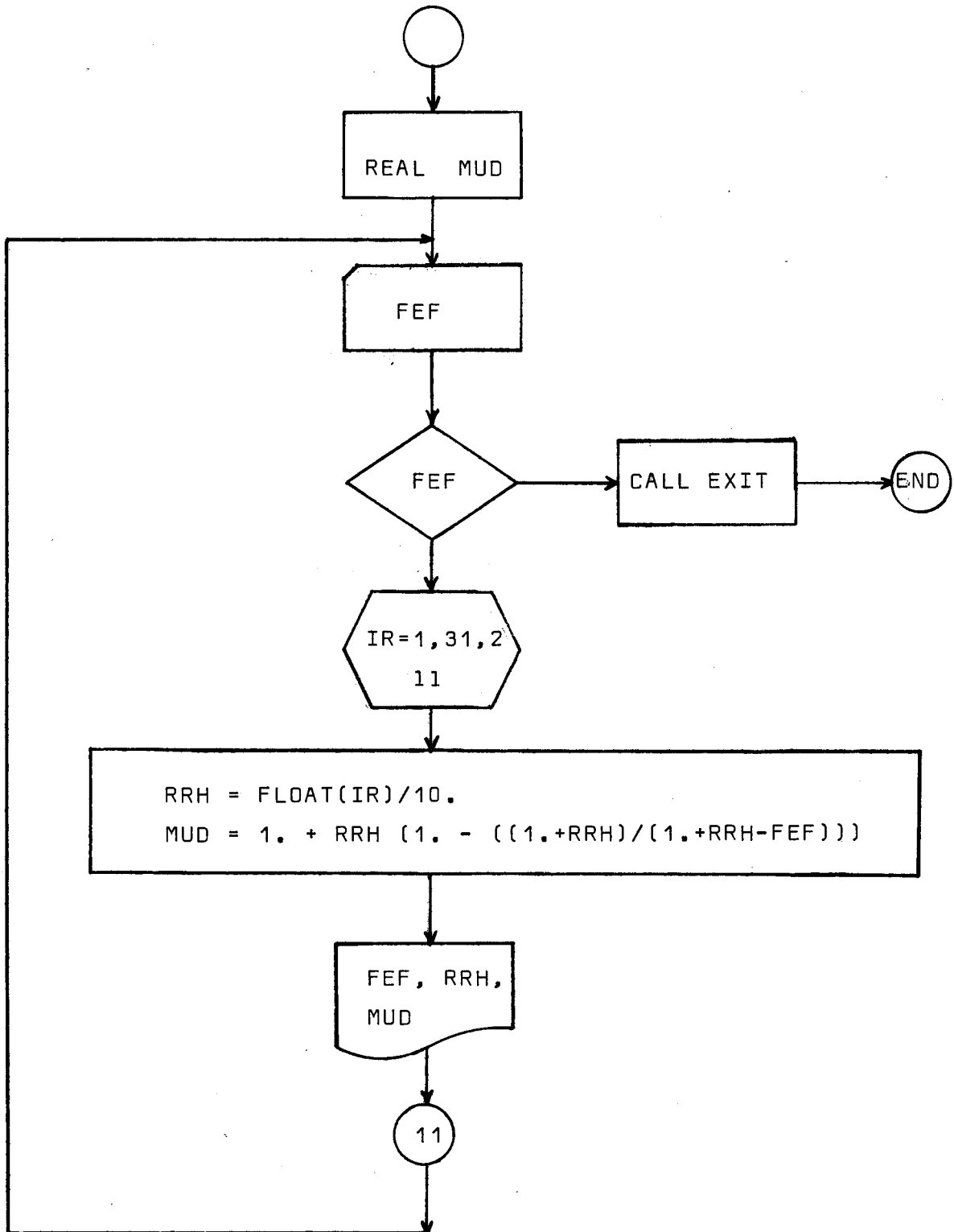
H - ALTERNATIVAS DE PROCEDIMENTO:

- 1) podemos obter uma curva para um determinado valor de FEF, fazendo apenas RRH variar de 0.0 até 3.0. Por exemplo, podemos obter a curva para FEF = 0.60.
- 2) podemos fazer RRH variar através do comando DØ e, então, fazer FEF variar de 0.0 até 1.0. Para cada um valor de RRH obteremos diversos valores de FEF. Aqui obteremos para cada valor de RRH um ponto de cada curva de FEF.

Para a primeira alternativa, utilizaremos um cartão declarando o valor de FEF, ou, então, o comando READ para os diferentes valores de FEF que se desejar obter. Entretanto, é mais fácil empregar comandos DØ embutidos, conforme exemplificaremos adiante, escolhendo inclusive o grau de precisão desejado e o número de pontos satisfatórios.

Tais procedimentos são válidos para a segunda alternativa, porém a listagem dos resultados é melhor para a primeira e menos confusa, pois apresenta os pontos de uma curva de FEF de cada vez.

I - FLUXOGRAMA



Aqui utilizamos cartão "flag" para FEF=0.000.

Na página seguinte apresentamos a listagem com os resultados obtidos.

J - LISTAGEM DO PROBLEMA

```

1      REAL MUD
2      READ(8,1)FEF
3      1  FORMAT(F8.3)
4      IF(FEF)5,7,5
5      DO 11 IR=1,31,2
6      RRH=FLOAT(IR)/IC.
7      MUD=1.+RRH*(1.-((1.+RRH)/(1.+RRH-FEF)))
8      WRITE(5,100)FEF,RRH,MUD
9      100 FORMAT(1X,F8.3,3X,F8.3,3X,F8.3)
10     11 CONTINUE
11     GO TO 2
12     7  CALL EXIT
13     END
    
```

0 ERRC(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

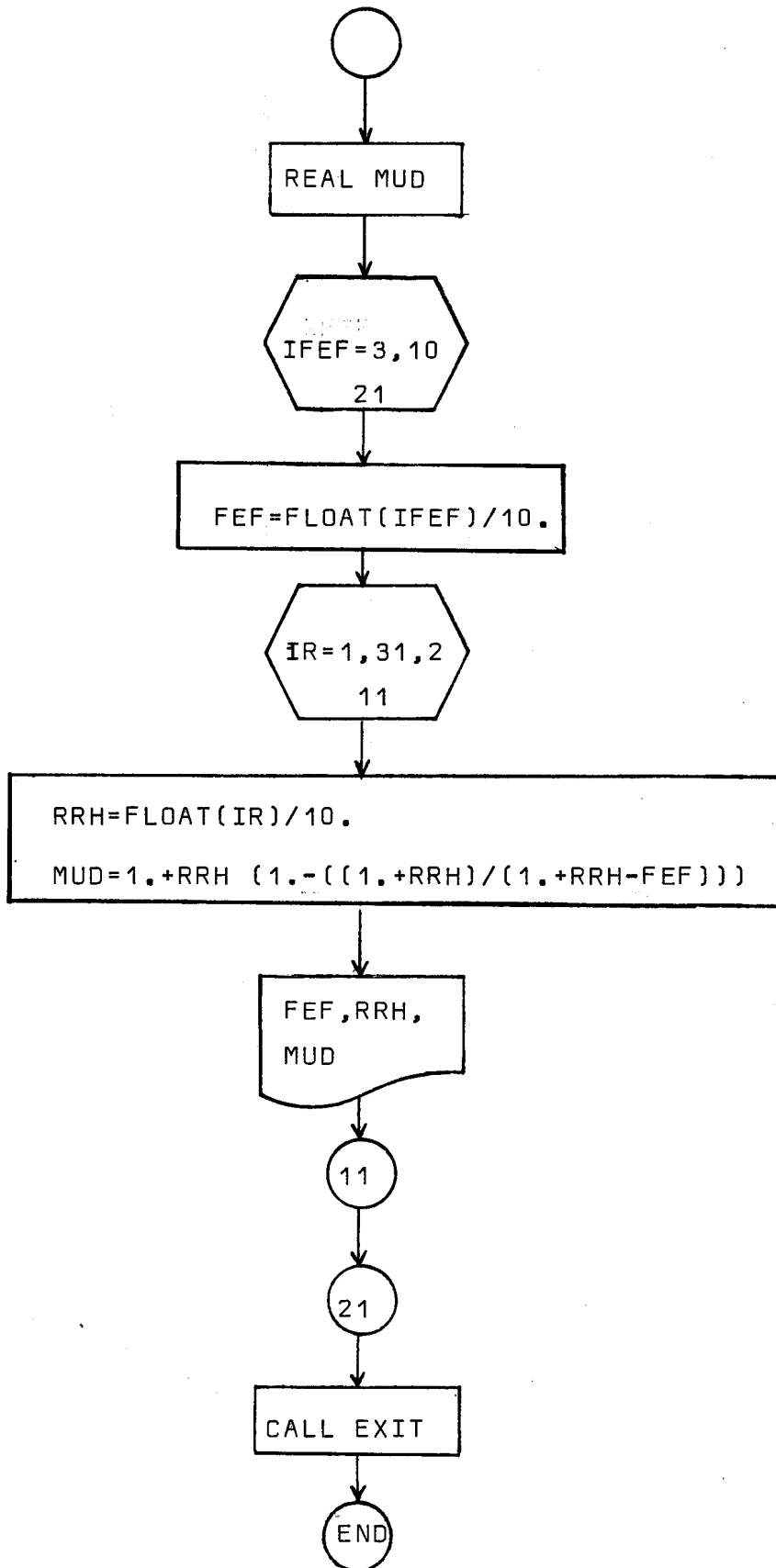
PAG. 2 \* COPPE-FORTRAN \* 25/05/76  
 FIM DA COMPILACAO \* ESTAO OCUPADAS 113 DAS 8042 PALAVRAS DISPON.

// XEC		
0.680	0.100	0.838
0.680	0.299	0.670
0.680	0.500	0.585
0.680	0.700	0.533
0.680	0.900	0.498
0.680	1.100	0.473
0.680	1.300	0.454
0.680	1.500	0.439
0.680	1.699	0.427
0.680	1.900	0.418
0.680	2.100	0.409
0.680	2.300	0.403
0.680	2.500	0.397
0.680	2.700	0.392
0.680	2.899	0.387
0.680	3.100	0.383
0.750	0.100	0.785
0.750	0.299	0.590
0.750	0.500	0.500
0.750	0.700	0.447
0.750	0.900	0.413
0.750	1.100	0.388
0.750	1.300	0.370
0.750	1.500	0.357
0.750	1.699	0.346
0.750	1.900	0.337
0.750	2.100	0.329
0.750	2.300	0.323
0.750	2.500	0.318
0.750	2.700	0.313
0.750	2.899	0.309
0.750	3.100	0.305
0.980	0.100	0.183
0.980	0.299	0.081
0.980	0.500	0.057

PAG. 3 \* COPPE-FORTRAN \* 25/05/76

0.980	0.700	0.047
0.980	0.900	0.041
0.980	1.100	0.037
0.980	1.300	0.034
0.980	1.500	0.032
0.980	1.699	0.031
0.980	1.900	0.030
0.980	2.100	0.029
0.980	2.300	0.028
0.980	2.500	0.027
0.980	2.700	0.027
0.980	2.899	0.026
0.980	3.100	0.026

K - OUTRA SOLUÇÃO PARA O PROBLEMA 0.1: COMANDO DØ EMBUTIDO



## L - LISTAGEM DO PROBLEMA PARA O COMANDO DØ EMBUTIDO

```
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*IOCS(2501READER,1403PRINTER)
1      REAL MUD
2      DO 21 IFEF=3,10
3      FEF=FLOAT(IFEF)/10.
4      DO 11 IR=1,31,2
5      RRH=FLOAT(IR)/10.
6      MUD=1.+RRH*(1.-((1.+RRH)/(1.+RRH-FEF)))
7      WRITE(5,100)FEF,RRH,MUD
8      100 FORMAT(1X,F8.3,3X,F8.3,3X,F8.3)
9      11 CONTINUE
10     21 CONTINUE
11     CALL EXIT
12     END
```

Ø ERRC(S) E Ø ADVERTENCIA(S) ASSINALADOS NESTA COMPILAÇÃO

## M - COMENTÁRIOS

m.1 - Os resultados obtidos com a listagem acima foram representados num gráfico semelhante ao publicado por Aiba / et al., para diversos intervalos de variação de RRH, e, também, para diversos valores de incrementos no comando DØ.

m.2 - um artifício para a obtenção de valores decimais, nos intervalos desejados, foi aplicado para cada comando DØ, e, utilizamos FLOAT, dentro da declaração seguinte, para dar a compatibilidade necessária ao processamento.

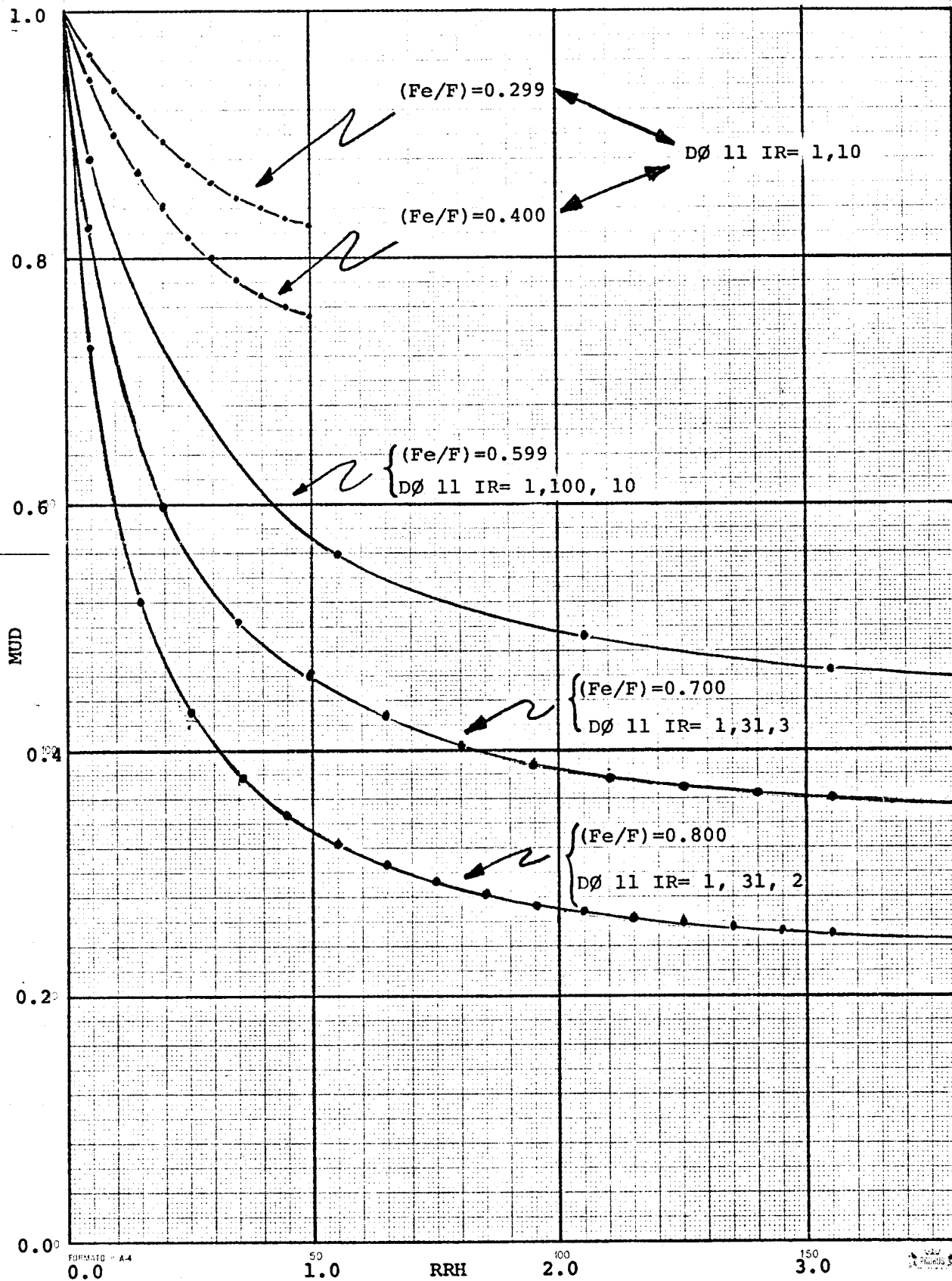
m.3 - observe-se que REAL é sempre utilizada no princípio do programa, ao passo que FLOAT é aplicado como função, durante o processamento. O mesmo ocorre para INTEGER e IFIX. REAL e INTEGER são comandos. FLOAT(I) e IFIX(X) são funções.

m.4 - conforme o valor final do comando DØ e o incremento, poderemos obter um grande número de valores para cada curva, sendo que muitos deles desnecessários. Então é necessário escolher intervalos de variação e incrementos adequados aos nossos objetivos.

m.5 - a primeira curva e a segunda foram obtidas para um intervalo de variação, que não correspondia ao desejado. Cada comando DØ externo alcançava apenas 10 pontos, isto é, cada curva ficou representada apenas por 10 pontos.

m.6 - a terceira curva também não correspondeu ao intervalo desejado, pois excedeu o limite horizontal. E apenas quatro pontos aparecem representados.

m.7 - as duas curvas da parte inferior do gráfico satisfazem ao problema.



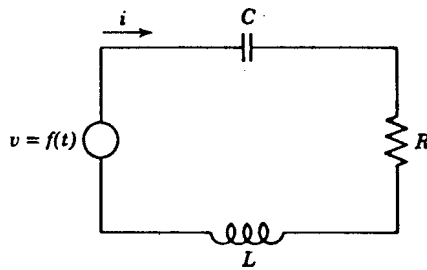
## 8. SISTEMA OSCILADOR

A equação diferencial ordinária de segunda ordem com coeficientes constantes é uma das mais familiares da Matemática Aplicada, pois descreve uma variedade de sistemas físicos comuns. Por exemplo, suponhamos a existência de uma partícula de massa  $m$  oscilando ao longo do eixo  $X$ , onde atua uma força  $kx$  forçando o retorno à posição original, com uma força de atrito  $r(dx/dt)$  que se opõe ao movimento, e sob a influência de uma força externa, que é função do tempo.

A equação diferencial descritiva do movimento é:

$$m \frac{d^2x}{dt^2} + r \frac{dx}{dt} + kx = f(t)$$

Podemos também considerar o exemplo de um circuito em série, contendo uma resistência  $R$ , um capacitor  $C$ , uma indutância  $L$ , e uma fonte de voltagem  $f(t)$ , apresentados na figura abaixo.



Neste caso a equação diferencial para a corrente é:

$$L \frac{d^2i}{dt^2} + R \frac{di}{dt} + \frac{i}{C} = \frac{df(t)}{dt}$$

A solução desta equação é dada em qualquer texto de equações diferenciais. Para nossos propósitos simplesmente consideramos uma voltagem  $V$  constante, para corrente contínua e apresentamos a solução sem demonstrá-la.



CASO 1. OSCILATÓRIO

Se  $R^2 - 4L/C$  for menor que zero, a solução é:

$$i = \frac{V}{\omega_n L} e^{\alpha t} \sin \omega_n t$$

onde

$$\omega_n = \sqrt{\frac{1}{LC} - \frac{R^2}{4L^2}}$$
$$\alpha = -\frac{R}{2L}$$

CASO 2. CRITICAMENTE AMORTECIDO

Se  $R^2 - 4L/C$  for igual a zero, a solução é:

$$i = \frac{V}{L} t e^{\alpha t}$$

onde

$$s = \frac{-R}{2L}$$

CASO 3. SUPER AMORTECIDO

Se  $R^2 - 4L/C$  for maior que zero, a solução é:

$$i = K_1 e^{s_1 t} + K_2 e^{s_2 t}$$

onde

$$K_1 = -\frac{V}{2L \sqrt{\frac{R^2}{4L^2} - \frac{1}{LC}}}$$

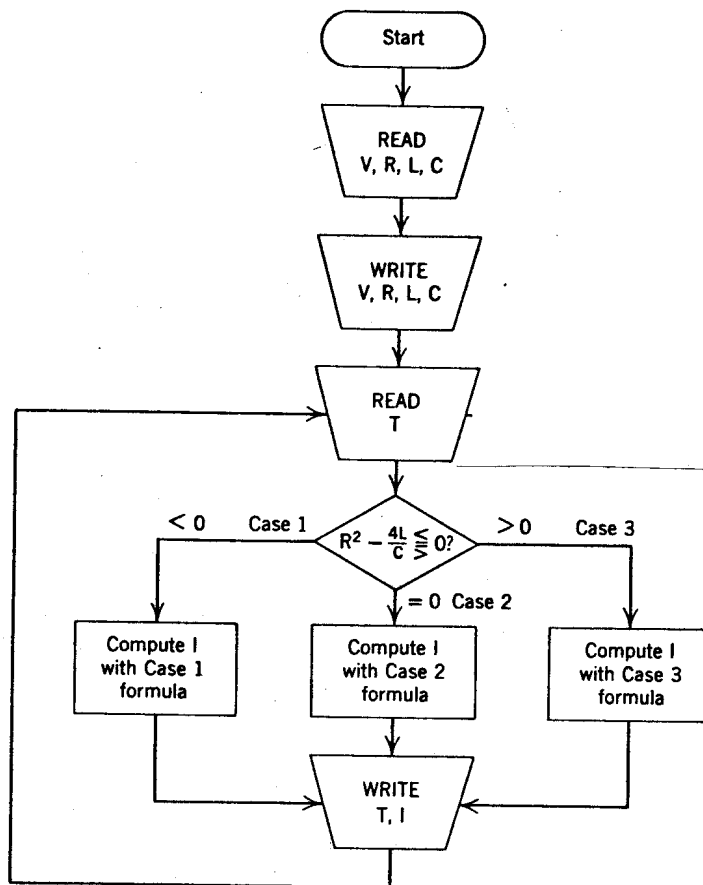
$$K_2 = -K_1$$

$$s_1 = \frac{-R}{2L} - \sqrt{\frac{R^2}{4L^2} - \frac{1}{LC}}$$

$$s_2 = \frac{-R}{2L} + \sqrt{\frac{R^2}{4L^2} - \frac{1}{LC}}$$

Deseja-se escrever um programa que lerá um cartão contendo valores de V, R, L e C e então leia uma série de cartões de tempo, calcule a corrente para cada um, de modo que possamos plotar a resposta do circuito. O programa deverá abranger todos os três casos apresentados.

FLUXOGRAMA:



Este fluxograma não indica uma forma elegante para o término do programa. O resultado poderá ser o apresentado na listagem seguinte, ou qualquer outro modo ao gosto do programador.





	.1000CE +02	.10000E+03	.25000E+00	.10000E-05
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.2000CE -03	.74850E-02		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.4000CE -03	.13259E-01		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.6000CE -03	.16577E-01		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.8000CE -03	.17125E-01		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.1000CE -02	.15032E-01		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.1600CE -02	-.61821E-03		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.2000CE -02	-.10018E-01		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.2600CE -02	-.10700E-01		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.9200CE -02	-.16462E-02		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.9400CE -02	-.43952E-03		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.9600CE -02	.74107E-03		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.9800CE -02	.17185E-02		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.1000CE -01	.23599E-02		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.2000CE -01	.31773E-03		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.3000CE -01	-.44771E-06		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.4000CE -01	-.58796E-05		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.5000CE -01	-.78339E-06		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.1000CE +00	-.36484E-10		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.2000CE +00	.71265E-19		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.3000CE +00	.15795E-28		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.4000CE +00	-.33361E-36		
CASO 1	-OSC ILATORIO	OU SUB-AMORTECIDO		
CASO 1	.5000CE +00	.58455E-45		

## 9. "IF" LÓGICO E TRANSFERÊNCIA CONTROLADA

O IF lógico tem a forma geral IF (e)S, onde e é uma expressão lógica e S é uma outra declaração qualquer (que não seja outro IF lógico ou algum comando DO).

Para caracterizar uma expressão lógica é necessário utilizar qualquer um dos seguintes operadores de relações ou comparação.

<u>Operador de Comparação</u>	<u>Significado</u>
.LT.	Menor que
.LE.	Menor que ou igual a
.EQ.	Igual a
.NE.	Não é igual a
.GT.	Maior que
.GE.	Maior que ou igual a

A pontuação nestes operadores é necessária para distinguí-los de possíveis nomes de variáveis.

A ação do IF lógico ocorre do seguinte modo: se a expressão lógica é verdadeira, a declaração S é executada; se a expressão lógica é falsa, a declaração S não é executada. Em ambos os casos, a próxima declaração executada é aquela seguinte ao IF, a menos que S seja um comando GO TO e a expressão seja verdadeira.

O poderio do IF lógico torna-se consideravelmente ampliado pela combinação de várias expressões comparativas com os operadores lógicos .AND., .OR., e .NOT..

A declaração de transferência controlada tem a seguinte forma:

```
GO TO (n1, n2, ..., nm)i
```

onde  $n_1, n_2, \dots, n_m$  são números de declarações e  $i$  é uma variável inteira, podendo tomar somente um dos valores  $1, 2, \dots, m$ . O valor da variável  $i$  é determinado pela própria lógica (ou execução) do programa. O controle do programa passará para as declarações  $n_1, n_2, \dots, n_m$  dependendo do valor obtido de  $i$ , isto é, o programa seguirá para a declaração  $n_k$  se  $i = k$ .

B6700/B7700 F O R T R A N C O M P I L A T I O N M A R K

FILE 8=CARTOES,UNIT=READER  
 FILE 5=SAIDA,UNIT=PRINTER

CC CC SISTEMA GSCILADOR

REAL I,L,K1,K2  
 INTEGER CASO  
 REAC(8,1)V,R,L,C  
 1 FORPA(8F10.0)  
 2 WRIT(5,2)V,R,L,C  
 3 FORMINACAO DO CASO  
 4 (IX,E13.5)  
 DETER IF(F\*+2.LT.4.0\*L/C) CASO = 1  
 IF(F\*+2.EQ.4.0\*L/C) CASO = 2  
 IF(F\*+2.GT.4.0\*L/C) CASO = 3

CC OBSERVE, NESTE EXEMPLD, QUE A VARIAVEL 'A' E USADA NO LUGAR 'A' E 'S',  
 CC POIS QUE AMBOS POSSUEM VALORES NUMERICOS IGUAIS. DE MODO SEMELHANTE,  
 CC O VALOR DE 'M' E USADOC PARA 'M' E 'D'.

M=SRRT(ABS(1.0/(L\*C)-R\*\*2/(4.0\*L\*\*2)))  
 A=-R/(2.0\*L)  
 K1=-V/(2.0\*L\*M)  
 K2=-K1  
 S1=A-M  
 S2=A+M

C LEIA UM CARTAO COM O VALOR DE TEMPO

C 6 ESCLEHA DA FORMULA  
 C 20 I=V/(M\*L)\*EXP(A\*T)\*SIN(W\*T)  
 C 30 I=V/L \*T\*EXP(A\*T)  
 C 40 I=K1\*EXP(S1\*T)+K2\*EXP(S2\*T)  
 C 50 WRIT(5,2)T,I

GO TO 6  
 CALL EXIT  
 END

10000CE+02	10000E+03
20000CE-03	74850EE-02
40000CE-03	132577E-01
60000CE-03	16577E-01
80000CE-02	17125EE-01
10000CE-02	15032EE-03
16000CE-02	61821EE-03
20000CE-02	10018EE-01
26000CE-02	10700EE-01
34000CE-02	16462EE-03
36000CE-02	43952EE-03
38000CE-01	74107EE-03
20000CE-01	17185EE-02
30000CE-01	33573EE-03
50000CE-01	447771E-06
10000CE+00	587796E-06
20000CE+00	783339E-10
30000CE+00	36484E-19
40000CE+00	71265E-28
50000CE+00	15795E-36
	33361E-45
	58455E-45

.25000E+00 .10000E-05

## 10. CÁLCULO DA PERDA DE CARGA EM HIDRÁULICA

Um dos procedimentos mais comuns ao engenheiro químico, de projeto ou de controle de processos, é basear seus cálculos em valores obtidos de tabelas.

Conforme as etapas do raciocínio empregado, ele escolhe uma linha e uma coluna da tabela, e, retira um determinado valor para uma constante, ou uma variável, que ele vai empregar.

O exemplo abaixo ilustra, uma situação que nos é bastante familiar.

Determine a perda de carga em 200 ft de uma tubulação de ferro estrudado, com 6 polegadas de diâmetro nominal, por onde passam 250 GPM de água.

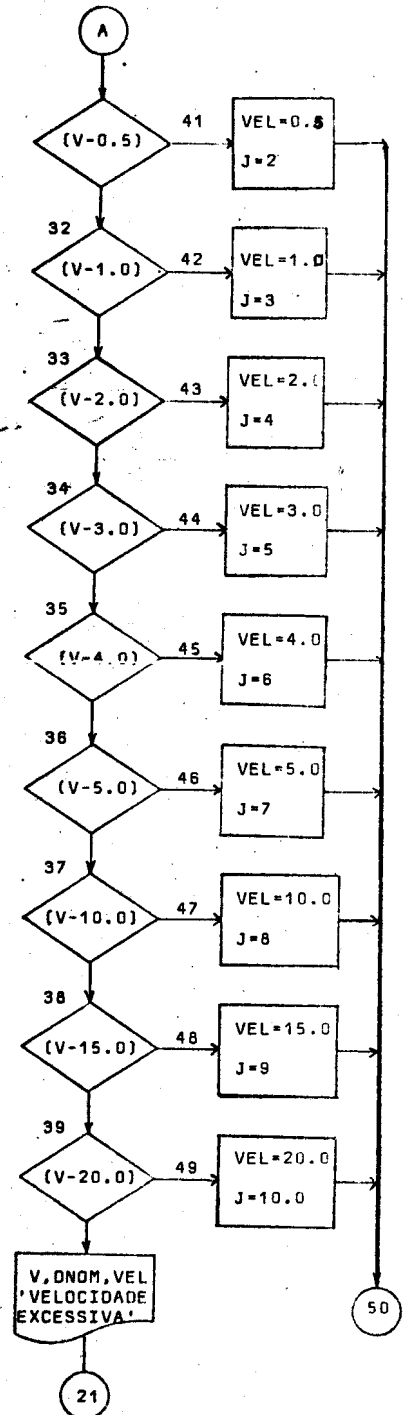
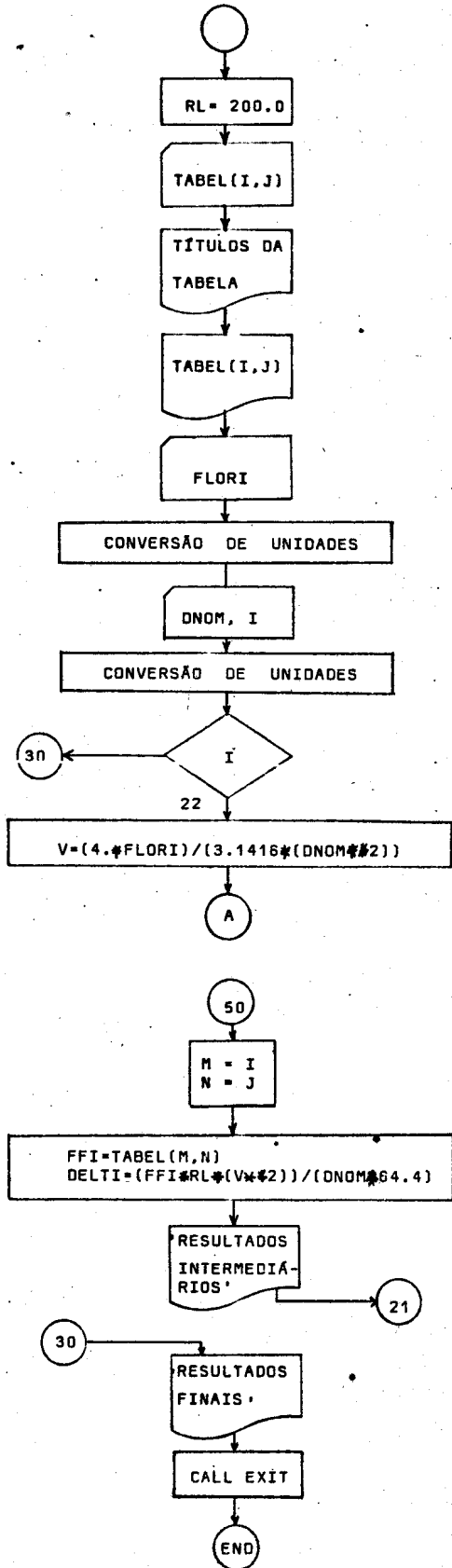
SOLUÇÃO:  $Q = 0.557 \text{ ft}^3/\text{sec}$   $V = Q/A = 2.83 \text{ ft/sec}$   
 $V^2/2g = 0.124 \text{ ft}$  Da tabela abaixo,  $f = 0.025$ .  
 Empregando a formula de Darcy-Weisbach:

$$h_f = 0.025 \times \frac{200}{\frac{1}{2}} \times 0.124 = 1.2 \text{ ft de água}$$

Diâmetro Nominal em pol.	VELOCIDADE MÉDIA DA ÁGUA (V) EM PÉS POR SEGUNDO								
	0.5	1.0	2.0	3.0	4.0	5.0	10.0	15.0	20.0
1/2	0.042	0.038	0.034	0.032	0.030	0.029	0.025	0.024	0.023
3/4	.041	.037	.033	.031	.029	.028	.025	.024	.023
1	.040	.035	.032	.030	.028	.027	.024	.023	.023
1 1/2	.038	.034	.031	.029	.028	.027	.024	.023	.023
2	.036	.033	.030	.028	.027	.026	.024	.023	.022
3	.035	.032	.029	.027	.026	.025	.023	.022	.022
4	.034	.031	.028	.026	.026	.025	.023	.022	.021
5	.033	.030	.027	.026	.025	.024	.022	.022	.021
6	.032	.029	.026	.025	.024	.024	.022	.021	.021
8	.030	.028	.025	.024	.023	.023	.021	.021	.020

Um projeto de hidráulica, vinculado à escolha de tubulações e bombas, poderá necessitar de repetidas consultas à uma tabela. O programa, proposto a seguir, sugere uma forma de incluímos uma tabela num projeto a ser executado por computador digital IBM 1130. Este permite localizar o valor desejado e inseri-lo automaticamente nas equações que dele necessitam.

CÁLCULO DA PERDA DE CARGA EM HIDRAULICA







```

2 3X,'5.0',2X,'10.0',2X,'15.0',2X,'20.0'//1X,'J=1',8X,'J=2',3X,
3 'J=3',3X,'J=4',3X,'J=5',3X,'J=6',3X,'J=7',3X,'J=8',3X,'J=9',
4 2X,'J=10',/)
WRITE(5,800)((TABEL(I,J),J=1,10),I=1,10)
READ(5,801)FLORI
FORMAT(6X,F1C.3)
801 FLORI=FLORI#0.002228
21 READ(5,802)DNOM,I
802 FORMAT(5X,F5.2,5X,I2)
DNOM=DNOM/12.
IF(I)0,30,22
22 V=(4.*FLORI)/(3.1416*(DNOM**2))
IF(V-.5)41,41,32
41 VEL=0.5
J=2
GO TO 50
32 IF(V-.0)42,42,33
42 VEL=1.0
J=3
33 IF(V-.0)43,43,34
43 VEL=2.0
J=4
GO TO 50
34 IF(V-.0)44,44,35
44 VEL=3.0
J=5
GO TO 50
35 IF(V-.0)45,45,36
45 VEL=4.0
J=6
GO TO 50
36 IF(V-.0)46,46,37
46 VEL=5.0
J=7
GO TO 50
37 IF(V-.0)47,47,38
47 VEL=10.0
J=8
GO TO 50
38 IF(V-.0)48,48,39
48 VEL=15.0
J=9
GO TO 50
39 IF(V-.0)49,49,40
49 VEL=20.0
J=10
GO TO 50
40 VEL=V
WRITE(5,805)V, DNOM, VEL
805 FORMAT('VELOCIDADE EXCESSIVA',/3X,'V=',E15.5,2X,'DNOM=',F6.3,'VEL=
1',E15.5)
GO TO 21
50 M=I
51 N=J
FFI=TBEL(M,N)
DELT=(FFI*RL*(V**2))/(DNOM*64.4)
WRITE(5,804)DELT,I,FLORI,RL, DNOM,I,V,J,FFI,VEL
804 FORMAT('RESULTADOS INTERMEDIARIOS',/3X,'DELT=',F15.5,IX,'FLORI=',
F15.5,IX,'RL=',F10.5,IX,'DNOM=',F10.5,IX,'I=',I3,IX,'V=',

```

É recomendável que as conversões de unidades sejam feitas imediatamente após a leitura de dados, e fora de 'loops' (ou circuito de cálculos e operações), para evitar a repetição das conversões e resultados anômalos.

PAG. 3

\* COPPE-FORTRAN \* 17/12/76

62 2E15.5,IX,'J'=,I3,IX,'FFI=',F10.5,/,,' VEL=',F10.5)  
 GO TO 21  
 63 30 WRITE(5,803)DELTII,FLORI,RL,DNOM,I,V,J,FFI,VEL  
 64 803 FORMAT('RESULTADOS FINAIS',/3X,'DELTII=',F15.5,IX,  
 1 'FLORI=',F15.5,IX,'RL=',F10.5,IX,'DNOM=',F10.5,IX,  
 2 'I=',I3,IX,'V=',F10.5,IX,'J=',I3,IX,'FFI=',F10.5,/,,'  
 3) VEL=',F10.5  
 65 CALL EXIT  
 66 END

O ERRC(S) E O ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

FIM DA COMPILACAO \* ESTADO OCUPADAS 1053 DAS 8042 PALAVRAS DISPONIVEIS

// XEQ

DNOM	VELOCIDADE MEDIA (V) EM FEET PER SECOND									
IN.	0.5	2.0	3.0	4.0	5.0	10.0	15.0	20.0		
J=1	J=2	J=3	J=4	J=5	J=6	J=7	J=8	J=9	J=10	
0.50	0.042	0.038	0.033	0.032	0.030	0.029	0.025	0.024	0.023	
0.75	0.041	0.037	0.033	0.031	0.025	0.028	0.025	0.024	0.023	
1.00	C.040	0.035	0.032	0.030	0.028	0.027	0.024	0.023	0.023	
1.50	0.038	0.033	C.031	0.029	0.028	0.027	0.024	0.023	0.023	
2.00	0.036	0.033	0.030	0.028	0.027	0.026	0.024	0.023	0.022	
3.00	0.035	0.032	0.029	0.027	0.026	0.025	0.023	0.022	0.022	
4.00	0.033	0.031	0.028	0.026	0.025	0.025	0.023	0.022	0.021	
5.00	C.033	0.030	C.027	0.026	0.025	0.024	0.022	0.022	0.021	
6.00	0.032	0.029	0.026	0.025	0.024	0.024	0.022	0.021	0.021	
8.00	C.030	0.028	0.025	C.024	0.023	0.023	0.021	0.021	0.020	

VELOCIDADE EXCESSIVA  
 V= 4.08495E+02 DNOM= 0.041VEL= 4.08495E+02

VELOCIDADE EXCESSIVA  
 V= 1.81553E+02 DNOM= 0.067VEL= 1.81553E+02

RESULTADOS INTERMEDIARIOS  
 CELTI= 1.24957 FLORI= 0.55700 RL= 200.00000 DNOM= 0.50000 I= 9 V= 2.83677E+00 J= 5 FFI= 0.02500  
 VEL= 3.00000

RESULTADOS FINAIS  
 CELTI= 1.24957 FLORI= 0.55700 RL= 200.00000 DNOM= 0.16666 I= 0 V= 2.83677 J= 5 FFI= 0.02500  
 VEL= 3.00000

FIM DA EXECUCAO 60 INSTRUICOES  
 FCRAM EXECUTADAS

// FCR

```

* LIST SOURCE PROGRAM
* ONE WORD INTEGERS
* IOCS(2501 READER,1403 PRINTER)
C
C      CALCULO DA PERDA DE CARGA EM HIDRAULICA
C
C      PROGRAM* ELABORADO PELO ENGENHEIRO QUIMICO LUIZ FERNANDO LEITE
C
1      REAL LENG
2      DIMENSION TABEL(11,11)
3      READ(9,10)((TABEL(I,J),J=1,11),I=1,11)
4      10 FORMAT(8F10.6)
5      WRITE(5,1)
6      1 FORMAT(1H,1X,'DNOM',5X,'DINT',15X,'VELOCIDADE          MEDIA          (V)
* EM FEET PER SECOND'//2X,'IN',7X,'IN')
7      WRITE(5,2)(TABEL(1,J),J=3,11),((TABEL(I,J),J=1,11),I=2,11)
8      2 FORMAT('+',19X,9(2X,F4.1,3X)//99('')//'(F5.2,5X,F6.4,5X,F6.3,3X,
*F6.3,7X,F6.3,3X,F6.3,3X,F6.3,3X,F6.3,3X,F6.3,3X,F6.3,3X,F6.3,3X,F6.3)
9      READ(9,3)FLORI,DNOM,LENG
10     3 FORMAT(3F10.3)
11     FLORI=FLORI*0.002228
12     DO 5 I=2,11
13     IF(TAPEL(I,1)-DNOM)5,7,7
14     5 CONTINUE
15     WRITE(5,6)
16     6 FORMAT('DIAMETRO NOMINAL EXCEDE VALORES DA TABELA')
17     GO TO 13
18     7 M=I
19     DINT=TABEL(M,2)/12.
20     V=(4.*FLORI)/(3.1416*(DINT**2))
21     DO 11 J=3,11
22     IF(V-TABEL(I,J))8,8,11
23     8 N=J
24     VEL=TABEL(1,N)
25     FFI=TABEL(M,N)
26     DELTI=FFI*LENG*(V**2)/(DINT*64.4)
27     WRITE(5,9)DELTI,FLORI,LENG,DNOM,M,N,DINT,V,FFI,VEL
28     9 FORMAT(' RESULTADOS FINAIS'/1X,'DELTI=',F8.5,4X,'FLORI=',F8.5,4X,
*'LENG=',F7.2,5X,'DNOM=',F4.1,3X,'N=',I2,3X,'N=',I2,3X,'DINT=',F7.4

```

PAG. 2

\* COPPE-FORTRAN \* 20/12/76

```

* ,5X,/' V=',F5.1,2X,'FFI=',F5.3,2X,'VEL=',F5.2)
29     GO TO 13
30     11 CONTINUE
31     WRITE(5,12)V,DNOM,VEL
32     12 FORMAT(' VELOCIDADE EXCEDE VALORES DA TABELA'/' V=',F8.5,5X,'DNOM=
*',F4.1,' VEL=',F4.2)
33     13 CALL EXIT
34     END

```

0 ERR(C) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

FIM DA COMPILACAO \* ESTADO OCUPADAS 851 DAS 8042 PALAVRAS DISPONIVEIS

// XEC	CNOM	DINT	VELOCIDADE	MEDIA	(V)	EM	FEET	PER	SECOND		
IN	IN		0.5	1.0	2.0	3.0	4.0	5.0	10.0	15.0	20.0

\*\*\*\*\*

0.50	0.6220	0.042	0.038	0.033	0.032	0.030	0.029	0.025	0.024	0.023
0.75	0.8240	0.041	0.037	0.033	0.031	0.029	0.028	0.025	0.024	0.023
1.00	1.0490	0.040	0.035	0.032	0.030	0.028	0.027	0.024	0.023	0.023
1.50	1.6100	0.038	0.033	0.031	0.029	0.028	0.027	0.024	0.023	0.023
2.00	2.0670	0.036	0.033	0.030	0.028	0.027	0.026	0.024	0.023	0.022
3.00	3.0680	0.035	0.032	0.029	0.027	0.026	0.025	0.023	0.022	0.022
4.00	4.0260	0.033	0.031	0.028	0.026	0.026	0.025	0.023	0.022	0.021
5.00	5.0470	0.033	0.030	0.027	0.026	0.025	0.024	0.022	0.022	0.021
6.00	6.0650	0.032	0.029	0.026	0.025	0.024	0.024	0.022	0.021	0.021
8.00	7.9810	0.030	0.028	0.025	0.024	0.023	0.023	0.021	0.021	0.020

RESULTADOS FINAIS

DELTI= 1.18403 FLORI= 0.55700 LENG= 200.00 DNOM= 6.0 N=10 N= 6 DINT= 0.5054  
V= 2.7 FFI=0.025 VEL= 3.00

FIM DA EXECUCAO

FCRAM EXECUTADAS 41 INSTRUCOES

## 11. ROTINAS E SUBROTINAS FORTRAN

Uma rotina em FORTRAN pode ser considerada uma sequência de declarações destinadas a efetuar uma computação desejada. Uma rotina pode ser parte de um programa maior, destinado a resolver um tipo específico de problema. Após as linhas gerais do programa terem sido esboçadas em fluxograma (ou definidas), uma boa forma de escrever o programa é desenvolver rotinas para cada uma das fases em separado, e, então, conectar estas partes logicamente.

Uma rotina pode ser um conjunto de declarações que irá resultar no cálculo do valor de uma variável ou conjunto de variáveis. Usualmente a rotina consistirá de etapas que estabelecem os valores para certas variáveis (iniciação), seguidas por etapas que executam cálculos usando as variáveis.

Algumas rotinas são usadas tão frequentemente que podem ser escritas pelo próprio programador e anexadas aos seus programas tantas vezes quantas forem necessárias. Em certos casos, são incluídas como parte do compilador. Estas rotinas previamente escritas são denominadas de SUBROUTINE ou subrotinas.

Consideremos o cálculo do volume de um dado número de moles de um gás sob temperatura e pressão conhecidas, através da equação do gás ideal,  $PV = nRT$ . A rotina pode consistir em estabelecer os valores para as variáveis, seguida por uma declaração que executa o cálculo do volume. Uma sequência possível é:

```
PATM = 1.200
TKELV = 25.0 + 273.
XMOL = 2.32 E-2
VOLUM = (XMOL*.0821*TKELV)/PATM
```

Outro exemplo de uma rotina é o caso no qual a percentagem de água num hidrato é calculada a partir de dados experimentais. Estes são coletados em laboratório pelo aquecimento de amostras de hidrato, anteriormente pesadas em balanças de precisão, para retirar a água e, então, determinar a massa do resíduo. Uma sequência de declarações apropriadas para o cálculo seria:

```
HYD = 2.1823
ANH = 1.8213
PH2O = ((HYD - ANH)/HYD*100.
```

Suponha que a experiência tenha sido realizada três vezes, de modo que três conjuntos de dados foram coletados. Neste caso, necessitaria mos de uma rotina para calcular a percentagem média da massa de água no hidrato, poderíamos ter:

```
HYD1 = 2.0035
ANH1 = 1.7863
HYD2 = 3.2704
ANH2 = 2.7342
HYD3 = 2.7248
AND3 = 2.2704
FNUM = 3.0
PH201= ((HYD1 - ANH1) / (HYD1))*100.
PH202= ((HYD2 - ANH2) / HYD2)*100.
PH203= ((HYD3 - ANH3) / HYD3)*100.
AVEPC= (PH201 + PH202 + PH203)/FNUM
```

Observe que as últimas cinco declarações são genéricas e podem ser usadas para calcular a percentagem média de qualquer grupo de 3 conjuntos de dados experimentais. Novos conjuntos de dados podem ser introduzidos pela redefinição dos dados das variáveis, que estão indicadas nas seis primeiras declarações da rotina acima.

## FÓRMULAS EMPÍRICAS DE COMPOSTOS

Os cálculos envolvem a determinação do número de moles de cada elemento e as razões molares. Entretanto, para encontrar as razões molares, é necessário dividir o número de moles de cada elemento pelo menor número de moles. Para tanto, o menor número de moles deve ser determinado pela ordenação dos valores molares. Isto pode ser executado por uma subrotina.

É necessário introduzir as porcentagens, o número de gramas por mol e o símbolo de cada elemento. Se ordenamos os valores molares dos elementos, deveremos observar quais elementos trocam suas posições, ordenando simultaneamente os seus símbolos.

O programa abaixo destina-se a ler cartões, contendo a composição percentual, gramas por mol e os símbolos dos elementos. Determinam-se as razões molares, e daí, a fórmula empírica.

```
C
C  ESTA E UMA SUBROTINA QUE ORDENA OS ELEMENTOS DE UM VETOR X(I)
C  DE ACORDO COM O TAMANHO E A ORDEM DOS ELEMENTOS DO VETOR Y(I)
C
C
      SUBROUTINE ORD(N,X,Y)
      DIMENSION X(20),Y(20)
      NM=N-1
      DO 10 K=1,NM
      JO=K+1
      DO 10 J=JO,N
      IF(X(K)-X(J))5,10,10
5  SAVE=X(K)
   SAV=Y(J)
   X(K)=X(J)
   Y(K)=Y(J)
   X(J)=SAVE
   Y(J)=SAV
10 CONTINUE
   RETURN
   END
```

Podemos observar que X e Y poderão conter os valores molares e os símbolos dos elementos e que a ordenação ocorre simultaneamente para os N elementos. A ordenação é efetuada comparando-se os elementos K (anterior) e J (posterior).

PROGRAMA PRINCIPAL:

C  
C  
C  
C  
C

DETERMINACAO DAS FORMULAS EMPIRICAS

```
DIMENSION P(20),GPM(20),EL(20),FMOL(20)
READ(8,10)N,(P(I),GPM(I),EL(I),I=1,N)
10 FORMAT(I3,/, (F6.3,2X,F7.3,2X,A2))
DO 1 I=1,N
  1 FMOL(I)=(P(I)*100.)/GPM(I)
  CALL ORD(N,FMOL,EL)
DO 2 I=1,N
  2 FMOL(I)=FMOL(I)/FMOL(N)
  WRITE(5,20)EL(I),FMOL(I),I=1,N)
20 FORMAT(A2,2X,E14.7)
CALL EXIT
END
```

Os dados do programa, de acordo com a FORMAT 10 seriam:

47.000	012.010	C	Cartão de dados para Carbono
09.900	001.008	H	Cartão de dados para Hidrogênio
27.400	014.010	N	Cartão de dados para o Nitrogênio.

E os resultados seriam:

```
// XEQ
H 5J0218340E+00
C 2.0009790E+00
N 1J0000000E+00
```

FIM DA EXECUCAO

A partir desta listagem deduzimos uma fórmula em pírca de  $C_2H_5N$ .



## 12. DIMENSIONAMENTO DE TUBOS

O dimensionamento de tubos é um dos problemas práticos fundamentais no projeto de uma unidade industrial. O número ou valor exato obtido a partir de um programa de computador certamente não representará o diâmetro interno real do tubo finalmente especificado.

Ao invés deste, o tamanho padrão de tubos mais próximos será normalmente utilizado. Os valores exatos assim obtidos são frequentemente utilizados como critérios de decisão em procedimentos de projeto. Apresentaremos a seguir as bases da solução deste problema e o programa em forma de subrotina.

### SOLUÇÃO:

#### A) PROCEDIMENTO DE PROJETO

Considera-se que algumas propriedades do fluido escoando (viscosidade e massa específica) são conhecidas. Também devem ser especificadas a vazão volumétrica, a perda de carga PERMISSÍVEL, o comprimento equivalente total, e a rugosidade do tubo.

Para escoamento turbulento de um fluido newtoniano, através de um duto horizontal, podemos considerar a Equação de Bernoulli, também denominada de Equação de Darcy-Weisbach.

$$\frac{\Delta P}{L} = \frac{f u^2 \rho}{2 g_c D}$$

sendo  $u$  = velocidade linear do fluido  
 $f$  = fator de atrito de Moody

Utilizaremos também a Equação de Colebrook e o Número de Reynolds e a expressão para o fator de atrito em regime laminar. A Equação de Colebrook é válida para regime turbulento.

$$\frac{1}{\sqrt{f}} = -2 \log_{10} \left( \frac{\epsilon}{3.7D} + \frac{2.51}{Re\sqrt{f}} \right)$$

$$Re = \frac{Du\rho}{\mu}$$

$$f = \frac{64}{Re}$$

## B1 ALGORÍTMO

- a) Considerar  $f=64/Re$  na Eq. Bernoulli, supondo, portanto,  $Re = 2100$  e encontrar um valor para D.
- b) Calcule Re com este valor de D.
- c) Calcule o fator de atrito com este valor de Re, somente se  $Re \leq 2100$ .
- d) Sendo  $Re > 2100$ , que é o caso mais comum, calcule o fator de atrito através da Eq. Colebrook.
- e) Para este caso, admitir um valor inicial para o fator de atrito. Por exemplo, fazer  $f = 0.03$ .
- f) Utilizar novamente a Eq. Bernoulli para calcular um valor para D.
- g) Calcular Re com este novo valor de D.
- h) Aqui é necessário introduzir o Método de Interpolação de NEWTON-RAPHSON, para não interromper nosso raciocínio, suponhamos que tenhamos obtido o valor de f desejado, dentro da tolerância desejada.
- i) Com este fator de atrito obtido pelo método iterativo de Newton-Raphson, calcularemos o valor final de D com a Eq. Bernoulli.

C) MÉTODO DE NEWTON-RAPHSON

Este método encontra a raiz  $x$  de uma função  $y(x)=0$ , onde  $x$  está implicitamente definida. O mesmo será aplicado à Eq. Colebrook. A função  $y$  é definida por:

$$y = x + 2 \log_{10} (A + Bx)$$

para  
 $y=0$

$$\frac{1}{\sqrt{f}} = -2 \log_{10} \left( \frac{\epsilon}{3.7D} + \frac{2.51}{\text{Re} \sqrt{f}} \right)$$

Observamos que:

$$x = 1 / \sqrt{f}$$

$$A = \epsilon / 3.7D$$

$$B = 2.51 / \text{Re}$$

A raiz é obtida por aproximações sucessivas, que são fornecidas pela seguinte fórmula de interpolação:

$$x_{i+1} = x_i - y(x_i) / y'(x_i)$$

ou  $x_{\text{novo}} = x - y/y'$

Torna-se necessário obter a derivada primeira de  $y$ , que é dada pela seguinte expressão:

$$y' = \frac{dy}{dx} = 1 + \frac{2B \log_{10} e}{A + Bx}$$

Portanto, valores sucessivos de  $x$  são obtidos até que  $y(x)$  esteja suficientemente próximo ou igual a zero. Então, usaremos este método para procurar o valor de  $f$ , dentro do procedimento de projeto, que visa a obter o tamanho da tubulação, isto é, o seu diâmetro  $D$ .

D) SIMPLIFICAÇÕES MATEMÁTICAS NECESSÁRIAS

Quando fazemos um programa ou subrotina, temos de elaborá-lo de modo sucinto, para que haja economia de memória e tempo de processamento. Portanto, é interessante expressar certas variáveis em função das dimensões ou funções que apresentem valores conhecidos, conforme descrevemos abaixo:

$$\frac{\Delta P}{L} = f \frac{\rho u^2}{2 g_c D}$$

$$u^2 = \frac{F^2}{A^2} = \frac{16}{\pi^2} \times \frac{F^2}{D^4}$$

$$f = \frac{64}{Re} = \frac{64 \mu}{D \frac{4F}{\pi D^2} \rho} = \frac{16 \pi \mu D}{F \rho} \quad \therefore \frac{\Delta P}{L} = \frac{128 \mu F}{3.1416 \times 32.2 \times D^4}$$

$$\therefore D = \sqrt[4]{\frac{L \times \mu F 128}{\Delta P \times 101.1595}}$$

$$Re = \frac{4F\rho}{\pi D \mu}$$

$$\frac{\Delta P}{L} = f \cdot \frac{\rho \left(\frac{4F}{\pi D^2}\right)^2}{2 g_c D} = f \cdot \frac{\rho}{64.4 D^5} \left(\frac{4F}{\pi}\right)^2$$

$$\therefore D^5 = f \cdot \frac{\rho L}{64.4 \Delta P} \left(\frac{4F}{\pi}\right)^2$$

$$Q = \frac{\rho L}{64.4 \Delta P} \left(\frac{4F}{\pi}\right)^2$$

$$D^5 = f \times Q$$

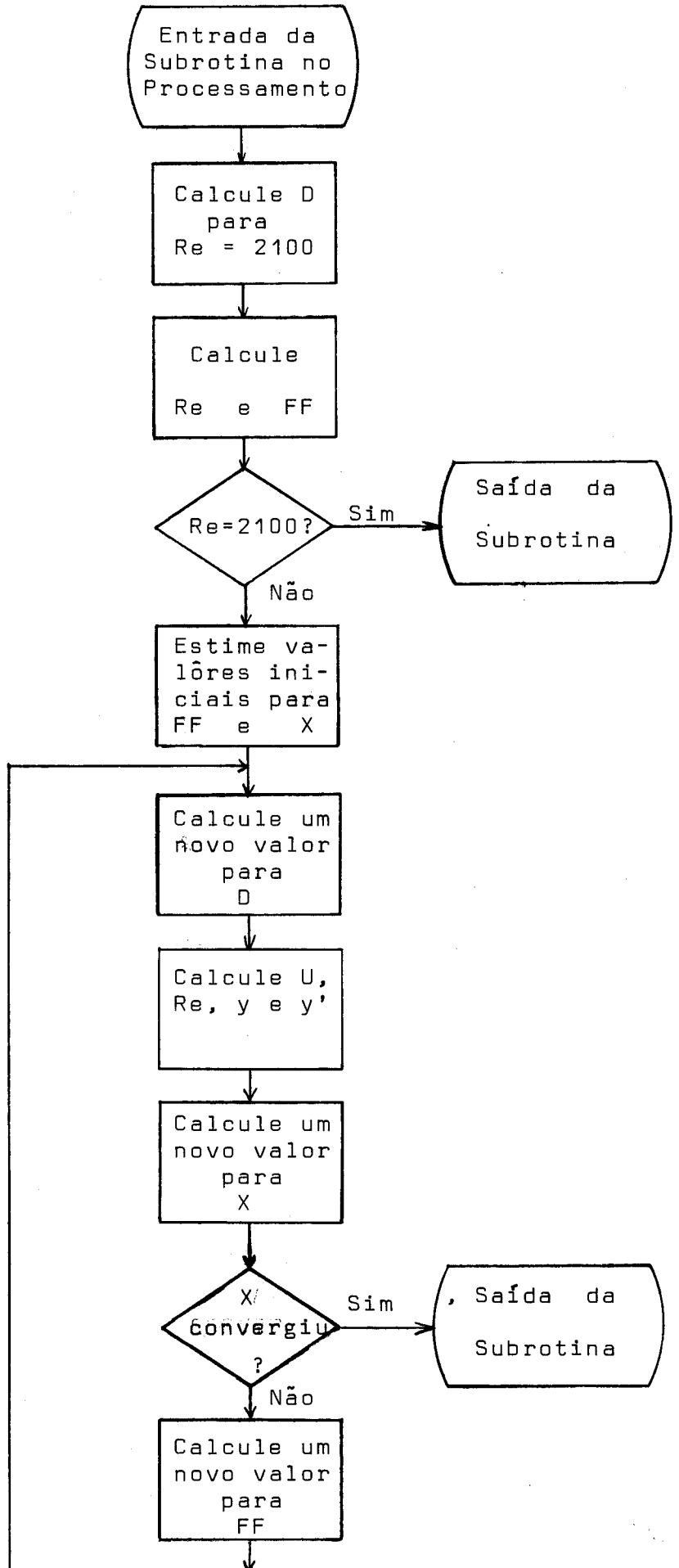
$\therefore$

$$D = (f \times Q)^{0.20}$$

$$\log_e \frac{1}{10} = 0.43429$$

Observamos que f pode ser incluído numa expressão bem simples para o cálculo de D, isto é, com os dados de entrada calcula-se o valor de Q apenas uma vez em cada iteração, atendendo-se à economia mencionada.

E) FLUXOGRAMA



F) ASPECTO DA SUBROTINA NA LISTAGEM DO IBM-1130

```

1  *LIST SOURCE PROGRAM
2  *CNE WORD INTEGERS
3  SUBROUTINE PIPE (RHO,LENGT,FLORA,DP,VISCO,ROUGH,RE,FF,DFT)
4  REAL LENGT
5  DFT*((128.0*VISCO*LENGT*FLORA)/(101.1595*DP))*0.25
6  RE=(FLORA*4.0*RHO)/(3.1416*DFT*VISCO)
7  FF=64.0/RE
8  IF(RE-2100.)8,8,9
9  RETURN
10 8 FF=C.03
11 Q=(RHO*LENGT*(4.0*FLORA/3.1416)**2)/(64.4*DP)
12 DFT*(FF*Q)**0.20
13 AREA=(3.1416*DFT**2)/4.0
14 U=FLORA/AREA
15 RE=DFT*U*RHO/VISCO
16 A=ROUGH/(3.7*DFT)
    B=2.51/RE
    X=1.0/SQRT(FF)

```

\* COPPE-FORTRAN \* 19/05/76

```

17 Y=X+0.8695*A*LOG(A+B*X)
18 YPRIM=1.0+(2.0*0.43429*B)/(A+B*X)
19 XNEW=X-Y/YPRIM
20 IF(ABS(XNEW-X)/(XNEW)-1.0E-6)8,8,22
21 22 FF=1.0/(XNEW**2)
22 GO TO 11
23 END

```

G) OBSERVAÇÕES

O PROGRAMA PRINCIPAL

lê cartões de dados e informações  
chama subrotinas e as utiliza  
imprime resultados.

A SUBROTINA

recebe os dados do programa principal, que já estão armazenados na memória  
executa os cálculos e toma as decisões de sua especialidade e não armazena dados após a sua utilização.

### 13. AJUSTAMENTO DE CURVAS

A transformação de dados tabelados numa expressão algébrica é denominada ajustamento de curvas. Este assunto é muito importante para o engenheiro projetista, porque é mais fácil introduzir em seus cálculos (por computador) uma expressão matemática, do que inúmeros valores numéricos. Além disto, não conhecemos nenhum computador que "leia gráficos" e empregue os valores lidos em seus cálculos.

Deseja-se, portanto, partir de dados experimentais representativos do fenômeno em questão, obter a melhor curva, através de métodos numéricos processados por computador, e empregar esta expressão nos cálculos de engenharia ou pesquisa.

O processo de ajustamento de curvas envolve a escolha da melhor curva representativa dos dados colhidos. Isto é, numa situação experimental alguns erros de medida existem e uma curva ajustada tão próxima quanto possível da realidade é escolhida para representar a função. Frequentemente conhece-se a natureza provável da função.

Reproduzimos a seguir o programa apresentado por Pacitti para Ajustamento de Curvas para Estudos Macroeconômicos. Dada uma série histórica, depois de caracterizada uma certa regularidade de variação, escolhe-se dentre as seguintes curvas: reta, parábola do 2º ou 3º grau, exponencial ( $Y=A.B^X$ ) e hipérbole, aquela que melhor possa expressar a sua "tendência".

O programa ajustará aos dados a equação da curva escolhida; pelo método dos "mínimos quadrados", calculará os logaritmos dos valores observados, variância explicada, percentagem de variância explicada, elasticidade e erro padrão de estimativa.

Os dados de entrada constam de: um 1º cartão que fornece o código da equação escolhida (1 para reta, 2 para parábola do 2º grau, 3 para parábola do 3º grau, 4 para exponencial e 5 para hipérbole) e o número de dados observados, seguido dos cartões dos valores observados, respeitado o FORMAT especificado no programa principal.

Evidentemente, podemos testar todas as curvas acima mudando o primeiro cartão.





```

77 DJ 41 J= S,NN
78 TEMP=(K,KJ)
79 A(K,J)=ALL(J)
80 A(L,J)=TEMS
81 TEMPE=(K)
82 A(K)=F(L)
83 D(L)=F(E)
84 A(L)=A(L,K)/A(K,K)
85 A(L)=A(L,K)/A(K,K)
86 A(L)=A(L,K)/A(K,K)
87 A(L)=A(L,K)/A(K,K)
88 A(L)=A(L,K)/A(K,K)
89 A(L)=A(L,K)/A(K,K)
90 A(L)=A(L,K)/A(K,K)
91 A(L)=A(L,K)/A(K,K)
92 A(L)=A(L,K)/A(K,K)
93 SUMED.
94 DO 700 J=IPI,NN
95 SUM=SUM+A(I,J)*C(J)
96 Y(I)=Y(I)-SUM/A(I,I)
97 I=I+1
98 A(I,I)=800,800,710
99 A(I,I)=
100 A(I,I)=
101 A(I,I)=
102 A(I,I)=
103 A(I,I)=

```

O ERRO(S) E O ADVERTENCIA(S) ASSINALADOS VESTA COMPILACAO  
 FIM DA COMPILACAO \* ESTAO OCUPADAS 1385 DAS 8042 PALAVRAS DISPONIVEIS

// DUP

\*STORE WS JA APCL  
 K 21 \* OPERACAO COMPLETADA

// FOR

\*LIST SOURCE PROGRAM  
 \*ONE WORD INTEGERS  
 \*JCSYZ50ARCADE,1403PRINTER)

```

101 DIMENSION Y(50),X(50),XC(6),Z(50),YY(50),YK(50)
102 I=READ(1,1000)
103 FPRMA=I*83H
104 WRITE(5,1001)
105 FORMY(I,X) DEPARTAMENTO DE ENGENHARIA QJIMICA - ESCOLA DE QJIMICA
106 I = UNIVERSIDADE FEDERAL DO RIO DE JANEIRO,777)
107 WRITE(5,1000)
108 XEAD=I*1002IK,M
109 FPRMA=I*1003I,Y(I),I=1,M)
110 DO 100 I=1,M)
111 X(I)=I*104I(I-1)
112 DO 20 J=1,20,30,35),K
113 CALL A(I,J,X,K,XC)
114 DO 30 J=1,36
115 CALL A(J,I,Y,M,AA,38)
116 DO 35 J=1,18
117 CALL A(I,J)=ALJ*FLOAT(I)
118 YY(I)=ALJ*Y(I)
119 Y(I)=ALJ*Y(I)
120 CALL A(I,J)=ALJ*Y(I),M,XC)
121 SUM=0.
122 DO 50 I=1,M
123 Z(I)=SUM+Y(I)*X(2I)*YK(I)
124 SUM=SUM+Y(I)*Z(I)*I*62
125 DO 36 I=1,M
126 Z(I)=Z(I)+I*62
127 Y(I)=Z(I)+I*62
128 Y(I)=Z(I)+I*62
129 SUM=SUM+(Y(I)-Z(I))*2
130 DO 40 J=1,90
131 XX=VA*Y(M)
132 DO 55 J=1,60,70,80)K
133 WRITE(5,1004)XC(2),XC(1)
134 1004 FJIMATY(I,X ,Y*,E(L),K*,E(L),L)
135

```

\* CPPE-FORTRAN # 06/12/76

```

137 WRITEL5(1005)
138 FORMAT(IX,*,VALORES OBSERVADOS E VALORES AJUSTADOS)
139 DO 110 I=1,M
140 I1=I-1
141 Z(I)=YC(2)*FLOAT(I1)+XC(1)
142 SUM=SM+Y(I1)-Z(I1)**2
143 WRITE(5,1006)I1,Y(I1),Z(I1)
144 EXRMA(1,X1,Y1,12,*)=FILL(3)
145 E=XC(2)*FLOAT(M-I)+XC(1)
146 SUM=SM/FLOAT(M-2)
147 CD=2*(I-SUM)
148 Z(I)=Z(I)+CD
149 CD=2*(I-SUM)
150 Z(I)=Z(I)+CD
151 Z(I)=Z(I)+CD
152 * PARABOLA DE SEGUNDO GRAU,*,/IK,*,Y=*,EI
153 FORMAT(IX,*,VALORES OBSERVADOS E VALORES AJUSTADOS)
154 DO 120 I=1,M
155 I1=I-1
156 Z(I1)=YC(3)*FLOAT(I1)**2+XC(2)*FLOAT(I1)+XC(1)
157 SUM=SM+(Y(I1)-Z(I1))**2
158 WRITE(5,1009)I1,Y(I1),Z(I1)
159 FORMAT(IX,*,Y(1),Z(1),I1,Z(I1))
160 E=XC(2)*FLOAT(M-1)+XC(3)
161 I1+XC(3)*FLOAT(M-1)**2
162 SUM=SM/FLOAT(M-3)
163 EP=SUM*(SUM)
164 CD=I1/666
165 WRITEL5(1010)XC(4),XC(3),XC(2),XC(1)
166 FORMAT(IX,*,AJUSTAMENTO DE PARABOLA DE TERCERO GRAU,*,/IK,*,Y=*,EI
167 DO 170 I=1,M
168 I1=I-1
169 Z(I1)=YC(4)*FLOAT(I1)**3+XC(3)*FLOAT(I1)**2+XC(2)*FLOAT(I1)+XC(1)
170 SUM=SM+(Y(I1)-Z(I1))**3

```

\* CPPE-FORTRAN # 06/12/76

```

172 WRITEL5(1012)I1,Y(I1),Z(I1)
173 FORMAT(IX,*,Y(1),Z(1),I1,Z(I1))
174 E=XC(2)*FLOAT(M-1)+XC(3)*FLOAT(M-1)**2+XC(4)*FLOAT(M-1)**3
175 I1+XC(4)*FLOAT(M-1)**3
176 SUM=SM/FLOAT(M-4)
177 EP=SUM*(SUM)
178 VV=VEI*(Y,Z)
179 DO 180 I=1,88
180 E=FLOAT(I)**88
181 SUM=SM/ELOAT(M-2)
182 EP=SUM*(SUM)
183 A=EXP(AA)
184 B=EXP(BB)
185 C=EXP(CC)
186 DO 187 I=1,777
187 I1=I-1
188 I2=I1-1
189 I3=I2-1
190 I4=I3-1
191 I5=I4-1
192 I6=I5-1
193 I7=I6-1
194 I8=I7-1
195 I9=I8-1
196 I10=I9-1
197 I11=I10-1
198 I12=I11-1
199 I13=I12-1
200 I14=I13-1
201 I15=I14-1
202 I16=I15-1
203 I17=I16-1
204 I18=I17-1

```



## 14. CINÉTICA DE REAÇÕES QUÍMICAS

A maneira pela qual a velocidade da reação varia com a concentração dos reagentes pode ser estabelecida através da ordem da reação. Geralmente descobre-se nos experimentos que a velocidade (ou taxa) da reação é proporcional à potência do reagente A, à potência do reagente B, etc.

$$\text{TAXA} = v = k (A)^{\alpha} (B)^{\beta} \dots = k C_A^{\alpha} C_B^{\beta} \dots$$

sendo que a ordem da reação é simplesmente

$$\text{ORDEM} = n = \alpha + \beta + \dots$$

Portanto, tal reação é dita ser de ordem  $\alpha$  com relação ao reagente A, e de ordem  $\beta$  com relação ao reagente B, etc.

Dentre os métodos empregados na determinação da ordem de reação, destaca-se, como adequado para uso de computadores, O MÉTODO DAS TABELAS. O procedimento usado é o de calcular as constantes de velocidade, usando as expressões listadas na tabela 1, para várias etapas da reação. Um exemplo do uso deste método para uma reação de segunda ordem é visto na tabela 4, onde o valor de k é apresentado conforme a bibliografia consultada. A coluna da direita apresenta resultados bastante próximos entre si.

A tabela 1 nos apresenta equações para o seguinte mecanismo de reação irreversível:



onde  $a_0$  = quantidade inicial do reagente A  
 $x$  = quantidade do reagente A consumida no tempo  $t$ .  
 $a_0 - x$  = quantidade remanescente do reagente A no tempo  $t$ .  
 $-\frac{d(a_0 - x)}{dt} = \frac{dx}{dt} =$  taxa de consumo do reagente A.

Também podemos verificar, de modo semelhante, qual o mecanismo de uma reação, através da tabela 2, onde as reações são irreversíveis.

É nosso intuito elaborar duas subrotinas para verificarmos se as reações seguem um dos diversos modelos expostos nas duas tabelas.

TABELA 1 - EQUAÇÕES DA TAXA PARA A REAÇÃO:  $A \rightarrow B$

ORDEM	FORMA DIFERENCIAL	FORMA INTEGRADA	UNIDADES DA CONSTANTE DA TAXA
0	$\frac{dx}{dt} = k$	$k = \frac{x}{t}$	Mols Litro <sup>-1</sup> segundos <sup>-1</sup>
1/2	$\frac{dx}{dt} = k(a_0 - x)^{1/2}$	$k = \frac{2}{t} \left[ a_0^{1/2} - (a_0 - x)^{1/2} \right]$	Mols <sup>1/2</sup> Litro <sup>-1/2</sup> segundos <sup>-1</sup>
1	$\frac{dx}{dt} = k(a_0 - x)$	$k = \frac{1}{t} \ln \frac{a_0}{a_0 - x}$	Segundos <sup>-1</sup>
3/2	$\frac{dx}{dt} = k(a_0 - x)^{3/2}$	$k = \frac{2}{t} \left[ \frac{1}{(a_0 - x)^{1/2}} - \frac{1}{a_0^{1/2}} \right]$	Litros <sup>1/2</sup> mol <sup>-1/2</sup> segundos <sup>-1</sup>
2	$\frac{dx}{dt} = k(a_0 - x)^2$	$k = \frac{1}{t} \frac{x}{a_0(a_0 - x)}$	Litros mol <sup>-1</sup> segundos <sup>-1</sup>
3	$\frac{dx}{dt} = k(a_0 - x)^3$	$k = \frac{1}{2t} \frac{2a_0x - x^2}{a_0^2(a_0 - x)^2}$	Litros <sup>2</sup> mol <sup>-2</sup> segundos <sup>-1</sup>

NOTA: Em cada caso, a expressão de k refere-se ao consumo de reagente, onde  $a_0$  = concentração inicial do reagente e  $x$  = concentração do reagente consumida no tempo  $t$ .

TABELA 2 - FUNDAMENTOS DA TAXA PARA DIVERSOS MECANISMOS DE REAÇÃO.

ESQUEMA ESTEQUIOMÉTRICA	ECUAÇÃO DA TAXA	ECUAÇÃO INTEGRADA
A = ...	$\frac{dx}{dt} = kax = k(a_0 - x)(x + x_0)$ auto-catalítica	$\frac{1}{a_0 - x_0} \ln \frac{a_0(x_0 + x)}{x_0(a_0 - x)} = kt$
A + B = ...	$\frac{dx}{dt} = k_a b = k(a_0 - x)(b_0 - x)$	$\frac{1}{b_0 - a_0} \ln \frac{a_0(b_0 - x)}{b_0(a_0 - x)} = kt$
A + 2B = ...	$\frac{dx}{dt} = k_a b = k(a_0 - x)(b_0 - 2x)$	$\frac{1}{b_0 - 2a_0} \ln \frac{a_0(b_0 - 2x)}{b_0(a_0 - x)} = kt$
A + B + C = ...	$\frac{dx}{dt} = k_a b c = k(a_0 - x)(b_0 - x)(c_0 - x)$	$\frac{1}{(a_0 - b_0)(b_0 - c_0)(c_0 - a_0)} \ln \left( \frac{b_0 - x}{a_0} \right)^{b_0 - c_0} \left( \frac{c_0 - x}{b_0} \right)^{c_0 - a_0} \left( \frac{a_0 - x}{c_0} \right)^{a_0 - b_0} = kt$
2A + B = ...	$\frac{dx}{dt} = k_a^2 b = k(a_0 - 2x)^2(b_0 - x)$	$\frac{2}{2b_0 - a_0} \left( \frac{1}{(a_0 - x)} - \frac{1}{a_0} \right) + \frac{2}{(2b_0 - a_0)^2} \ln \frac{b_0(a_0 - x)}{a_0(b_0 - x)} = kt$
A + B = ...	$\frac{dx}{dt} = k_a^2 b = k(a_0 - x)^2(b_0 - x)$	$\frac{1}{(b_0 - a_0)} \left( \frac{x}{a_0(a_0 - x)} \right) + \frac{1}{(b_0 - a_0)^2} \ln \frac{b_0(a_0 - x)}{a_0(b_0 - x)} = kt$

NOTA: Para estes casos, as concentrações iniciais dos reagentes são designadas entre si, e a equação da taxa não comensura necessariamente com a estequiometria da reação, sendo o símbolo  $x$  indicadores de uma quantidade inicial, e as demais variáveis pertencentes a determinadas substâncias para os reagentes a, b e c no instante t, e x = reagente consumido (a ou b ou c).

Sejam as seguintes subrotinas:

CINER - indica a ordem de uma reação química, segundo a Tabela 1. Exige que os valores de  $a_0$ ,  $x$  e  $t$  sejam fornecidos. Calcula o valor de  $k$  para todos os instantes da amostragem e suas respectivas concentrações de reagentes consumidos.

CINEQ - indica o mecanismo mais adequado para a reação segundo a tabela 2. Exige que os valores de  $a_0$ ,  $b_0$ ,  $c_0$ ,  $x$  e  $t$  sejam fornecidos. Calcula o valor de  $k$  para todos os instantes da amostragem e as respectivas concentrações de reagentes consumidos.

Para caracterizar bem a validade das duas subrotinas iremos inicialmente testá-las para exemplos já consagrados na bibliografia de Cinética Química.

Conforme já dissemos, a tabela 4, exposta abaixo, indica valores de  $k$ , segundo uma reação de segunda ordem.

Table 4 ALKALINE HYDROLYSIS OF ETHYL NITROBENZOATE

$a_0 = 0.05$  mole per liter

$t$ , sec	Percentage change	$k \times 10^2$ , liters mole <sup>-1</sup> sec <sup>-1</sup>
120	32.95	8.19
180	41.75	7.96
240	48.8	7.94
330	58.05	8.39
530	69.0	8.40
600	70.35	7.92

Quando é dada a percentagem de conversão (  $n\%$  ) do reagente A no tempo  $t$ , o valor de  $x$  será dado por:

$$X = n \cdot a_0 / 100$$

// FOR

```

*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
SUBROUTINE INVER(AO,X,T,KJ,KJ5,K1,K15,K2,K3,I)
REAL Y(20),T(20),KJ(20),KJ5(20),K1(20),K15(20),K2(20),K3(20),AJ(20)
1) KJ(I)=X(I)/T(I)
2) KJ5(I)=((SQRT(AO(I))-SQRT(AO(I))-X(I)))#2.)/T(I)
3) K1(I)=ALOG(AO(I)/AJ(I))-X(I))/T(I)
4) K15(I)=((1./SQRT(AO(I))-X(I)))-(1./SQRT(AO(I)))#2.)/T(I)
5) K2(I)=X(I)/T(I)#AO(I)#AJ(I)-X(I))
6) K3(I)=((2.#AJ(I)#X(I)-X(I))#2)/(2.#T(I)#(AO(I)##2)#((AJ(I)-X(I)))##2)
7)
8)
9) RETURN
10) END

```

0 ERRO(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

PAG. 2 \* CPPPE-FORTRAN # 06/12/76

FIM DA COMPILACAO \* ESTAD OCUPADAS 448 DAS 8042 PALAVRAS DISPONIVEIS

// DUP

\*STORE WS UA CINDER OPERACAO COMPLETADA

// FOR

```

*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*ICJCS(250)RUFADER,1403PRINTER)
REAL X(20),T(20),KJ(20),KJ5(20),K1(20),K15(20),K2(20),K3(20),PCT(20)
10) AJ(20)=T(10)/T(1),PCT(I),I=1,6)
11) REAMAT(2F7.2)
12) FORMAT(5,10)(T(I),PCT(I),I=1,5)
13) WRITE(5,13)
14) WRITE(5,13)
15) FORMAT(1H,4X,'ZERO',8X,'MEID',7X,'JUM',7X,'UM E MEID',3X,'DJIS',5)
16) IX,'TRF',//)
17) DO 12 I=1,6
18) AJ(I)=PCT(I)/100.
19) X(I)=AJ(I)#AJ(I)/100.
20) CALL CINDER(AO,X,T,KJ,KJ5,K1,K15,K2,K3,I)
21) CONTINUE
22) WRITE(5,11)(KJ(I),KJ5(I),K1(I),K15(I),K2(I),K3(I),I=1,5)
23) FORMAT(6(IX,E10.3,1X)
24) CALL EXIT
25) END

```

0 ERRO(S) E 0 ADVERTENCIA(S) ASSINALADOS NESTA COMPILACAO

Observe que o REAL DIMENSIONALIZADO dispensa a declaração ... DIMENSION. Quando o argumento do programa principal é indexado, as variáveis da subrotina também deverão ser indexadas.



// XEQ 32.95  
 120.00 41.75  
 240.00 48.80  
 330.00 58.05

PAG. 3 \* COPPE-FORTRAN # 06/12/76

530.00 69.00  
 600.00 70.35  
 ZERO

	MEIO	UM	UM E MEIO	DOIS	TRES
1.372E-04	6.751E-04	-1.384E+04	1.649E-02	8.190E-02	2.010E+00
1.159E-04	6.882E-04	-1.172E-04	1.541E-02	7.953E-02	2.153E+00
1.016E-04	5.300E-04	-1.029E-04	1.481E-02	7.942E-02	2.345E+00
8.795E-05	4.774E-04	-8.925E-05	1.474E-02	8.386E-02	2.837E+00
6.509E-05	3.739E-04	-6.624E-05	1.343E-02	8.399E-02	3.549E+00
5.862E-05	3.394E-04	-5.968E-05	1.246E-02	7.908E-02	3.458E+00

FIM DA EXECUCAO

FORAM EXECUTADAS

72 INSTRUCCES

Observando estes resultados, notamos que os valores pertinentes às ordens de reação 'um e meio' e 'dois' são aparentemente constantes. Confrontando a diferença dos valores máximo e mínimo do primeiro caso com a do segundo, chegamos aos seguintes aspectos:

primeiro caso:  $1.64 - 1.24 = 0.40$  e  $0.40 / 1.24 = 0.32$   
 segundo caso:  $8.39 - 7.90 = 0.49$  e  $0.49 / 7.90 = 0.06$

Portanto, a variação do valor de k para a ordem 'dois' é bem menor. O uso do Método das Tabelas, neste caso, conduziria o pesquisador à obtenção de novos dados, e à aplicação de outros métodos, que confirmassem a ordem 'dois'.

// XEQ 32.95  
 120.00 41.75  
 180.00 48.80  
 240.00 58.05  
 330.00

PAG. 3 \* COPPE-FORTRAN # 06/12/76

530.00 69.00  
 600.00 70.35  
 ZERO

	MEIO	UM	UM E MEIO	DOIS	TRES
1.372E-04	6.751E-04	-1.384E-04	1.649E-02	8.190E-02	2.040E+00
1.159E-04	6.882E-04	-1.172E-04	1.541E-02	7.953E-02	2.153E+00
1.016E-04	5.300E-04	-1.029E-04	1.481E-02	7.942E-02	2.345E+00
8.795E-05	4.774E-04	-8.925E-05	1.474E-02	8.386E-02	2.837E+00
6.509E-05	3.739E-04	-6.624E-05	1.343E-02	8.399E-02	3.349E+00
5.862E-05	3.394E-04	-5.968E-05	1.246E-02	7.908E-02	3.458E+00

FIM DA EXECUCAO

FORAM EXECUTADAS 72 INSTRUCCES

Observando estes resultados, notamos que os valores pertinentes às ordens de reação 'um e meio' e 'dois' são aparentemente constantes. Confrontando a diferença dos valores máximo e mínimo do primeiro caso com a do segundo, chegamos aos seguintes aspectos:

primeiro caso: 1.64 - 1.24 = 0.40 e 0.40/ 1.24 = 0.32  
 segundo caso: 8.39 - 7.90 = 0.49 e 0.49/ 7.90 = 0.06

Portanto, a variação do valor de k para a ordem 'dois' é bem menor. O uso do Método das Tabelas, neste caso, conduziria o pesquisador à obtenção de novos dados, e à aplicação de outros métodos, que confirmassem a ordem 'dois'.

A Tabela 2 pode ser testada para os valores abaixo:

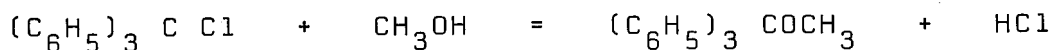
REACTION OF 0.106M TRITYL CHLORIDE WITH 0.054M METHANOL  
IN DRY BENZENE SOLUTION IN THE PRESENCE OF PYRIDINE AT 25°

(Swain<sup>9</sup>)

(Run 46, 0.064M pyridine; run 50, 0.108M pyridine; run 47, 0.215M  
pyridine.  $b/a = 1.963$ .)

Run	$t$ , min	$x$ , moles/liter	$x_{cor}$	$k_2$	$k_3$
50	20	0.0010	.....	.....	.....
47	22	0.0003	.....	.....	.....
46	22	0.0010	.....	.....	.....
47	168	0.0067	0.0091	0.0107	0.224
46	174	0.0086	0.0110	0.0127	0.278
47	418	0.0157	0.0181	0.0101	0.234
50	426	0.0165	0.0189	0.0105	0.248
46	444	0.0183	0.0207	0.0115	0.278
50	1,150	0.0294	0.0318	0.0089	0.272
47	1,440	0.0310	0.0334	0.0077	0.252
46	1,510	0.0321	0.0345	0.0080	0.264
50	1,660	0.0330	0.0354	0.0077	0.263
47	2,890	0.0394	0.0418	0.0066	0.296
46	2,900	0.0390	0.0414	0.0064	0.281
50	3,120	0.0392	0.0416	0.0060	0.269
47	193,000	0.0490	0.0514	.....	.....
					Av. 0.263

O Cloreto de Trifenil-Metila (Trytila) reage com Metanol, em solução de benzeno, do seguinte modo:



Piridina foi adicionada para remover o HCl e evitar a reação reversa. A vantagem desta adição de Piridina é que o produto resultante, Cloreto de Piridina, é levemente solúvel em Benzeno e precipita à medida em que a reação prossegue.

Após um certo tempo, cada amostra foi filtrada e o reagente Cloreto de Trytila não reagido foi hidrolizado com água, e o HCl / resultante foi titulado com NaOH padrão. Uma correção foi efetuada para a solubilidade do Cloreto de Piridina.

A estequiometria sugere a hipótese de que a reação seja de segunda ordem, mas os pesquisadores suspeitaram que a mesma fosse realmente de terceira ordem. Ou seja, primeira ordem com relação ao Cloreto de Trytila e segunda ordem com relação ao Metanol. Ambas as constantes de segunda e terceira ordem foram calculadas. É possível observar que a terceira ordem evidencia-se como razoável.

Nas expressões da Tabela 2,  $a_0$  é a concentração inicial de Metanol, em moles por litro;  $b_0$  é a concentração inicial de Cloreto de Trytila, em moles por litro; e  $x$  é a concentração de qualquer um destes reagentes, que reage no tempo  $t$ .

O programa resultante é apresentado na página seguinte.



```

B6700/87700  F O R T R A N  C O M P I L A T I O N  M A R K  2.8.001  MEDNES
FILE  8=CARTOES,UNIT=READER
      5=SAIDA,UNIT=PRINTER
CCCCC
PREZADO ESTUDANTE - CRIE VOCE MESMO A SUBROTINA 'CINEO'.
CCCCC
      REAL X(20),T(20),K(A(20)),KAB(20),KA2B(20),K2AB(20),KAAB(20)
      FEAC(8,5)(T(I),X(I),I=1,12)
      FORPA(2(1X,F9.4))
      WRITE(5,5)(T(I),X(I),I=1,12)
      DO 4 I=1,12
      AO=C.054
      BO=C.106
      XO=C.00001
CCCCC
      G VALGR ATRIBUIDO A 'XO' E APENAS SIMBOLICO, DESTITUIDO DE
      QUALQUE R SENTIDO PRATICO, MAS QUE MANTEM O PROGRAMA CORRETO.
      A EXPRESSAO PARA O CALCULO DE 'KABC(I)' FOI RETIRADA DO
      PROGRAMA, POIS NAO TEMOS NENHUM VALDR PARA 'CC'.
CCCCC
      1 F(AO-X(I))10.10.9
      5 KAC(I)=ALOG(AO+XO+K(I))/(XO+(AC-X(I)))/(T(I)+(AO-XO))
      10 KAC(I)=0.0
      11 IF(BO-X(I))11.11.12
      11 IF((AO-X(I))/(BO-X(I)))/(BO+(AO-X(I)))/(T(I)+(BO-AO))
      112 IF(KAB(I)=ALOG(AO+(BO-X(I)))/(AG-X(I)))/(BO+(AO-X(I)))/(T(I)+(BO-AO))
      117 IF((BO-2.1+X(I))/(AG-X(I)))/(BO+(AO-X(I)))/(T(I)+(BO-2.*AO))
      114 KA2EC(I)=ALOG(AO+(BO-2.*X(I)))/(BO+(AO-X(I)))/(T(I)+(BO-2.*AO))
      15 GO TO 17
      15 KA2EC(I)=0.0
      17 K2AEC(I)=(2.*X(I))/(T(I)+(BO-AO)+(AO-X(I)))+(2./(T(I)+(BO-AO)+
      1+2)))+(ALOG(BO+(AO-X(I)))/(AO+(BO-X(I))))
      1 KAAEC(I)=(X(I)/(T(I)+(BO-AO)+AC+(AO-X(I)))+(1./(T(I)+(BO-AO)+*2)
      1)))+(ALOG(BO+(AO-X(I)))/(AO+(BO-X(I))))
      8 CONTINUE
      7 WRITE(5,7)(KAC(I),KAB(I),KA2B(I),K2AB(I),KAAB(I),I=1,12)
      7 FORPA(5(1X,E12.5,1X))
      7 CALL EXIT
      END
00001000
00002000
00003000
00004000
00005000
00006000
00007000
00008000
00009000
00010000
00011000
00012000
00013000
00014000
00015000
00016000
00017000
00018000
00019000
00020000
00021000
00022000
00023000
00024000
00025000
00026000
00027000
00028000
00029000
00030000
00031000
00032000
00033000
00034000
00035000
00036000
00037000
00038000
00039000
00040000
00041000
00042000
CCCCC

```



## 15. BIBLIOGRAFIA

1. Chemical Engineering, July 29, 1968, pp. 151-156.
2. Dickson, T.R., "The Computer and Chemistry"  
W.H. Freeman and Company - San Francisco (1968).
3. McCracken, D.D., "FORTRAN with Engineering Applications"  
John Wiley & Sons, Inc. - NY (1967).
4. Aiba, S. e outros, "Biochemical Engineering", 2<sup>nd</sup> Ed.  
Academic Press, Inc. - NY & London (1973).
5. Hydrocarbon Processing, March 1972, pp. 112-114.
6. King, H.W. e outros, "Hydraulics", 5<sup>th</sup> Ed.  
John Wiley & Sons, Inc. - NY (1948).
7. Pacitti, T., "FORTRAN-MONITOR, Princípios", 3<sup>a</sup> Ed.  
Livros Técnicos e Científicos Editora S.A. - RJ (1974).
8. Laidler, K.J., "Chemical Kinetics", 2<sup>nd</sup> Ed.  
McGraw Hill Book Company - NY (1965).
9. Frost, A.A. & Pearson, R.G., "Kinetics and Mechanism", 2<sup>nd</sup> Ed.  
John Wiley & Sons - NY (1961)