# UNIVERSIDADE DE SÃO PAULO INSTITUTO DE CIÊNCIAS MATEMÁTICAS DE SÃO CARLOS

## FORMULAÇÃO MATEMÁTICA DOS PROBLEMAS DE PROCESSAMENTO DA INFORMAÇÃO

Dissertação apresentada ao ICMSC da USP, para obtenção do título de Mestre.

Fernão Stella de Rodrigues Germano

Orientador: Prof. Dr. Odelar L. Linhares

Dedico este trabalho à memória do Dr. Emygdio Rodrigues Germano e do Dr. Arlindo Vieira de Almeida Ramos, dois homens notáveis que marcaram minha existência, embora não tivesse eu tido a felicidade de, com eles, conviver: o primeiro me deu a vida e o segundo, a razão de vivê-la.

É apresentado um panorama histórico da evolução, da Teoria do Processamento da Informação com citação dos trabalhos considerados fundamentais e dos que apresentam bibliografia. São detalhados a Álgebra da Informação da CODASYL e o metodo gráfico de Young e Kent a formulação Abstrata dos problemas de Processamento de Dados. Para ilustração da Álgebra da Informação, é apre sentado o equacionamento do problema de montagem de pro gramas. São resolvidos dois problemas típicos de processamento de dados: a) Pagamento de pessoal e b) Faturamento, controle de vendas e cobrança, pelos dois meto dos, para efeito de comparação. É resolvido um problema de pesquisa de informação em tabelas: determinação dos pontos de sela de matrizes l) sem a formulação matemáti ca do problema, 2) com essa formulação e 3) de um modo sugerido pela literatura especializada, considerado oti mo. Realizado o processamento comparativo das três soluções, utilizando-se 4 matrizes de ordem 80, concluiuse que a solução 2 é extremamente mais eficiente (cerca de 1000 vezes) em tempo e aproveitamento da memoria, que a solução l e, aproximadamente, equivalente a solução 3.

#### **ABSTRACT**

A revue of advances towards a theory of Information Processing is presented with citation of both basic papers and those that include bibliography.CODASYL Information Algebra and Young & Kent Graphic Method for the Abstract Formulation of Data Processing Problems are de tailled. An Assemby Program Specification in Information Algebra is presented for illustration purpose, Two typical Data Processing problems: a) Payroll, b) Billing Sales Control are formulated by both methods, for comparison. A problem of table search: that of determining the saddle points in a matrix is solved 1) Without the mathematical formulation of the problem, 2) with this for mulation and 3) using an approach suggested by the literature, considered to be an optimum. After processing the three solutions using four different matrices, of order 80, results indicate that solutions 2 and 3 are comparable and their performance, both in time and memory ocupa tion is considerably (arround 1000 times) better than that of solution 1.

#### **AGRADECIMENTOS**

Ao Prof.Dr.Odelar Leite Linhares, pela efetiva orientação a este trabalho e a todas as minhas atividades de pos-graduação.

Ao Prof.Dr. Nelson Onuchic e à Profa. Izette Alves Coelho Loibel, por me terem ajudado a vencer uma das mais difíceis etapas que conduziram a este trabalho.

Ao Prof.Alfredo Américo Hamar e ao Prof.Oswaldo Sangiorgi que, na Escola de Comunicações e Artes da USP, me acolheram e incentivaram no ensino da Informática.

Ao Eng. Hermínio Amorim Jr. que, na Rede Ferroviária Federal S/A, me deu oportunidade de conviver com problemas administrativos que despertaram meu interesse para o assunto desse trabalho.

Ao Pe.Arthur Alonso S.J. que, jã em 1960, me propiciou, na Pontifícia Universidade Católica do Rio de Janeiro, a oportunidade de travar contato com computadores.

A meu irmão Helio Rodrigues Germano, que, em 1950, me in<u>i</u> ciou nas técnicas relativas ao equipamento convencional de cartão perfurado.

Ao Prof.Antenor Braga Paraguassu e ao Eng.Carlos Roberto M.Ramos pelo interesse e leitura dos originais.

Em especial à Bibliotecária Maria Eunice Dória, que me as sessorou nas pesquisas bibliográficas e foi responsável pela apresentação gráfica deste trabalho, em suas várias fases. Para essa apresentação muito contribuiu a competência da Sra. Neide dos Reis, como operadora de equipamento de reprografia.

À Fundação de Amparo à Pesquisa do Estado de São Paulo - (FAPESP), pelo apoio que sempre tem dado ao Departamento de Ciências de Computação e Estatística, para que nele existam condições necessárias ao desenvolvimento eficiente de trabalhos de pesquisa e de formação de pessoal qualificado, em todos os níveis.

À CAPES e ao CNPq, pela permanente ajuda aos programas de pos-graduação do Instituto de Ciências Matemáticas, de São Carlos.

A CAPES, pelo auxílio concedido a publicação deste traba-

#### SUMERIO

İNTRODUÇÃO		3. Un problema de faturamento, controle
		de vendas e de cobrança
1. Resumo histórico	1	3.1 - Enunciado 87
2. Considerações gerais	5	3.2 - Solução algébrica 90
3. Apresentação	6	3.2.1 - Expressões correspondentes
•		ao problema 1 93
		3.2.2 - Expressões correspondentes
CAPÍTULO O - PRÉ-REQUISITOS MATEMÁTICOS		ao problema 2 98
************************	7	3.3 - Solução gráfica
		4. Conclusões
CAPÍTULO I - A·ÁLGEBRA DA INFORMAÇÃO		•
DA CODASYL		· · · · · · · · · · · · · · · · · · ·
		CAPÍTULO V - PESQUISA DE INFORMAÇÃO EM TABELAS
1. Introdução	11	
1.1 - Apresentação	11	1. Introdução
1.2 - Resumo histórico	11	2. Solução natural do problema
2. Espaços de propriedades	13	3. Formulação matemática do problema 119
3. Āreas e linhas	15	3.1 - Aplicação do método algébrico 120
4. Funções de áreas e funções de linhas	17	3.2 - Aplicação do método gráfico 123
5. Malhas e funções de malhas	20	4. Solução sugerida pela formulação matemáti-
6. Feixes e funções de feixes	24	ca do problema
· · · · · · · · · · · · · · · · · · ·		5. Solução considerada ótima para
		efeito de comparação
CAPITULO II - EQUACIONAMENTO DO PROBLEMA		6. Resultados do processamento 127
DE MONTAGEM DE PROGRAMAS	•	
1. Introdução	29	EIBLIOGRAFIA
2. Expressões correspondentes à fase 1	33	129
3. Expressões correspondentes à fase 2	37	147
	40	APÉNDICE - 'I - UNA ÁLGEBRA DA INFORMAÇÃO
4. Expressões correspondentes à fase 3 5. Expressões correspondentes à fase 4	41	(Language Structure Group of the
•	46	CODASYL Development Committee)
6. Expressões correspondentes à fase 5 7. Expressões correspondentes à fase 6		CODYSIT DEASTONMENT COMMITTEES
8. Expressões correspondentes à fase 7	49 51	1. Introdução 133
9. Conclusões	54	2. Espaços de propriedades
, .		3. Linhas e funções de linhas
•		4. Āreas e funções de areas
CAPÍTULO III - O MÉTODO GRÁFICO DE YOUNG . KENT		5. Feixes e funções de feixes
		6. Malhas e funções de malhas
1. Introducão	57	7. Ordenações de áreas
2. Componentes básicos dos problemas		3. Operações sobre FOLs
de processamento de dados	58	9. Construções e extensões da álgebra 145
2.1 - Conjuntos de informação	58	10. Glossário
2.2 - Documentos	59	
2.3 - Relações e condições	59	
•	68	influence of posturations policy of these
2.4 - Exigências operacionais	69	APÊNDICE - II - DOCUMENTAÇÃO DO PROGRAMA ELABORA DO PARA A COMPARAÇÃO DAS SOLUÇÕES
3. Marcha de solução	03	•
•		DADAS AO PROBLEMA DE DETERMINAÇÃO DOS PONTOS DE SELA DE MATRIZES.
ALBERTA TV - ANVELDAÇÃO DOS VETODOS		nos rontos de sela de maintans.
CAPÍTULO IV - COMPARAÇÃO DOS MÉTODOS		***************************************
1, Introdução	73	
2. Um problema de pagamento de pessoal	74	
2.1 - Enunciado	74	
2.2 - Solução algébrica	76	
2.3 m Salvana massica	79	त त न न न
4. 1 # 30110000 PTO1 (AA	7.72	

#### INTRODUÇÃO

#### 1 - RESUMO HISTÓRICO

No começo da década de 1960 foram feitos sérios esforços no sentido de ser desenvolvida uma teoria do processamento da informação. A motivação desses esforços foi, principalmente, o fato de que estavam sendo infrutíferas as tentativas de padronização, quer de computadores, quer de linguagens de programação, bem como de desenvolvimento de uma linguagem de programação efetivamente independente da máquina, a ser utilizada no processamento. Esses esforços foram tão notáveis que mereceram um relato, na série Advances in Computers, elaborado por William C.McGee (23).

Nesse relato, é apresentado o panorama relativo a quatro áreas de pesquisa:

- linguagens de programação processuais;
- linguagens de programação não processuais;
- metodos de organização e descrição de dados e
- desenvolvimentos em teoria do processamento de dados.

Quanto às linguagens de programação processuais, assim denominadas por terem sido desenvolvidas para descrever o processo de solução, isto é, os passos que deverão ser dados para a realização de determinada tarefa, McGee relata as características das linguagens FACT, COMTRAN e COBOL que haviam sido recentemente desenvolvidas, salientando a ênfase na padronização dada no desenvolvimento dessa última.

Quanto às linguagens de programação não processuais, assim denominadas por terem sido desenvolvidas, com a finalidade de descrever, não o processo de solução, mas sim as relações entre os dados e os resultados do problema, apresenta McGee a descrição do Sistema SHARE 7090 9PAC (10), que consiste de um conjunto de rotinas para processamento de dados, desenvolvidas pela organização SHARE de usuários de computador, com a preocupação da mais ampla generalização; da linguagem DETAB-X (5), desenvolvida pelo "Systems Group" da CODASYL (Conference on Data Systems Language) uti-

lizando tábuas de decisão; e da Algebraic Data System Language (16), desenvolvida por Lionello Lombardi, com base numa teoria e numa álgebra de arquivos.

Quanto aos métodos de organização e descrição de dados, McGee apresenta os conceitos de itens elementares, matrizes retangulares e árvores, que já estavam sendo utilizados, na ocasião.

Finalmente quanto ao desenvolvimento em teoria do processamento de dados, menciona McGee os esforços de Evans (6) baseados em tabuas de decisão e de Young e Kent (38), cuja característica e o reconhecimento de que dados de um determinado tipo podem ocorrer em diferentes localizações em um sistema e é preferível tratar essas ocorrências como casos individuais de uma ocorrência geral do que como eventos isolados. Menciona ainda os esforços do SHARE Committee on Theory of Information Handling (7) que esta beleceu os conceitos de entidade (objeto, pessoa ou ideia capaz de ser descrita com a finalidade de processamento de dados), propriedade (características em termos das quais as entidades descritas), medida (valores associados as propriedades) e datum que e a menor unidade de informação, definida como a terna (D, i, j) ou  $D_{ij}$  em que D  $\tilde{\mathbf{e}}$  a medida, i  $\tilde{\mathbf{e}}$  o indice da entidade e j o indice da propriedade. Todavia, apresenta com minúcias, a Álgebra da Informação desenvolvida pelo "Language Structure Group" da CODASYL (1) por considera-la capaz de um impacto consideravel, na direção a ser tomada, pelo desenvolvimento da teoria do processamento dados.

Nosso trabalho é feito com base no método gráfico de Young e Kent (38), na Álgebra da Informação da CODASYL (1) e num trabalho de aplicação desta última ao problema de montagem de um programa (11) elaborado pelo próprio McGee, juntamente com Jesse Katz.

Tinha razão McGee ao qualificar a importância que a Algebra da Informação teria, no desenvolvimento das pesquisas posteriomente elaboradas.

Em agosto de 1963, foi realizada uma conferência em Prince ton, New Jersey, organizada pela ACM e pelo X3.4.1 Grupo de Trabalho em Teoria da Linguagem da American Standarda Association com a finalidade, de acordo com S.Gorn (9), de reunir os pesquisadores que haviam desenvolvido esforços independentes, na área de teoria das linguagens mecânicas, a fim de que expusessem e debates-

sem seus métodos e suas conclusões, para que os desenvolvimentos posteriores se fizessem de maneira mais articulada. a essa conferência entre outros: R. Bosak, em cujo trabalho (2) baseada a Algebra da Informação da CODASYL; T.B. Steel que apresentou trabalho (34) em que se propunha abranger os desenvolvimentos alcançados tanto pelo Language Structure Group do CODASYL Development Committee, como pelo SHARE Theory of Information Handling Com mittee, e que ja havia antes publicado trabalhos (36,37). relatando os desenvolvimentos alcançados por esse último Committee; Lio nello Lombardi que apresentou trabalho (15) e que já havia além de (16) publicado outros sobre o assunto (17,18,19); bem como outros pesquisadores, cuja relação consta de (14), a quem se deve a mais significativa parcela do desenvolvimento alcançado, até o presente pela Teoria da Programação, bem como por suas ramificações e especializações, tais como Schemata, Analise de Programas, Verificação de Programas quanto à Correção, Definição Formal da Semântica e da Sintaxe das Linguagens de Progração, etc.

Em 1969, Jean E. Sammet, que era um dos membros do Language Structure Group da CODASYL, na época do desenvolvimento da Álgebra da Informação, afirmou (32) que a grande significação desse trabalho será evidenciada, quando mais for conhecido sobre a teoria e ciência da programação, em geral, e do processamento de dados, em particular, mencionando também os trabalhos de Young e Kent (38) Lionello Lombardi (16) e Katz e McGee (11).

Em 1972, a mesma Miss Sammet ao relatar (33) os desenvolvimentos alcançados na área de Linguagens de Programação, menciona a Álgebra da Informação e dá um panorama dos conhecimentos, hoje disponíveis, que foram fruto, em grande parte, do trabalho dos pesquisadores, relacionados em (14).

Uma análise da bibliografia publicada, desde o relato ela borado por McGee (23) até nossos dias, mostra como tal relato foca lizou trabalhos significativos que abriram caminhos seguidos em ou tros importantes trabalhos.

Em 1962, McGee apresentou um metodo (24) de projeto e processamento de arquivos baseado na Algebra da Informação. Em 1968, apresentou uma técnica (21) de representação de arquivos, utilizan do recursos da teoria dos grafos e em 1969 apresentou um levantamento (22) das linguagens existentes para o processamento de arquivos.

Em 1963, Borje Langefors apresentou uma abordagem (13)diferente, para o problema do desenvolvimento de uma teoria dos sistemas de informação, utilizando conceitos de Topologia.

Em setembro de 1964, foi realizada uma conferência sob os auspícios do Technical Committee 2 da International Federation for Information Processing cujos anais foram editados por Steel (35) que incluem uma bibliografia na qual são mencionados os trabalhos desenvolvidos na área de Linguagens Formais para Definição de Linguagens.

Também os trabalhos na área de tábuas de decisão prossegui ram, não so com o trabalho de Lombardi (15), mas com vários outros dos quais nos dão notícia dois livros, publicados em 1970 (8,20).

Importante trabalho (4) que utilizou conceitos desenvolvidos até 1969, foi a linguagem para descrição de dados, preparados pelo Data Base Task Group da CODASYL, cuja história é relatada por Patterson (30), juntamente com a do grupo de trabalho da SHARE/GUIDE que se dedicou à mesma área de estudos.

Paul Namian desenvolveu estudos teóricos (26,27,28) sobre sistemas de informação gerenciais (Management Information Systems), chegando a resultados análogos aos da Álgebra da Informação.

Ainda dentro da aplicação dos recursos da Álgebra e da Topologia à Teoria da Programação para Computadores, duas teses de doutoramento (3,29) foram, recentemente, apresentadas à Universidade de Purdue.

Quanto ao progresso na área de Métodos de Organização e Descrição de Dados, a série "Advances in Computers" em 1972 apresentou um relato (31) preparado por David C.Roberts, do desenvolvimento obtido até então, em que são mencionados os estudos de McGee (21 e 24), Lombardi (19), Knuth (12), Namian (26), SHARE (7) e outros num total de 203.

Tendo em vista a grande quantidade de referências biblio gráficas, reunidas durante a preparação de nossa dissertação, decidimos incluir, na bibliografia, apenas os trabalhos citados, reservando o material reunido, para publicação posterior de uma bibliografia extensiva, sobre o assunto.

#### 2 - CONSIDERAÇÕES GERAIS

Os problemas de processamento de dados, tanto de natureza científica, como comercial, bem como aqueles ligados aos sistemas de informação, têm usualmente sido resolvidos visando a obtenção de um processo de solução apresentado, em geral, sob a forma de um fluxograma, que depois é codificado em linguagem de programação, dando origem a uma sequência de instruções executáveis pelo computador.

Tal modo de encarar os problemas é uma decorrência natural da concepção dos sistemas computacionais, cuja linguagem de ma quina exige a descrição, passo a passo, do processo de solução. A grande maioria das linguagens de programação foi projetada, igualmente, para a descrição de tipo processual da solução dos problemas.

Katz e McGee (11) apresentam tres razões para justificar a adoção dessa concepção e opção de projeto: 1) Razões econômicas orientaram a concepção de modo sequencial de operação para os computadores, obrigando, assim, a que seu comando fosse dado, passo a passo, através das chamadas instruções de máquina. 2) Muitos processos computacionais são de natureza intrinsecamente sequencial, nos quais as operações devem ser efetuadas em uma certa ordem, como, por exemplo, no caso dos algorítmos numéricos, o que, certamente, influiu na opção de projeto adotada na criação da linguagem FORTRAN, ALGOL, etc. 3) A programação passo a passo é, muitas vezes, utilizada independentemente da natureza sequencial do problema, pois limita as coisas que devem ser consideradas, simultaneamente, numa certa fase da solução do problema, facilitando, assim, o raciocínio de quem o está resolvendo.

Algumas desvantagens de grande importância são apresenta das por esse tipo de solução processual, entre as quais situam-se:

1) A dificuldade de dar solução a um problema totalmente desvinculada das peculiariedades do computador a ser utilizado. 2) A dificuldade de reconhecer problemas e soluções Canalogo, consequentemente duplicando esforços em muitas áreas de atividades. 3) A dificuldade de otimizar a solução de um problema, através da comparação de diferentes soluções, antes de processa-las.

Os trabalhos de Young e Kent (\$3) Se do Language Structure Group da CODASYL(1) além de constituirem um marco no desenvolvi mento da teoria do processamento da informação, apresentam métodos para relacionar os resultados aos dados, independentemente do processo de solução, nos problemas de processamento da informação. Es sa a razão pela qual baseamos nosso trabalho nesses métodos.

## 3 - APRESENTAÇÃO

No capítulo zero, apresentamos os pré-requisitos matemáticos que, a nosso ver, seriam necessários, para que este trabalho pudesse ser lido por não matemáticos.

No Capítulo 1, apresentamos a Algebra da Informação da CO-DASYL (1), cuja exposição formal está contida no Apêndice I.

No Capítulo 2, aplicamos a Algebra da CODASYL ao problema da montagem de programas, e obtemos a solução relacional desse problema, constituída pelas 19 expressões, que podem ser observadas, nas páginas 55 e 56.

No Capítulo 3, apresentamos o Método Gráfico de Young e Kent (38) utilizando a notação resumida, na página 72.

No Capítulo 4, resolvemos dois problemas típicos de proces samento comercial da informação, por ambos os metodos, de modo a permitir a sua comparação, chegando às conclusões relatadas, à pagina 114.

No Capítulo 5, resolvemos um problema típico de processamento da informação, relativo à pesquisa de informações em tabelas, com e sem o auxílio dos recursos constituídos por esses métodos, com parando os resultados com uma solução considerada ótima para o problema e chegando às conclusões relatadas à página 127.

Na Bibliografia, apresentamos 38 referências relativas aos trabalhos citados nesta introdução e nos vários capítulos que se se guem.

No Apêndice I, conforme ja mencionamos, apresentamos a exposição formal da Álgebra da Informação da CODASYL.

No Apêndice II, figura a documentação dos programas utilizados na preparação do Capítulo 5.

#### CAPÍTULO O

## PRÉ-REQUISITOS MATEMÁTICOS

Apresentamos a seguir uma relação dos assuntos com os quais o leitor deverá estar familiarizado, para poder acompanhar com todos os pormenores,os assuntos apresentados, no presente trabalho.

Em relação a cada um desses assuntos, mencionamos os conceitos efetivamente utilizados, quer na Álgebra da Informação da CODASYL, quer no método gráfico de Young e Kent. A notação utilizada na primeira está reunida no glossário que se encontra, na página 147 e, neste último, na relação constante da página 72.

Noções sucintas de Lógica Matemática:
Proposição
Os conectivos e o modificador
Tabelas verdades
Quantificadores

Elementos da teoria dos conjuntos

Conjunto, elemento e relação de pertinência
Conjunto unitário e conjunto vazio
Conjunto universo
Sub-conjunto
Igualdade de conjuntos
Conjuntos de partes de um conjunto
Operações entre conjuntos
Conjunto complementar
Diagramas de Euler - Venn
Número de elementos de um conjunto finito

#### Produto Cartesiano:

Par ordenado
Produto Cartesiano de dois conjuntos
Representação gráfica do produto cartesiano
n-upla ordenada
Produto Cartesiano de n conjuntos

#### Relações

Relação como sub-conjunto do produto cartesiano de dois conjuntos.

Domínio e imagem
Conjunto de partida e conjunto de chegada
Relação sobre
Propriedades das relações sobre um conjunto:
reflexiva, simétrica, anti-simétrica e transitiva
Relação de equivalência
Equivalência modulo R
Classe de equivalência
Conjunto quociente
Partição
Relação de ordem
Ordem total é parcial
Ordem estrita
Lei da tricotomia

Aplicações ou Funções

Aplicação como caso particular de relação
Domínio e Contra-domínimo
Imagem
Igualdade de duas aplicações
Homomorfismo
Isomorfismo

Teorema: "Toda a função determina uma relação de equivalência definida como aRb ↔ f(a) = f(b)

Esses assuntos podem ser encontrados em livros didáticos que cuidam de fundamentos da Matemática, Teoria dos conjuntos ou Noções de Álgebra. Como referência, citamos os seguintes livros, a-grupados em 2 categorias:

a) Livros elementares:

MONJALLON, A. - Introduction aux Mathématiques Modernes Paris, 1961

ZEHNA - JOHNSON - Elements of Set Theory - Boston 1962 SELBY - SWEET - Sets, Relations, Functions - New York, 1963. b) Livros mais avançados
BOURBAKI, N. - Théorie des ensembles: Fascicule de Résultats. Paris, Hermann & Cie. Editeurs, 1951.
BOURBAKI, N. - Algèbre Linéaire - Paris, 1955
FARAH, E. - Teoria dos Conjuntos - São Paulo, 1961.
HALMOS, P.R. - Naive Set Theory
BIRKHOFF - MACLANE - Modern Algebra - New York, 1944.

#### Matriz

Chamamos de matriz uma aplicação definida no produto cartesiano de dois conjuntos finitos e que toma valores em outro conjunto qualquer.

Sejam, por exemplo, os conjuntos A = {1,2,3} e B = {1,2}.

O produto cartesiano A×B desses conjuntos será o conjunto A×B =

{(1,1), (1,2), (2,1), (2.2), (3,1), (3,2)}. Seja C = {4,7,9} o ou

tro conjunto. Podemos definir a seguinte aplicação (matriz) em

A×B, tomando valores em C:

$$\{((1,1),4),((1,2),4),((2,1),7),((2,2),9),((3,1)4),((3,2),7)\}$$

Essa aplicação (matriz) pode ser escrita mais simplesmente,como segue:

$$\begin{pmatrix} 4 & 4 \\ 7 & 9 \\ 4 & 7 \end{pmatrix}$$

Assim,

$$\begin{pmatrix} 3 & 5 & 1 \\ 2 & 4 & 6 \end{pmatrix}$$

e um exemplo de matriz definida no produto cartesiano dos conjunto  $B = \{1,2\}$  e  $A = \{1,2,3\}$  e que toma valores no conjunto  $D = \{1,2,3,4,5,6\}$ 

Outro exemplo de matriz definida em  $B\times A$ , tomando valor em D  $\tilde{a}$ :

$$\begin{pmatrix}
3 & 5 & 1 \\
4 & 1 & 3
\end{pmatrix}$$

Essa última também pode ser escrita através dos pares or denados:

((1,1),3)

((1,2),5)

((1,3),1)

((2,1),4)

((2,2),1)

((2,3),3)

É usual chamar de elemento da matriz apenas o segundo ele mento do par ordenado. Nos chamaremos de elemento da matriz, o par ordenado, chamando o segundo elemento do par de "valor do elemento".

Assim, a cada elemento da matriz, podemos associar uma ter na ordenada, cujo primeiro elemento sera a "linha do elemento da matriz", o segundo, a "coluna do elemento da matriz" e o terceiro, o "valor do elemento da matriz".

Então a matriz acima poderia ser escrita:

(1,1,3)

(1,2,5)

(1,3,1)

(2,1,4)

(2,2,1)

(2,3,3)

## CAPÍTULO I

## A ÁLGEBRA DA INFORMAÇÃO DA CODASYL

## 1 - INTRODUÇÃO

#### 1.1- APRESENTAÇÃO

Neste capítulo, apresentamos os conceitos da Álgebra da Informação introduzida pela CODASYL, cuja exposição formal está contida, no Apêndice I.

Visam a apresentar a Algebra quanto as suas possibilidades de aplicação aos Sistemas de Informação mais conhecidos, e são enunciados com a finalidade de ilustrar e explicar os conceitos for mais, apresentados no Apêndice I.

Assim, a leitura deste capítulo deve ser feita com frequentes consultas ao apêndice mencionado. Não faremos, no decorrer da exposição, referências específicas a essa necessidade, pois sua frequência iria obrigar a uma enfadonha repetição. Nem faremos referência aos pré-requisitos matemáticos, pois o leitor, quando deles tiver necessidade, poderá consultar o Capítulo O.

#### 1.2- RESUMO HISTÓRICO

Em Maio de 1959, no Departamento de Defesa dos Estados Unidos da América foi realizada uma reunião de fabricantes, usuários e entidades governamentais interessadas em computadores, na qual foi criada a Conferênce ou DAta SYstems Languages que passou a ser conhecida como CODASYL. Do grupo, foi destacado um comitê executivo que continuou a existir, apesar de os 40 representantes presentes à primeira reunião não mais se haverem reunido.

Um dos principais frutos da atuação da CODASYL foi o desenvolvimento da Linguagem COBOL (COmon Business Oriented Language) cuja primeira versão foi completada, em Dezembro de 1959.

Em Julho de 1959, foi formado o Language Structure Group (LSG) do Development Comitee da CODASYL, para estudar a estrutura das linguagens de programação, para processamento de dados, e fazer recomendações para seu desenvolvimento futuro. Segundo SAMMET (32), reconhecendo as limitações das linguagens de programação, quanto: à dependência do equipamento, para o qual foram desenvolvidas,

a necessidade de instruções sequenciais de tipo processual e a problemas de divisão de aplicações de grande porte em processamentos separados, o LSG decidiu tentar desenvolver uma linguagem para ser vir de base teórica à programação automática. Como resultado de tal esforço, foi desenvolvida a Álgebra da Informação, a partir dos trabalhos de Robert Bosak da System Development Corporation, divulgada pelo Phase I Report do LSG (1), publicado em 1962.

As principais restrições a esse trabalho são relacionadas pela própria Miss SAMMET, que era um dos membros do LSG como sendo: O LSG não produziu uma linguagem, orientada, para o usuário, apropriada à definição de problemas, nem conseguiu especificar, mesmo em termos gerais, um algoritmo para traduzir as relações da Álgebra da Informação em programas em linguagem de máquina.

Katz e McGee(11), já em 1963, apresentaram a aplicação da Álgebra da Informação a um problema de processamento bastante elaborado, qual seja o da montagem de um programa, que ilustra fartamente a maneira de utilizar esse recurso de programação não processual. Nesse trabalho, os dados, (isto é, o programa escrito em linguagem ASSEMBLER) são relacionados aos resultados, (isto é, o programa objeto em linguagem de máquina e mensagens de erro) através de um conjunto de 19 expressões (pg.55), escritas na Álgebra da Informação. Assim, esse recurso cuja eficiência nos problemas de processamento comercial já havia sido evidenciada, no Relatório de Apresentação (1), teve sua aplicabilidade estendida a uma classe muito mais ampla de processamentos, sendo mesmo sugerida a possibilidade de aplicação à programação de "software".

Namian (26), em 1968, apresentou uma abordagem diferente, aplicada aos Sistemas de Informação Gerencial (Management Information System - MIS) na qual é introduzido o fator tempo. Apesar da diferença de abordagem, chegou ao final, como o proprio Namian relata, (à maioria das operações integrantes da Álgebra da Informação), apenas com uma diferente classificação.

Na mesma linha de pesquisa, da formulação abstrata dos problemas de processamento, utilizando os recursos da Álgebra, Lombardi (16), em 1962, já havia divulgado uma Álgebra de Arquivos que permite, por exemplo, expressar através de uma fórmula, certo méto do de ordenação. Tal fórmula é independente do equipamento e da se quência de realização da tarefa e, até certo ponto, até do processo de solução utilizado para resolver o problema.

#### 2 - ESPAÇOS DE PROPRIEDADES

Conforme a aplicação, em estudo, deve-se definir um espaço de propriedades que reflita as características do problema.

Para definir esse espaço, deve-se escolher um conjunto de propriedades e, para cada propriedade, um conjunto de valores, de modo a poder representar, adequadamente, as entidades envolvidas.

Por exemplo, num faturamento, cada cliente pode ser considerado uma entidade e o conjunto de propriedades escolhido, pode ser {Razão social da firma, Endereço, Número do CGC, Capital Registrado, Conceito comercial}.

O conjunto de valores de cada propriedade deve incluir to dos os valores que possam ocorrer não só para o conjunto de entida des existentes em certo período de tempo, mas para outras que pos sam vir a ingressar, posteriormente, no sistema.

Assim, o conjunto de valores da propriedade "Endereço" do nosso exemplo, não deve ser definido como o conjunto formado pelos endereços dos atuais clientes, mas como o formado por todos os Arranjos dos caracteres alfa-numéricos disponíveis, com tamanho adequado a abranger até a menor abreviatura exequível do mais longo endereço que possa ocorrer.

O conjunto de valores da propriedade "Conceito comercial" ja poderia ser constituído apenas de poucos valores. Por exemplo, poderia ser o conjunto {Excelente, Bom, Regular, Ruim, Pessimo} em que cada classificação dessas teria certa especificação, para que o cliente fosse nela enquadrado.

O conjunto de propriedades distintas escolhido para definir um espaço de propriedades constitui o conjunto de coordenadas desse espaço.

Espaço de propriedades é o conjunto de todos os pontos que podem ser determinados, tomando-se um valor do conjunto de valores, de cada propriedade, que forma o conjunto de coordenadas. Em nosso exemplo, um desses pontos poderia ser: (IRMÃOS SILVA LTDA., Rua das Cassias 315, 517947134, CR\$ 10.000,00, Bom).

Vejamos outro exemplo, para ilustrar melhor. Num Controle de Estoque, podemos definir um Espaço de Propriedades com o Conjunto de Coordenadas {Nº da Peça, Custo Unitário, Estoque Mínimo}. Cada peça constituiria uma entidade e os conjuntos de valores das propriedades poderiam ser:

V<sub>N</sub> da Peça = {inteiros de 1 a 999}

V custo unitário [números com duas casas decimais de 0,01 a 9,99],

V<sub>Estoque mínimo</sub>={inteiros de 1 a 99}.

Um ponto desse espaço poderia ser: (864 , 2,66 , 57), que representaria a peça 864 que custa CR\$ 2,66 e cujo estoque mínimo  $\tilde{\epsilon}$  57.

O espaço todo seria formado por todos os pontos possíveis. Enquanto que poderíamos ter apenas 999 peças diferentes, para determinar o número de pontos do espaço, temos que considerar que ca da peça pode vir a custar qualquer preço entre Or\$ 0,01 e Cr\$ 9,99 e seu estoque mínimo pode vir a ser estabelecido em qualquer valor, entre 1 e 99, ainda que, em dado instante, certa peça tem apenas certo custo e determinado estoque mínimo. Assim, nosso espaço teria 999×999×99=97812099 pontos.

A fim de atender ao Postulado 3 da Álgebra da Informação, devemos reservar um dos valores, do conjunto de valores da propriedade, para significar "irrelevante" e outro para "desconhecido". Em nosso exemplo, poderíamos usar o seguinte código:

- 998 Número da peça irrelevante
- 999 Número da peça desconhecido
- 9.98 Custo unitário irrelevante.
- 9,99 Custo unitário desconhecido
  - 98 Estoque minimo irrelevante
  - 99 Estoque mínimo desconhecido

Ficaríamos, assim, apenas com as seguintes faixas utiliz $\underline{\tilde{a}}$  veis para valores relevantes, conhecidos

Número da peça 001 a 997 Custo unitário 0,01 a 9,97 Estoque mínimo 01 a 97

Em qualquer aplicação prática será muito grande o número de pontos do espaço de propriedades que venha a ser definido, mas é evidente que, dele apenas pequena parcela, estará efetivamente envolvida, em determinado processamento.

A dimensão de um espaço de propriedades é dada pelo número de propriedades que constitui o conjunto de coordenadas desse espaço. Assim, o espaço que definimos para o nosso exemplo de faturamento é de dimensão 5 e o que definimos para o exemplo de Contro

le de Estoque e de dimensão 3.

Cada entidade será representada por apenas um ponto, em determinado espaço de propriedades, ainda que um deles possa representar várias entidades. Quando cada ponto representa apenas uma en tidade, o espaço é discriminatório para aquele conjunto de entidades.

Em nosso exemplo de faturamento, o espaço de propriedades e o correspondente conjunto de coordenadas é discriminatório. Vários subconjuntos desse conjunto de coordenadas seriam também discriminatórios, como por exemplo: {Endereço, Número do CGC, Capital Registrado, Conceito Comercial}, {Número do CGC, Razão social da Firma}, {Número do CGC, Endereço} etc, pois não há duas firmas com o mesmo número do CGC. Já o conjunto de coordenadas {Endereço, Capital Registrado} poderia ser discriminatório para certo conjunto de firmas e não para outro, em que duas firmas, com igual capital, tivessem o mesmo endereço. O conjunto de coordenadas discriminatório básico, em nosso exemplo, seria o conjunto constituído pelo único elemento {Número do CGC}.

As operações sobre conjuntos: União, Interseção e Complementação aplicam-se aos conjuntos de coordenadas, bem como aos conjuntos de pontos de determinado espaço de propriedades.

#### 3 - AREAS E LINHAS

Uma das mais importantes características da Álgebra da CODASYL é a de nos dar um modo de fazer referência e manipular conjuntos de pontos, em um espaço de propriedades.

Um arquivo, em processamento de dados, pode ser representado por um conjunto de pontos em um espaço de propriedades. Esse método de representação é bastante geral e desvinculado de qualquer peculiaridade do processamento. Assim, por exemplo, em nada influira na representação, o fato de o arquivo ser colocado em fita magnética, disco, ou na memória central do computador, e, muito menos do tipo de computador, em que vai ser processado.

O processamento de dados é constituido pela obtenção de certos relatórios ou arquivos de saída, a partir de certos arquivos de entrada. Assim, isso corresponde a transformações de dadas áreas em outras novas, de acordo com a finalidade do processamento.

Essas transformações independem, todavia, da sequência de passos do processamento. Pode-se, inclusive, raciocinar como se essas trans-formações estivessem todas ocorrendo, simultaneamente. Daí poder-mos estabelecer expressões para representar essas transformações, com as quais vamos descrever relações e não processos para obter resultados a partir de dados.

Sendo uma área, um conjunto de pontos, num espaço de propriedades, será através de áreas que faremos a representação dos ar quivos na Álgebra da Informação.

Nas paginas 32, 76, 92 e 120 apresentamos varios exemplos de espaços de propriedades, cada um com diversas areas, nele estabelecidas.

Podemos, por outro lado, formar, no espaço de propriedades conjuntos de pontos em que a ordem é significativa. Nesse caso, o conjunto constitui uma linha.

Em nosso exemplo de controle de estoque, tomemos 5 pontos dos 97812099 que formam o espaço de propriedades. Sejam eles, por exemplo:

 $q_1$   $q_2$   $q_3$ A = (861 , 5,22 , 43)

B = (641 , 4,25 , 50)

C = (531 , 3,70 , 45)

D = (964 , 2,20 , 50)

E = (841 , 2,40 , 40)

Com eles podemos formar uma area que os abranja a todos, ou as seguintes linhas que os englobem todos:

ABCDE , ABCED , ABDEC , ABDCE , ABECD , ABEDC , ACBDE , ACBED, etc.

Se quisermos calcular a quantidade de linhas de ordem de grandeza 5, isto é, de 5 pontos que com eles podemos formar, basta rá calcular a quantidade de Permutações de 5 elementos que é igual a 120.

Com esses 5 pontos podemos formar 5 linhas de ordem de grandeza 1, por exemplo a linha B; 20 linhas de ordem de grandeza 2, isto  $\tilde{\epsilon}$ ,  $A_5^2$ , por exemplo a linha CD;  $A_5^3=60$  linhas de ordem de grandeza 3, por exemplo a linha EBC;  $A_5^4=120$  linhas de ordem de grandeza 4, por exemplo a linha CBDE.

Assim, o total de linhas, que se pode formar com os 5 po $\underline{\mathbf{n}}$ 

tos dados, é de 325.

Podemos, outrossim, formar com esses pontos 5 áreas de um só ponto, por exemplo, C; 10 áreas de 2 pontos, isto é,  $C_5^2$ , por exemplo a área DE;  $C_5^3$ =10 áreas de 3 pontos, por exemplo a área ACD;  $C_5^4$ =5 áreas de 4 pontos, por exemplo ABDE.

Assim, o total de areas que se pode formar com os 5 pontos dados é de 31.

Note-se que um ponto pode ser considerado como uma linha ou como uma area, conforme a conveniência.

Resumindo, podemos dizer que:

- a) A quantidade de areas que se pode formar com m pontos  $\tilde{\mathbf{e}}$  dada por  $\sum_{k=1}^{m} C_{m}^{k} = 2^{m} - 1$
- b) A quantidade de linhas que se pode formar com m pontos  $\stackrel{m}{\text{e}} \text{ dada por } \stackrel{m}{\sum} \stackrel{A}{\stackrel{k}{m}} _{k=1}^{k}$
- c) A quantidade de linhas de ordem de grandeza n que se pode formar com m pontos  $\tilde{e}$  dada por  $A_m^n = \frac{m!}{(m-n)!}$
- d) A quantidade de areas de n pontos que se pode formar com m pontos é dada por  $C_m^n = \frac{m!}{(m-n)!n!}$
- e) A quantidade de linhas de ordem de grandesa m que se pode formar com uma area de m pontos  $\tilde{\mathbf{e}}$  dada por  $\mathbf{P}_{\mathbf{m}}$ = m!

## 4 - FUNÇÕES DE ĀREAS E FUNÇÕES DE LINHAS

Essas funções permitem associar um valor a uma área ou a uma linha.

Em nosso exemplo de controle de estoque, podemos definir a FOA "estoque mínimo médio". Essa função teria para a área ABC o valor 46, já para a ABCDE teria o valor 45,6 enquanto que para BC teria o valor 47,5 e para a E teria o valor 40.

Nesse mesmo exemplo, podemos definir a FOL "valor do estoque mínimo do primeiro ponto da linha". Essa função teria para a linha CBA o valor 3,70×45=166,5, jã para a linha ACB, o valor 5,22×43=224,46, enquanto que, para a linha ADCBE valeria 5,22×43=224,46 e para a linha C, 3,70×45=166,5.

Poderíamos, também, ter definido a FOL "estoque mínimo médio". Essa função é independente da ordem dos pontos, na linha, isto é, tanto para a linha ABC, como para a BCA teria o valor 46. Os valores dessa FOL coincidem com os da FOA, acima definida, desde que a linha seja formada pelos mesmos pontos que a área correspondente, para todas as linhas que satisfaçam essa exigência. O teorema 4.4 mostra que é sempre possível definir uma FOL correspondente a uma FOA.

Observamos ainda que, podendo um ponto ser considerado uma linha ou uma área, podemos aplicar-lhe tanto FOAs como FOLs.

Como os conjuntos imagem de ambas as FOLs, acima definidas, podem ser considerados como contidos no: conjunto ordenado  $\Omega$ < 0<- $\infty$ < conjunto números reais < + $\infty$ , ambas podem ser consideradas OFOLs ou seja FOLs de ordem.

Podemos operar com FOLs em vários operadores, tanto lógicos como aritméticos, apresentados formalmente em 8, os quais nos vão permitir expressar as funções de linhas e as funções de áreas.

Considerando q<sub>i:j</sub> como sendo a propriedade q<sub>j</sub> para o i-é-simo ponto de uma linha, podemos definir uma FOL f=q<sub>i:j</sub>. Essa função associa a cada linha o valor de q<sub>j</sub> para o i-ésimo ponto da linha.

Com esse recurso a FOL "valor do estoque mínimo do primei ro ponto da linha" pode ser expressa como:

em que

- q<sub>1:2</sub> e a FOL que a cada linha associa o valor de q<sub>2</sub> para o seu primeiro ponto, isto e, o valor do custo unitario da peça correspondente.
- $q_{1:3}$  é a FOL que associa a cada linha o valor de  $q_3$  para o primeiro ponto da linha, isto é, o valor do estoque mínimo da peça.
- \* indica a operação multiplicação de FOLs.

Então f $_{20}$  vai associar a cada linha um valor que é o produto dos valores das FOLs q $_{1:2}$  e q $_{1:3}$  como desejamos.

Outro exemplo de função de áreas é a que associa a cada área um número que expressa a quantidade de pontos de que ela é formada. Podemos escrever essa função do seguinte modo:

$$f_1(A) = \Sigma 1$$

Assim, a FOA, anteriormente definida: "estoque minimo me-dio" pode ser escrita:

$$f_2(A) = \frac{\sum q_3}{\sum 1}$$

Ja a FOL "estoque minimo medio" seria escrita como:

$$f_{21}(L) = \frac{\sum_{i=1}^{n} q_{i:3}}{n}$$

em que n é o número de pontos da linha.

Outra FOA mais elaborada seria "valor do estoque mínimo". Podemos escrevê-la como:

$$f_3(A) = \Sigma q_2 * q_3$$

E teria, para as áreas abaixo, os seguintes valores:

Ārea	Pontos que a formam	valor de f <sub>3</sub>
A <sub>1</sub>	ABCDE	5,22×43+4,25×50++2,40×40=809,46
A <sub>2</sub> .	BDE	4,25×50+2,20×50+2,40×40=418,50
A <sub>3</sub>	ACD	5,22×43+3,70×45+2,20×50=500,96
A <sub>4</sub>	· <b>D</b>	2,20×50=110,00

Note-se que o valor associado por uma função não é neces sariamente numérico. Em nosso exemplo de faturamento, se tivermos agrupado os clientes em áreas (veremos no próximo item), de acordo com a propriedade "Conceito comercial", podemos definir a FOA.

$$f_{10}(A)=q_5$$

que, à area que agrupa os clientes classificados como "BOM" associara o valor "BOM"; à area que agrupa os clientes clas sificados como "RUIM", associara o valor "RUIM", e, assim por diante.

Note-se ainda que o Conjunto Imagem de uma função pode estar contido no conjunto de valores de uma das propriedades do Espaço, mas isso não é obrigatório. No exemplo acima, o conjunto imagem de  $f_{10}$  coincide com o conjunto de valores, da propriedade  $q_5$ ; já no anterior, o conjunto imagem de  $f_3$  não está contido em nenhum dos conjuntos de valores das propriedades daquele Espaço.

Lembrando o que foi dito anteriormente, que o processamen to de dados consiste em submeter os arquivos de entrada a transfor mações que resultem nos arquivos de saída, e que as áreas são a ma neira de representar arquivos, utilizando-se a Álgebra da Informação, verificamos que as funções de áreas são ferramenta preciosa para representar essas transformações.

#### 5 - MALHAS E FUNÇÕES DE MALHAS

Além de representar arquivos, áreas podem ser utilizadas para representar agrupamentos, dentro de um arquivo.

No item anterior, foi levantado o problema de, em um sistema de faturamento, agrupar os clientes em áreas, de acordo com a propriedade "conceito comercial". A solução desse problema é feita com o estabelecimento de malhas, na área que engloba os clientes.

Consideremos cada ponto dessa area, como constituindo uma linha de um só ponto. Então, podemos aplicar a cada ponto a FOL f=q5. Essa função associa a cada ponto o valor correspondente da propriedade q5 "CONCEITO COMERCIAL". A essa função fazemos corresponder a relação de equivalência R definida por ARB++f(A)=f(B), is to é, dois pontos são equivalentes se e somente se tiver o mesmo valor para a propriedade "CONCEITO COMERCIAL". A relação de equivalência R estabelece uma partição na área que engloba os clientes, de modo que cada classe de equivalência (elemento da malha) reune os clientes com mesmo "conceito comercial". A função f é chamada função que determina a malha.

Cada elemento da malha é constituído por um conjunto de pontos e pode, assim, ser considerado uma área, à qual, consequentemente, podem ser aplicadas funções de áreas.

Uma das principais finalidades do estabelecimento de malhas é a obtenção de novas áreas, a partir das existentes. No processamento de dados, corresponderia a obter arquivos que resumissem outros tantos. Essa obtenção é conseguida através da Função de uma malha que associa a cada elemento seu um ponto, através de um conjunto de FOAs. Como o ponto pertence ao espaço de propriedades, em que se está trabalhando, para definir a FOG temos necessidade de tantas FOAs quantas forem as coordenadas desse espaço. Por convenção, quan do não se explicita a FOA correspondente a uma coordenada, considera-se que ela é definida pela FOA que associa a todos os elementos o valor Ω.

Vamos entrar em maiores detalhes no problema de faturamento, a fim de melhor verificar o funcionamento das malhas e funções de malhas.

Vejamos alguns pontos desse espaço constituindo uma área A<sub>1</sub>. Para simplificar, consideremos que, nesse arquivo, seja irrelevante considerar a razão social da firma e o endereço, de vez que o cliente está perfeitamente determinado, pelo número do CGC. Temos, então, nessa área A<sub>1</sub>:

•	Razão social da firma	Endereço	N9 do CGC	Capital Registrado	Conceito Comercial	Contador
	q <sub>1</sub>	9 <sub>2</sub>	<sup>q</sup> 3	q <sub>4</sub>	<sup>q</sup> 5	<b>9</b> 6
P <sub>1</sub>	Ω	Ω	517947134	10.000,00	вом	Ω
P 2	Ω	Ω	517947128	15.000,00	RUIM	Ω
P 3	Ω	Ω	517947135	118.000,00	EXCELENTE	Ω
P 4	Ω	Ω	517947431	12.000,00	EXCELENTE	Ω
P <sub>S</sub>	Ω	Ω	517942133	23.500,00	PESSIMO	. Ω
P <sub>6</sub>	Ω	Ω	517946147	42.750,00	REGULAR	Ω
P 7	Ω	, Ω	517948112	15.000,00	REGULAR	Ω
P 8	Ω	Ω	517943114	41.214,00	RUIM	Ω
P 9	Ω	Ω	517941132	50.000,00	BOM	Ω
P <sub>10</sub>	Ω	. υ	517912413	75.000,00	EXCELENTE	: <b>Ω</b>
P <sub>11</sub>		Ω	517944512	47.500,00	вом	Ω
P <sub>.12</sub>	Ω	Ω	517944413	15.000,00	REGULAR	Ω
P <sub>13</sub>	Ω	Ω	517752218	42.750,00	REGULAR	Ω
P <sub>14</sub>	Ω	Ω	517572221	50.000,00	REGULAR	Ω

Utilizando como função que determina a malha f= $q_5$  obtemos a malha G=G( $q_5$  ,  $A_1$ ) cujos elementos são formados dos seguinte pontos:

Elemento da Malha Pontos que estão incluídos no elemento 
$$A_1^{(1)} = G(q_5, A_1)|_{q_5 = "EXCELENTE"}$$
  $P_3$ ,  $P_4$ ,  $P_{10}$   $A_1^{(2)} = G(q_5, A_1)|_{q_5 = "BOM"}$   $P_1$ ,  $P_9$ ,  $P_{11}$   $P_6$ ,  $P_7$ ,  $P_{12}$ ,  $P_{13}$ ,  $P_{14}$   $P_1$   $P_2$ ,  $P_3$   $P_4$   $P_5$   $P_5$   $P_5$   $P_5$   $P_5$ 

Podemos, agora, estabelecer uma função dessa malha que as socie a cada elemento um ponto, numa nova área em que  $q_1$ ,  $q_2$ ,  $q_3$  e  $q_4$  são irrelevantes, sendo apenas relevantes  $q_5$  e  $q_6$ . Notemos que fizemos uma modificação no Espaço de Propriedades introduzindo a coordenada  $q_6$ . Isso nos obrigaria a rever turo o que foi feito até este ponto, pois a modificação do espaço altera, totalmente, o problema. Chamemos essa nova área de  $A_2$ .

Enquanto cada ponto de  $A_1$  representa um cliente, cada ponto de  $A_2$  representará a quantidade de clientes com certa classificação, quanto ao "conceito comercial".

A função da malha que permite determinar  $\mathbf{A_2}$  a partir de  $\mathbf{A_1}$  é a função H.

$$A_2 = H(q_5, A_1)$$
 $H = (q_5^{\dagger} = q_5, q_6^{\dagger} = \Sigma 1)$ 

Como não mencionamos a função de áreas utilizada para determinar  $q_1'$  subentende-se que ela é a FOA que associa a qualquer área o valor  $\Omega$ . O mesmo vale para  $q_2'$ ,  $q_3'$  e  $q_4'$ . O valor da coorde nada  $q_5$  do ponto correspondente a cada elemento da malha é dado pe la FOA  $q_5$ , isto é, a função que associa a cada elemento, o valor da coordenada  $q_5$  de todos os seus pontos. (Essa FOA só tem sentido, porque todos os pontos de certo elemento têm o mesmo valor de  $q_5$ ). A FOA, usada para definir  $q_6$  do ponto imagem de cada elemento da malha, é aquela que conta o número de pontos que constituem o elemento. Temos, então,  $A_2$  constituída dos seguintes pontos:

**	Razão social da firma	Ende- reço	N9 do CGC	Capital Registrado	Conceito Comercial	Contador
٠	q <sub>1</sub>	<sup>q</sup> <sub>2</sub>	<sup>q</sup> 3	94	95	<sup>9</sup> 6
P <sub>100</sub>	$\mathbf{\Omega}$	Ω	Ω	Ω	вой	3
P <sub>200</sub>	Ω	Ω.	Ω	Ω	EXCELENTE	3
P <sub>300</sub>	Ω	Ω	Ω	Ω	PESSIMO	1
P <sub>400</sub>	Ω	Ω	Ω	Ω	RUIM	2
P <sub>500</sub>	Ω	Ω	Ω	- Ω	REGULAR	5

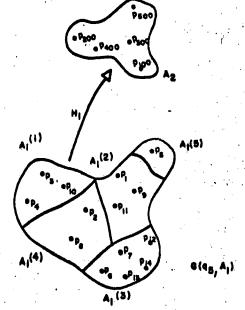
Em suma, o que foi feito através da malha G e da FOG H foi resumir o arquivo  $A_1$  e criar o arquivo  $A_2$ , através de transforma - ções expressáveis pela Álgebra da Informação.

A FOG H associa, a cada elemento da malha estabelecida em  ${\bf A}_1$ , um ponto de  ${\bf A}_2$ :

Elemento da Malha	Ponto de A <sub>2</sub> correspondente
A <sub>1</sub> (1)	P <sub>200</sub>
A <sub>1</sub> <sup>(2)</sup>	P <sub>100</sub>
A <sub>1</sub> (3)	P <sub>500</sub>
A <sub>1</sub> (4)	P400
A <sub>1</sub> (5)	P300

Esquematicamente, podemos figurar esse procedimento, da



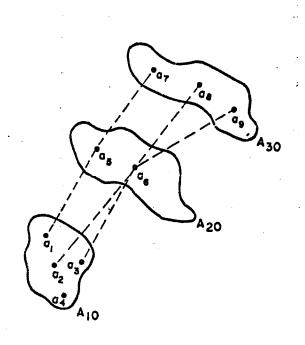


## 6 - FEIXES E FUNÇÕES DE FEIXES

Já vimos como criar novo arquivo a partir de um existente. Resta, agora, estudar o problema de como criar novo arquivo, levan do em conta informações que constam de mais de um deles.

Em outras palavras, trata-se de obter nova área, colhendo informações em várias outras.

Para isso, consideremos as areas em que vamos recolher in formação, formando um conjunto ordenado. Essas areas não são neces sariamente disjuntas, ainda que, na maioria dos casos,o sejam. Con sideremos por exemplo o conjunto de areas  $\{A_{10}, A_{20}, A_{30}\}$  que evi-



dentemente difere do conjunto  $\{A_{20}, A_{10}, A_{30}\}$  ou do conjunto  $\{A_{30}, A_{20}, A_{10}\}$ .

Vamos, agora, formar linhas de or dem de grandeza igual ao número de elementos do conjunto de áreas, tais que cada linha seja formada tomando um ponto de cada area do conjunto.

Alguns exemplos seriam:

a<sub>1</sub> a<sub>5</sub> a<sub>7</sub>

a<sub>2</sub> a<sub>6</sub> a<sub>8</sub>

a<sub>3</sub> a<sub>6</sub> a<sub>9</sub>

a<sub>3</sub> a<sub>6</sub> a

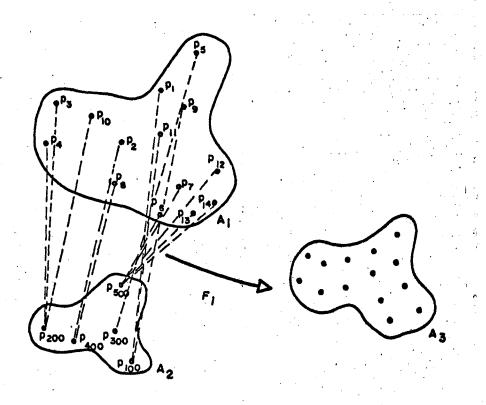
Como se vê, o primeiro ponto de cada linha é sempre pertencente à primeira área do conjunto; o segundo à segunda e, assim por diante.

Com<sup>5</sup> essa sistemática, poderiamos formar  $4\times2\times3=24$  linhas diferentes, em nosso conjunto de áreas  $\{A_{10}, A_{20}, A_{30}\}$ . De modo geral, dado um conjunto de n áreas em que mero de pontos que constitui cada área, pode-se formar  $\Pi$  mero diferentes nesse i=1 conjunto.  $\Pi$  significa produtório.

Dessas linhas, pode-se selecionar um feixe que englobara aquelas que satisfaçam determinada condição. Essa escolha é feita

através de uma OFOL seleção que exprima a condição desejada. As linhas que constituiriam o feixe seriam aquelas para as quais a OFOL de seleção teria o valor "verdadeiro".

Voltemos ao nosso exemplo de faturamento para esclarecer melhor essas noções. Tomemos o conjunto de áreas  $\{A_2, A_1\}$ , formado das áreas definidas no item anterior.



Nesse conjunto, podemos formar  $5\times14=70$  linhas, nas quais o primeiro ponto pertence a  $A_2$  e o segundo a  $A_1$ . Vejamos alguns e-xemplos dessas linhas:

$$L_1 = P_{200} P_{13}$$
 $L_2 = P_{100} P_1$ 
 $L_3 = P_{100} P_8$ 
 $L_4 = P_{500} P_{12}$ 

Vamos estabelecer a condição em que o conceito comercial do ponto de  $A_2$  seja igual ao do ponto  $A_1$ . A OFOL de seleção, que exprime essa condição, pode ser expressa por

Essa seleção permite formar um feixe constituído apenas

das linhas para as quais  $q_{1:5} = q_{2:5}$  seja verdade, isto  $\tilde{e}$ , para as quais f(L) tem o valor "verdadeiro". Em outras palavras, esse feixe reunir $\tilde{a}$  as linhas que unem pontos, nos quais a propriedade "conceito comercial" tem o mesmo valor.

Esse feixe incluirã, por exemplo, a linha  $L_2$  e a linha  $L_4$ , entre outras, mas as linhas  $L_1$  e  $L_3$  ficarão fora dele.

Na figura anterior, estão figuradas todas as linhas abrangidas por esse feixe, que é formalmente definido pela expressão:

$$B_1 = (q_{1:5} = q_{2:5}, A_2, A_1)$$

na qual a função que determina o feixe  $\tilde{e}$  a seleção  $q_{1:5}=q_{2:5}$  e o conjunto de  $\tilde{a}$  reas no qual o feixe  $\tilde{e}$  formado  $\tilde{e}$   $\{A_2,A_1\}$ .

Poderiamos, agora, estar interessados em formar uma área A<sub>3</sub> na qual tivéssemos, como propriedades relevantes, apenas q<sub>3</sub> eq<sub>6</sub>, isto é, Número do CGC e número de firmas de conceito comercial i-gual ao da considerada.

Para obtê-la precisamos colher informações nas áreas  $A_2$  (onde consta o número de firmas classificado em cada classe de conceito comercial) e  $A_1$  (onde consta o número do CGC de cada firma).

Podemos consegui-la, através da função do feixe  $B_1$  definida de modo a associar a cada linha de  $B_1$  um ponto em que os valores das propriedades  $q_1$ ,  $q_2$ ,  $q_4$  e  $q_5$  são irrelevantes, o valor de  $q_3$  deve ser tomado do primeiro ponto da linha e o valor de  $q_6$  do segundo. Chamando essa função de  $F_1$ , podemos escrever:

$$F_1 \equiv (q_1'=\Omega, q_2'=\Omega, q_3'=q_{1:3}, q_4'=\Omega, q_5'=\Omega, q_6'=q_{2:6})$$

Nessa expressão, mencionamos todas as FOLs usadas para de finir as coordenadas dos pontos desejados, pois, por convenção, se não o fizessemos, entender-se-ia que o valor das coordenadas omitidas, deveria ser copiado do último ponto da linha.

Assim, podemos escrever a expressão de  $A_3$  em função de  $A_2$  e  $A_1$ :

$$A_3 = F_1(q_{1:5} = q_{2:5}, A_2, A_1)$$

Essa sistemática nos permitiu criar um arquivo, a p

Como A<sub>2</sub> é uma área criada a partir de A<sub>1</sub>, podemos exp. in the mir A<sub>3</sub> diretamente em função de A<sub>1</sub>, usando a expressão:

 $A_3 = F_1(q_{1:5} = q_{2:5}, H(q_5, A_1), A_1)$ 

Então, a área A3 seria formada dos seguintes pontos:

Razão da	social firms	Ende- reço	Nº do CGC	Capital Registrado	Conceito comercial	Contador
	۹ <sub>1</sub>	q <sub>2</sub>	<b>4</b> 3	4	9 <sub>5</sub>	96
i,	Ω	Ω	517947134	Ω	Ω	3
	Ω	Ω	517947128	Ω	Ω	<b>2</b>
	Ω	Ω	517947135	Ω	Ω	3
	Ω	Ω	517947431	Ω .	Ω	3
	Ω	Ω	517942133	Ω	Ω	1
	Ω	Q	517946147	Ω	Ω	5
	Ω	Ω	517948112	Ω	Ω	5
n	Ω	Ω	517943114	Ω	Ω	a 2
•	Ω	Ω	517941132	Ω	Ω	<b>3</b>
;	Ω	Ω	517912413	Ω	Ω	3
	Ω	Ω	517944512	ß	Ω	. 3
	Ω	Ω	517944413	Ω	Ω	5
.,	Ω	Ω	517752218	Ω	Ω	5
	Ω	Ω	517572221	Ω	Ω	5

No próximo capítulo, apresentaremos exemplo bem mais elaborado, onde se pode verificar a aplicação desses conceitos, de ma neira bastante completa.

Deixamos para apresentar os demais recursos fornecidos pela Álgebra da CODASYL, durante a exposição da dedução das expressões, que exprimem as relações entre os dados e os resultados do problema lá abordado.

	·		

## CAPITULO II

#### EQUACIONAMENTO DO PROBLEMA DE MONTAGEM DE PROGRAMAS

## 1 - INTRODUÇÃO

Entendemos por montagem de programas o processamento referente à transformação da informação em linguagem ASSEMBLER para linguagem de maquina.

As instruções em linguagem ASSEMBLER são divididas, em duas categorias: Instruções de Maquina e Pseudo Instruções. As primeiras são executadas pelo computador enquanto que as últimas des tinam-se a informar e controlar o Montador para que execute determinadas funções. Na tabela da pg. 31 apresentamos a relação das pseudo instruções e correspondentes funções relativas a um subconjunto do ASSEMBLER, usado pela família 704/709/7090 de computadores IBM, conhecido como SAP, que utilizaremos como ilustração.

O processo de montagem consiste, em resumo, das seguintes fases:

- 1) É inicialmente construída uma área que consiste das instruções "LIB" do programa em ASSEMBLER cujo campo de localização simbólica contém símbolos mutuamente distintos. Essa área é en tão, utilizada em conjunto com o arquivo de subrotinas com a fina lidade de:
  - a) Marcar cada instrução "LIB" válida com o valor da quantidade de memoria requerida pela correspondente subrotina.
  - b)- Marcar como erro cada instrução "LIB" não válida, isto é, aquelas cujo código simbólico de localiza ção é o mesmo que o de outra, e aquelas que não têm em correspondência nenhuma subrotina, no arquivo de subrotinas.

As subrotinas desse arquivo estão armazenadas em linguagem simbólica.

2) As instruções do programa principal são resequenciadas, para permitir a inserção das instruções que constituem as subrotinas chamadas. Essas últimas, por sua vez, também são submetidas a uma resequenciação para poderem ser inseridas, no programa principal. A reunião de todas essas instruções constitui o programa completo em ASSEMBLER.

- 3) As instruções BCD dão lugar a um número variável de palavras, no programa objeto, de acordo com o número que consta do campo de endereço da instrução. O próximo passo é assim a re-sequenciação do programa simbólico para deixar lugar as palavras geradas pelas instruções BCD.
- 4) Expurgamos a seguir o programa simbólico das instruções REM, LIB e SYN tendo em vista não haver palavras de codificação a elas correspondentes, no programa objeto. É introduzida nova propriedade: Número Sequencial Auxiliar para registrar a re-sequencia ção. O número Sequencial Auxiliar é então novamente ajustado de modo a refletir a expansão, causada pelas instruções BSS. Finalmente com o auxílio de instruções ORG, ao Número Sequencial Auxiliar é associada a Localização em binário; as instruções ORG são assim expurgadas do programa.
- 5) O programa em sua última configuração e o conjunto de instruções SYN expurgadas são operados de modo a produzir uma tabe la de símbolos. Todas as instruções cujos campos de localização con têm símbolos repetidos são marcadas como erros. Também são assinaladas como sendo erros as instruções SYN cujo campo de localização contém um símbolo não definido.
- 6) Todas as palavras do programa objeto, geradas por instruções BCD, DEC e OCT são agora montadas.
- 7) O conjunto de instruções de maquina do programa e então particionado em 2 sub-conjuntos: O das instruções com endereço simbólico e o das com endereço não simbólico.

As duas classes de instruções de máquina são montadas, se paradamente. Toda instrução cujo campo de localização contém um símbolo não definido ou cujo código simbólico de Operação é não vá lido, é assinalada como sendo erro. Os dois arquivos de saída, isto é, a listagem do programa e o programa objeto podem ser formados tomando-se simplesmente a união das áreas apropriadas desenvolvidas durante o processo descrito.

Vamos, a seguir, ver em detalhe cada uma dessas fases, e, com o auxílio da Algebra da Informação, escrever as expressões cor respondentes. Vamos utilizar para isso as propriedades e áreas que constam da tabela da pg.32.

PSEUDO	INS- FUNÇÃO TRU-		TIPO DE DADOS NO CAMPO DE		
TRU- ÇÃO			endereço simbólico	INDÍCADOR DE LOCAL <u>I</u> ZAÇÃO	
BCD	Provoca a montagem de uma a no- ve palavras de informação em co dificação BCD. O dígito de maior ordem do campo de endereço sim- bólico especifica o número de palavras reservadas	branco, símbolo	Contador de BCD: Intei- ro decimal Dados em BCD: Caracte- res alfa-numéricos	+ N em que n=1,2,,9	
355	Reserva o número de palavras es pecificado no campo de endereço simbólico	branco, símbolo	Inteiro decimal	+ #	
DEC	Provoca a montagem de uma pala- vra de informação decimal	branco, símbolo	Número decimal	+ 1 .	
LIB	Provoca a incorporação de uma subrotina do arquivo de subrotinas. O símbolo no campo de loca lização símbólica identifica a subrotina.	simbolo	branco	<b>→ N</b>	
OCT	Provoca a montagem de uma pala- vra de informação octal	branco, símbolo	Número octal	+ 1	
ORG	Ajusta o Indicador de Localiza- ção para o valor especificado no campo de endereço simbólico.	branço	Número decimal	E sjustado para N	
REM	Introduz comentários no progra	branco	Caracteres alfa-numéri- cos (No campo estendido de endereço)	Henhum	
STN	Associa ao símbolo no cempo de localização simbólica o endere- ço indicado no campo de endere- ço simbólico	símbolo	Símbolo <u>+</u> inteiro deci- mal	Kenhum	

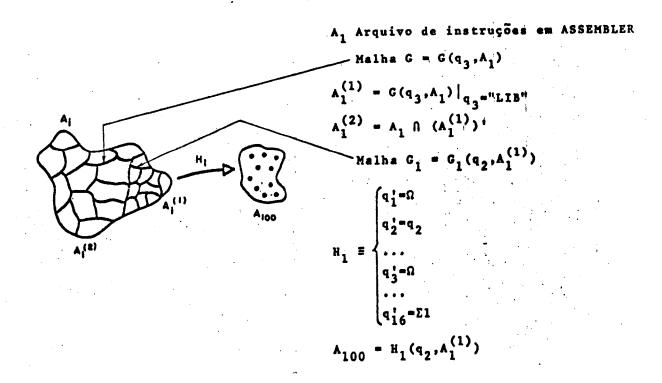
#### EXEMPLOS DE PSEUDO INSTRUÇÕES

#### CAMPO DE

		and the control of th
LOCALIZAÇÃO	operação	endereço
:	BCD	9 GUARDA ESTES CARACTERES NA MEMORIA
PM4	BCD ·	7 GUARDA ESTES NA POSIÇÃO PM4
• 9	BSS	26 RESERVA 26 PALAVRAS DE MEMORIA
P142	BSS	48 RESERVA 48 PALAYRAS E CHAMA A PRIMEIRA DE P142
, ,	DEC	137 GUARDA O NÚMERO DECIMAL 137 EM UMA POSIÇÃO DE MEMORIA
XR3	DEC	42 GUARDA O NÚMERO DECIMAL 42 NA POSIÇÃO DE MEMORIA XR3
SRQ	LIB	INCORPORA AO PROGRAMA A SUBROTINA SRO
•	OCT	142 GUARDA O NÚMERO OCTAL 142 EM UMA POSICÃO DE MEMÓRIA
AQ5	OCT	422 GUARDA O NÚMERO OCTAL 422 NA POSIÇÃO DE MEMORIA AQS
9	ORG	135 AJUSTA O INDICADOR DE LOCALIZAÇÃO PARA 135
	REM	ESTA INSTRUÇÃO SERVE PARA FAZER COMENTÁRIOS
XPT4	SYN	AQ41+5 CHAMA A POSIÇÃO DE MEMORIA AQ41+5 DE XPT4
VEL	SYN	ACT-8 CHAMA A POSIÇÃO DE MEMORIA ACT-8 DE VEL

	PROPRIEDADES E ÁREAS PARA	ESPECIFICAÇÃO DE ASS	SEMBL	ER			
	PROPRIEDADE	CONJUNTO DE VALORES DE PROP.	A <sub>1</sub>	A 2	A3	A4	٨
٩ <sub>1</sub>	Número sequencial	1,2,	×	×	Ω	Ω	×
q <sub>2</sub>	Código simb.de localização	branco, símbolo	×	×	ÿ	Ω	×
۹3	Codigo simb.de operação	3 caracteres	×	*	×	Ω	×
q_	Indicador simb.de endereço indireto	branco, *	×	×	υ	Ω	×
45	Endereço simbólico base	branco, símbolo, número decimal, número octal, *	×	×	Ω	Ω	×
۹6	Modificador de endereço simbólico	branco, + inteiro decimal	×	×	Ω	Ω	×
٠ ٩7	TAG simbólica	branco, 0,1,,7	×	×	Ω	Ω	×
۹ ه	Decremento simbólico	branco, inteiro decimal	×	×	Ω	Ω	×
۹9	Comentários	47 caracteres alfa-numéricos	×	×	Ω	.α 	*
q <sub>10</sub>	Contador simbólico BCD	1,2,,9	×	×	ถ	Ω	×
۹ <sub>11</sub>	Dados simbólicos em BCD	54 caracteres alfa-numéricos	×	×	Ω	Ω	×
q <sub>12</sub>	Subrotina E/S	Símbolo	Ω	×	Ω	Ω	Ω
q <sub>13</sub>	Localização em binário	15 bits	Ω	U	Ω	×	×
q <sub>14</sub>	Palavra em binário	36 bits	Ω	Ω	×	×	×
q <sub>15</sub>	Codigo de erro	E	Ω	Ω	Ω	Ω	×
9 <sub>16</sub>	Contador 1	1,2,	Ω	Ω	Ω	Ω	ιΩ
q <sub>17</sub>	Contador 2	1,2,	Ω	Ω	Ω	u	Ω
<sup>q</sup> 18	Número sequencial auxiliar	1,2,	Ω	ıΩ	Ω	Ω	U
A <sub>1</sub>	Arquivo de entrada em representação s	imbőlica					
A <sub>2</sub>	Arquivo de subrotinas em representaçã	o simbólica		•			•
A 3	Códigos simbólicos de operação e seus	equivalentes em bi	nário				
A	Programa objeto						

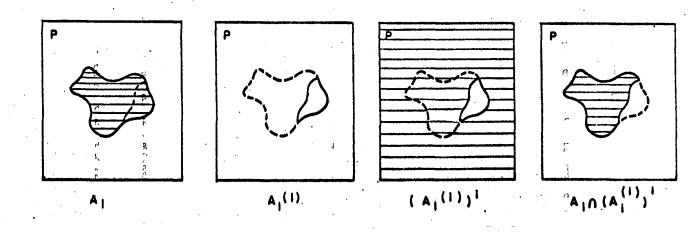
#### 2 - EXPRESSÕES CORRESPONDENTES À FASE 1



A partir do conjunto de instruções em ASSEMBLER, formamos uma Área que chamamos  $A_1$ . Adotando  $q_3$  como função que determina a malha, estabelecemos a malha G em  $A_1$  em cujos elementos estão agrupadas as instruções de mesmo código simbólico de Operação.

Chamamos de  $A_1^{(1)}$  o elemento no qual foram agrupadas as instruções de chamada de subrotinas, isto é, aquelas cujo código de operação é "LIB". O conjunto de todos os demais elemento chamamos de  $A_1^{(2)}$ .

Na figura abaixo, temos informações para esclarecer a expressão de  $\mathbb{A}_1^{(2)}$ .



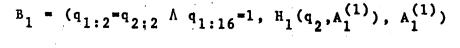
Adotando  $q_2$  como função que determina a malha, estabelece mos, agora, a malha  $G_1$  em  $A_1^{(1)}$  em cujos elementos estão agrupadas as instruções de mesmo código simbólico de localização.

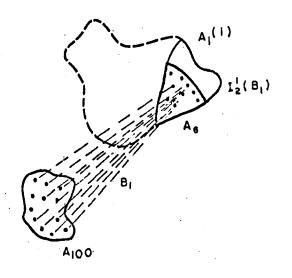
Pela função H<sub>1</sub> da malha G<sub>1</sub> estabelecemos a área A<sub>100</sub> na qual cada ponto tem como propriedades definidas a contagem dos pontos de um dos elementos da malha e seu código simbólico de localização. As demais propriedades são não definidas. A<sub>100</sub> nos informa, assim, qual o número de vezes que cada subrotina é chamada.

Nesse ASSEMBLER, so tem sentido chamar-se uma vez cada subrotina, pois usos repetidos da mesma subrotina se fazem através de desvios, evidentemente precedidos pelo carregamento de um inde-xador, com a informação necessária a uma volta adequada.

Com o conjunto de áreas  $\{A_{100},A_1^{(1)}\}$  podemos formar um feixe  $B_1$  utilizando como função que determina o feixe a seleção  $q_{1:2}$  =  $q_{2:2}$   $\Lambda$   $q_{1:16}$ =1. Dessa forma, teremos cada linha do feixe ligando pontos que correspondem a uma subrotina chamada uma única vez e a instrução de chamada correspondente. A interseção desse feixe com  $A_1^{(1)}$  é assim o subconjunto de  $A_1^{(1)}$  que inclui as chamadas de subrotinas que foram chamadas uma só vez. Chamamos essa área de  $A_6$ . Ela pode ser definida de dois modos equivalentes ou como a interseção mencionada ou como obtida pela função  $F \equiv ($ ) do feixe  $B_1$ .

Essa segunda maneira utiliza a função F que associa a cada linha do feixe um ponto cujas coordenadas são as mesmas que as do ponto da linha situado na última área do conjunto.



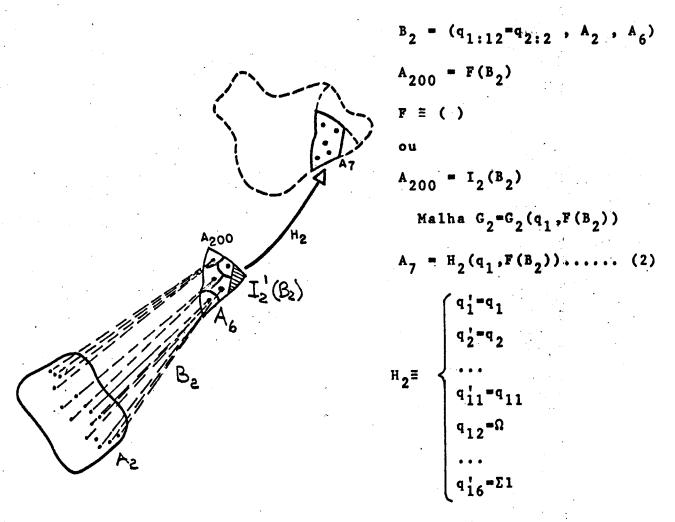


$$A_6 = F(B_1)$$
 .....(1)

ou

$$A_6 = I_2(B_1)$$

Note-se que o complemento da interseção do feixe  $B_1$  com  $A_1^{(1)}$ , isto  $\tilde{e}$ ,  $I_2^{'}(B_1)$   $\tilde{e}$  formado pelo conjunto das instruções "LIB" referentes a subrotinas chamadas mais de uma vez.



Considerando-se o arquivo de subrotinas em representação simbólica A2 formando um conjunto de áreas com A6, podemos, através da seleção q1:12=q2:2, determinar o feixe B2, em que cada maço de linhas liga todas as instruções de uma subrotina à sua instrução de chamada. A interseção de B2 com A6 nos dá uma área constituída das instruções de chamada de subrotinas que existem em A2. Podemos estabelecer, nessa área, uma malha G2, usando q1 como função que determina a malha; isto é, em cada elemento da malha, tere mos apenas um ponto, que é a instrução de chamada de subrotina, instrução essa que tem aquele número sequencial. Esse ponto, todavia, pertence a várias linhas do feixe, pois é por elas ligadas a todas as instruções da subrotina chamada.

Com a função  $H_2$  dessa malha, podemos criar a área  $A_7$ , que é constituída por todas as instruções de chamada das subrotinas, que são chamadas uma só vez, e que existem efetivamente em  $A_2$ . Nessas

instruções o contador l é utilizado para contar o número de instruções da subrotina chamada.

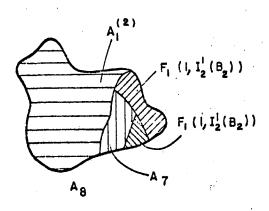
Note-se que cada ponto de  $I_2^{\dagger}(B_2)$  isto  $\tilde{\epsilon}$ , do complemento em relação a  $A_6$  da interseção de  $B_2$  com  $A_6$ , representa uma instrução de chamada de subrotina cuja correspondente subrotina não existe em  $A_2$ .

Consideremos, agora, a FOB  $F_1$  do feixe definido, no conjunto de áreas constituído pela unica área  $I_2^{'}(B_2)$ , em que a seleção  $\tilde{e}$  l, isto  $\tilde{e}$ , inclui todas as linhas, ou seja todos os pontos da área.

$$F_{1} = \begin{cases} q_{1}^{\dagger} = q_{2} : 1 \\ q_{2}^{\dagger} = q_{2} : 2 \\ q_{3}^{\dagger} = q_{2} : 3 \\ \vdots \\ q_{11}^{\dagger} = q_{2} : 11 \\ q_{11}^{\dagger} = q_{2} : 11 \\ q_{11}^{\dagger} = q_{2} : 11 \\ q_{12}^{\dagger} = q_{2} : 3 \\ \vdots \\ q_{15}^{\dagger} = q_{2}^{\dagger} = q_$$

 $F_1$  apenas assinala como "erro" tanto as instruções referentes a chamada de subrotina que são chamadas mais de uma vez, como aquelas que chamam subrotinas que não existem em  $A_2$ .

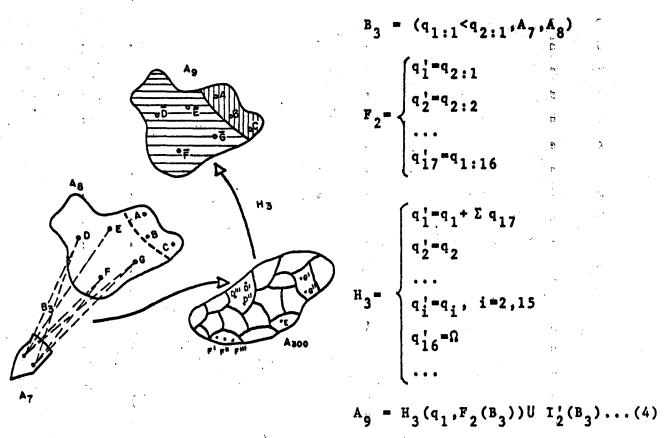
Podemos, agora, formar a área A<sub>8</sub> que engloba as mesmas instruções constantes de A<sub>1</sub>, isto é, do programa em representação simbólica, com a diferença que, as instruções "LIB" válidas foram marcadas com os requisitos de memória da subrotina correspondente, e as não válidas foram assinaladas como erros.



$$A_8 = A_1^{(2)} U A_7 U F_1(1, I_2^{'}(B_2)) U F_1(1, I_2^{'}(B_1)) \dots (3)$$

#### 3 - EXPRESSÕES CORRESPONDENTES À FASE 2

Tomemos o conjunto de áreas {A<sub>7</sub>, A<sub>8</sub>} e com a seleção q<sub>1:1</sub> < q<sub>2:1</sub> formemos o feixe B<sub>3</sub>, através do qual relacionamos cada instrução do programa em ASSEMBLER com todas as instruções "LIB"vã lidas, cujo número sequencial é inferior ao da instrução considera da. No esquema a seguir, vemos alguns exemplos elucidativos. Consideremos os pontos A, B, C, D, E, F, G de A<sub>8</sub>. Observamos que os pontos ABC não estão ligados a nenhum ponto de A<sub>7</sub>, o que significa ca que as instruções que são a eles correspondentes precedem qualquer instrução "LIB". Jã o ponto E está ligado a um dos pontos de A<sub>7</sub>, enquanto que o ponto D liga-se a todos. Isso significa que a instrução correspondente a E tem número sequencial maior que o de uma das instruções "LIB", ao passo que a correspondente a D é precedida pelas três instruções "LIB".

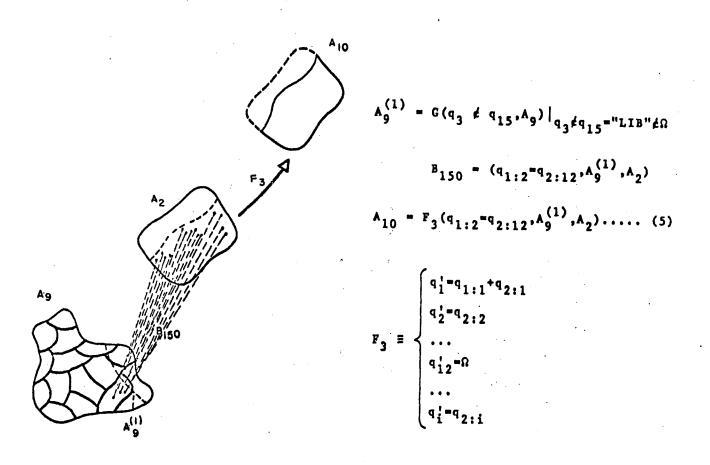


A função  $F_2$  leva cada linha do feixe a um ponto, na Area Produto  $A_{300}$ . Esse ponto tem todas as propriedades com valor copia do do ponto da linha pertencente a  $A_8$ , com exceção do contador 2 que  $\tilde{e}$  ajustado de acordo com o contador 1 do ponto da linha pertencente a  $A_7$ . Isto quer dizer que cada instrução do programa em

ASSEMBLER fica associada a tantos pontos em  $A_{300}$  quantas são as instruções "LIB" válidas que a precedem. Estabelecendo, nessa área, a malha  $G_3 = G_3(q_1,A_{300})$  teremos agrupados em cada elemento da malha os pontos mencionados. Aplicando a FOG  $H_3$  a  $G_3$  obteremos uma área na qual cada ponto corresponde a uma instrução do programa em ASSEMBLER, tendo o número sequencial acrescido da soma do número de instruções de todas as subrotinas chamadas por instruções "LIB" vá lidas, que a precedem.

Existem, todavia, pontos em  $A_8$  que não pertencem a nenhuma linha do feixe  $B_3$ , isto é, que representam instruções do programa em ASSEMBLER que precedem qualquer instrução "LIB" válida. A, B e C em nosso diagrama ilustram esse caso. Esses pontos pertencem ao complemento, em relação a  $A_8$ , da interseção do feixe  $B_3$  com a área  $A_8$ , isto é, a  $I_2'(B_3)$ .

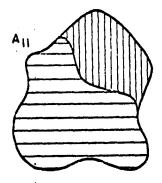
A união das áreas criadas de acordo com a descrição constante dos dois parágrafos anteriores a este nos dá a área A<sub>9</sub> que engloba as mesmas instruções constantes de A<sub>1</sub>, isto é, do programa em representação simbólica, com a diferença que o número sequencial de cada instrução foi ajustado de modo a permitir a inserção das subrotinas validamente chamadas.



Vamos, agora, estabelecer uma malha em  $A_9$  utilizando como função que determina a malha a concatenação de  $q_3$  com  $q_{15}$ . Cada elemento dessa malha agrupara as instruções de mesmo código simbólico de operação. As instruções "LIB", entretanto, estarão separadas em dois elementos: um das instruções marcadas com erro, isto é, em que  $q_{15}$ ""E", e outro das instruções em que  $q_{15}$ 0, isto é, das instruções válidas. Chamamos a esse segundo de  $A_9$ 0.

Utilizando o conjunto de áreas cemos o feixe que liga as chamadas de subrotinas, as instruções em ASSEMBLER que constituem essas subrotinas. Para isso, utilizamos uma sistemática análoga à que foi usada na fase 1, para estabelecer o feixe B2. Então, pela FOB F3 desse feixe, estabelecemos a area A<sub>10</sub> que é formada das instruções das subrotinas chamadas,com o número sequencial ajustado para permitir sua inserção no progra ma principal. F, ainda torna irrelevante o valor da propriedade q<sub>12</sub> que, antes disso, assinalava a instrução como pertencente alguma subrotina. Os valores das demais propriedades das instruções são mantidos inalterados, tal como figuravam em A2. É interessante notar que o ajuste do número sequencial é feito, somandose ao número sequencial de cada instrução o da instrução de chama da correspondente. Convem lembrar que as instruções do programa principal ja tiveram seus números sequenciais ajustados, de modo a permitir essa inserção.

A união das áreas A<sub>9</sub> e A<sub>10</sub> nos dá a área A<sub>11</sub> que constitui o programa completo em ASSEMBLER.



$$A_{11} = A_9 U A_{10} \dots (6)$$

Ocorre, agora, uma pergunta: E se alguma subrotina chamada incluir uma ou mais instruções "LIB", como prossegue o processa mento da montagem do programa ?

Conhecendo o processo de montagem, os projetistas do "software" do computador ja terão evitado incluir esse tipo de instrução, ao prepararem as subrotinas. Daí se depreende que o problema levantado não existe, na realidade.

## 4 - EXPRESSÕES CORRESPONDENTES À FASE 3

Vamos, agora, resequenciar as instruções do programa completo, em linguagem simbólica, para acomodar a expansão das instruções "BCD". Pela tabela da pg.31, vemos que essa instrução provoca a montagem de tantas palavras quanto as indicadas pelo dígito de maior ordem do campo de endereço simbólico.

Para isso, estabelecemos uma malha na área  $A_{11}$  utilizando  $q_3$  como função que a determine. Assim, em cada elemento estarão a grupadas todas as instruções que têm o mesmo código simbólico de operação.

Tomemos o elemento que reune as instruções "BCD" e formemos com ele e toda a área  $A_{11}$ , um conjunto de áreas no qual determinamos um feixe  $B_4$  utilizando a seleção  $q_{1:1} < q_{2:1}$ . Esse feixe vai, assim, interligar cada instrução do programa completo com todas as instruções "BCD", que a precedem. Pela FOB  $F_4$  obtemos uma área em que cada linha do feixe tem como imagem a instrução de  $A_{11}$  com seu contador l'ajustado, de acordo com o contador BCD da instrução desse tipo, que a precede. A cada instrução do programa completo corresponderão, assim, vários pontos nessa área, tantos quantas forem as instruções "BCD" de número sequencial menor que o da instrução considerada. Podemos agrupar esses pontos no mesmo elemento de uma malha, utilizando  $q_1$  como função que a determine. Depois, com a função  $H_4$  dessa malha estabelecemos a resequenciação desejada:

$$A_{12} = H_4(q_1, F_4(B_4)) U I_2(B_4) \dots (7)$$

em que

$$B_4 = (q_{1:1} < q_{2:1}, G(q_3, A_{11}) |_{q_3 = BCD}, A_{11})$$

$$F_4 \equiv (q_{16}^! = q_{1:10}^!)$$
 $H_4 \equiv (q_1^! = q_1 + \Sigma(q_{16}^{-1}), q_1^! = q_1, i = 2,3,...,11,15)$ 

É bom lembrar que I'(B4) leva para a área A12 as instruções de A11 que não eram precedidas de nenhuma instrução BCD. A12 constitui, pois, uma área que engloba todas as instruções do programa completo, já resequenciadas para permitir a expansão das instruções "BCD".

#### 5 - EXPRESSÕES CORRESPONDENTES À FASE 4

Pode-se então retirar do programa completo as instruções "LIB" pois ja cumpriram sua finalidade. Para isso, estabelecemos a malha G na area A<sub>12</sub>, determinada pela função q<sub>3</sub>. Todas as instruções que possuem o mesmo Código Simbólico de Operação ficam assim, agrupadas em um mesmo elemento dessa malha. Chamamos  $A_{12}^{(3)}$ elemento que reune as instruções "LIB". Vamos aproveitar essa malha, para retirar outras instruções do programa em ASSEMBLER não produzem instruções correspondentes em linguagem de maquina, e que a essa altura ja podem ser retiradas. As instruções "REM" e "SYN" estão nesse caso, pois, so vão interessar à listagem do programa fonte. Chamamos  $A_{12}^{(2)}$  o elemento da malha que reune as instruções "REM" e A<sub>12</sub> o que reune as instruções "SYN". O complemen to da União desses tres elementos é o conjunto de todos os pontos do Espaço de Propriedades que não pertencem a eles. A interseção desse complemento com a área A<sub>12</sub> dá o conjunto das instruções programa completo, que não são "REM", "LIB" ou "SYN". Chamamos de A<sub>12</sub> esse conjunto. Utilizando como conjunto de Áreas

$$\{ \bigcup_{i=1}^{3} A_{12}^{(i)}, A_{12}^{(4)} \}$$

estabelecemos o feixe  $B_5$ , que, pela seleção  $q_{1:1} < q_{2:1}$  reune as linhas que ligam os pontos de  $A_{12}^{(4)}$  aos pontos de

$$\begin{array}{c}
3\\U\\i=1
\end{array}$$

de número sequencial menor. Em outras palavras, cada linha desse feixe relaciona uma instrução do programa, expurgado com uma das

instruções "LIB", "REM" ou "SYN" que a precedem. Através da função  $F_5$  associamos, a cada linha desse feixe, um ponto que, além dos valores das propriedades do ponto da linha pertencente a  $A_{12}^{(4)}$ , tem o número sequencial auxiliar igual ao número sequancial do ponto da linha pertencente à primeira área do Conjunto de Áreas. Ou seja, dessa forma cada instrução do programa expurgado tem em correspondência tantos pontos quantas forem as instruções "LIB", "REM" ou "SYN" que a precedem. Pela malha determinada na função  $q_1$  grupamos todos os pontos correspondentes a uma instrução num só elemento e pela FOG  $H_5$  ajustamos o número sequencial auxiliar, para a retirada das instruções "LIB", "REM" e "SYN". Em resumo temos:

$$A_{13} = H_{5}(q_{1}, F_{5}(B_{5})) \cup F_{5}(1, I_{2}'(B_{5})) \dots (8)$$
em que
$$B_{5} = (q_{1:1} < q_{2:1}, \bigcup_{i=1}^{3} A_{12}^{(i)}, A_{12}^{(4)})$$
onde
$$A_{12}^{(1)} = G(q_{3}, A_{12}) |_{q_{3}} = "SYN"$$

$$A_{12}^{(2)} = G(q_{3}, A_{12}) |_{q_{3}} = "REM"$$

$$A_{12}^{(3)} = G(q_{3}, A_{12}) |_{q_{3}} = "LIB"$$

$$A_{12}^{(4)} = A_{12} \cap (\bigcup_{i=1}^{3} A_{12}^{(i)}) '$$

$$H_{5} \equiv (q_{1}' = q_{1}, i = 1, 2, \dots, 11, q_{18}' = q_{1} - \Sigma1)$$

$$F_{5} \equiv (q_{18}' = q_{1:1})$$

Na expressão (8) temos a Área  $F_5(1, I_2'(B_5))$  que representa as instruções de  $A_{12}^{(4)}$  que não são precedidas de nenhuma instrução de Código Simbólico de Operação igual ao das que foram retiradas. São formalmente definidas como sendo o conjunto complementar em relação a  $A_{12}^{(4)}$  da interseção do feixe  $B_5$  com essa área. Formamos, então, novo feixe com a seleção 1, isto é, que inclui todos esses pontos e com a função  $F_5$  ajustamos o número sequencial auxiliar, copiando simplesmente o número sequencial que já possuiam.

A area A13 representa, assim, o programa completo em

ASSEMBLER expurgado das instruções "LIB", "REM" e "SYN".

Vamos, agora, ajustar o número sequencial auxiliar das instruções para permitir a expansão causada pelas instruções BSS. Para isso, estabelecemos a malha G em  $A_{13}$  determinada pela função  $q_3$ . Tomando o conjunto de áreas formado pelo elemento que reune as instruções "BSS" e toda a área  $A_{13}$  construimos o feixe  $B_6$ , determinado pela seleção  $q_{1:1} < q_{2:1}$ , isto é,

$$B_6 = (q_{1:1} < q_{2:1}, G(q_3, A_{13})|_{q_3} = "BSS", A_{13})$$

Esse feixe liga cada instrução de A<sub>13</sub> a todas as instruções "BSS", que a precedem. Através da FOB F<sub>6</sub>, cada instrução é le vada em tantos pontos quantas são as instruções "BSS" que a precedem, tendo cada um desses pontos o contador l ajustado para o núme ro de palavras a ser reservado, de acordo com a instrução "BSS" cor respondente.

$$F_6 = (q_{16}^1 = q_{1:5}^1)$$

Através da malha determinada pela função  $q_1$  agrupamos esses pontos num mesmo elemento e, pela FOG  $H_6$  ajustamos o número se quencial auxiliar, conforme desejado.

$$H_6 \equiv (q_i^! = q_i^!, i = 1, 2, ..., 11, q_{18}^! = q_{18}^! + \Sigma(q_{16}^{-1})$$

As instruções que não tinham a precedê-las nenhuma instrução "BSS" estão contidas no complemento, em relação a  $A_{13}$  da interseção do feixe  $B_6$  com essa área.

Podemos, então, escrever a equação que define o programa em ASSEMBLER, agora com o número sequencial auxiliar ajustado, para refletir a expansão causada pelas instruções "BSS".

$$A_{14} = H_6(q_1, F_6(B_6)) U I'_2(B_6)$$
 ....(9)

Cuidemos agora das instruções "ORG". Essas instruções são utilizadas para ajustar o indicador de localização (Instruction Counter) para certo valor, isto é, para dar um salto correspondente à diferença entre o número indicado pelo seu campo de endereço simbólico e o número indicado, no seu campo de localização simbólica, na alocação de endereço para as instruções que se seguem.

Utilizando a malha G podemos, pelas expressões abaixo, se parar as instruções do programa em ASSEMBLER em duas areas:  $A_{14}^{(1)}$  onde ficam agrupadas as instruções "ORG" e  $A_{14}^{(2)}$  onde se reunem as demais.

$$A_{14}^{(1)} = G(q_3, A_{14})|_{q_3 = \text{"ORG"}}$$
 $A_{14}^{(2)} = A_{14} \cap (A_{14}^{(1)})$ 

Usando essas duas áreas para formar um Conjunto de Áreas, podemos através do feixe B<sub>7</sub> ligar cada instrução não "ORG" a todas as instruções "ORG" que a precedem, isto é,

$$B_7 = (q_{1:1} < q_{2:1}, A_{14}^{(1)}, A_{14}^{(2)})$$

Pela função F<sub>7</sub> fazemos corresponder a cada instrução não "ORG" tantos pontos quantas são as instruções "ORG" que a precedem. Cada um desses pontos tem o contador l ajustado, de acordo com o endereço simbólico, base da instrução "ORG", o contador 2 ajustado para o número sequencial auxiliar da instrução "ORG" e o valor das demais propriedades, igual ao da instrução não "ORG".

$$F_7 \equiv (q_{16}^1 - q_{1:5}, q_{17}^1 - q_{1:18})$$

Através da malha determinada por q<sub>1</sub>, agrupamos esses pontos todos num mesmo elemento e pela FOG H<sub>7</sub> calculamos a localização em binário da instrução.

$$H_7 \equiv (q_i' = q_i, i = 1, 2, ..., 11, q_{13}' = q_{16} (max{q_{17}}) + q_{18} - max{q_{17}} - 1)$$

Vejamos um exemplo para ilustrar melhor. Seja  $\ell_1$  uma instrução não "ORG" e  $\ell_2$ ,  $\ell_3$ ,  $\ell_4$  instruções "ORG" que a precedem.

<sup>q</sup> 1	<sup>q</sup> 2	<sup>q</sup> 3	<sup>q</sup> 5	<sup>q</sup> 18
17	*	ORG	83	73
29		ORG	145	85
38		ORG	272	94
: 67	TOT4	LDA	ATA	122

Pela FOB F, li e levada nos 3 pontos l<sub>100</sub>, l<sub>200</sub>, l<sub>300</sub>

			_	_	· ,q <sub>16</sub>		
£100	67	TOT4	LDA	ATA	83	73	122
<sup>2</sup> 200	67	TOT4	LDA	ATA	145	85	122
£300	67	TOT4	LDA	ATA	272	94	122

Quem tem max  $\{q_{17}\}$  é o ponto  $l_{300}$ . Então, pela FOG  $H_7$  a  $l_1$  corresponderá o ponto

$$q_1$$
  $q_2$   $q_3 \cdots q_5 \cdots q_{13} \cdots q_{16}$   $q_{17}$   $q_{18}$ 
67 TOT4 LDA ATA 299  $\Omega$   $\Omega$ 

em que q'a foi determinado, assim:

$$\max\{q_{17}\}=94\ q_{16}(\max\{q_{17}\})=272\ q_{13}^*=272+122-94-1=299$$

As instruções que não tinham nenhuma instrução "ORG" a precedê-las situam-se evidentemente, no complemento em relação a  $A_{14}^{(2)}$  da interseção do feixe  $B_7$  com essa área. Tomando o conjunto de áreas formado por essa área única e a seleção 1, formamos um feixe ao qual aplicamos a função  $F_8$ , assim definida:

$$F_8 \equiv (q_{13}^{\dagger} - K - 1 + q_{1:18}, q_{18}^{\dagger} - \Omega)$$

Essa função atribui a cada instrução uma localização em binário, calculada com origem arbitraria K, levando em conta o número sequencial auxíliar da instrução.

Podemos, então, escrever a equação pela qual determinamos a área A<sub>15</sub> que reune as instruções do programa em ASSEMBLER, já com as localizações em binário, calculadas para cada instrução que irá constituir o programa objeto.

$$A_{15} = H_7(q_1, F_7(B_7)) \cup F_8(1, I_2(B_7)) \dots (10)$$

Entremos, agora, em mais detalhes sobre a ação da ORG, nas instruções do programa em ASSEMBLER.

Consideremos ainda a instrução  $\ell_1$ . Se ela não fosse precedida de nenhuma instrução "ORG" sua localização em binário seria tomando K=0.

$$q_{13}^{\dagger} = 122-1 = 121$$

Se fosse precedida pela instrução "ORG" l<sub>2</sub> apenas, sua l<u>o</u> calização em binário seria

$$q_{13}^{\dagger} = 83-73+122-1 = 10+121 = 131$$

Se fosse precedida por  $\ell_1$  e  $\ell_2$ , teríamos

$$q_{13}^{\prime} = 145-85+122-1 = 60+121 = 181$$

Sendo precedida por  $\ell_1$ ,  $\ell_2$  e  $\ell_3$  sua localização binária tornou-se

$$q_{13}^{*} = 272-94+122-1 = 178+121 = 299$$

Note-se que a área A<sub>15</sub> já está afetada do efeito das instruções "ORG" e ao mesmo tempo já foi expurgada dessas instruções, pois a elas não vão corresponder instruções de máquina.

Todavia, a area  $A_{14}^{(1)}$ , que reune todas as instruções desse tipo irá integrar a listagem do programa fonte. Convém lembrar, neste ponto, que também a area  $A_{12}^{(4)}$ , que reune instruções "REM", "LIB" e "SYN", tal como descrito no início da dedução das equações correspondentes a esta fase da montagem, irá integrar a listagem do programa fonte.

### 6 - EXPRESSÕES CORRESPONDENTES À FASE 5

Inicialmente, consideremos uma area que consiste da união de  $A_{15}$  e  $A_{12}^{(1)}$ , isto é, de todas as instruções em linguagem simbólica, exceto as que foram expurgadas e mais das instruções "SYN" que haviam sido anteriormente expurgadas. Pela função  $\mathbf{q}_2$  determinemos uma malha nessa area. Assim, cada elemento dessa malha agrupara instruções com mesmo código simbólico de localização. Pela FOG  $\mathbf{H}_1 \equiv (\mathbf{q}_2^{'} = \mathbf{q}_2^{'}, \mathbf{q}_1^{'} \mathbf{f}_0 = \Sigma \mathbf{1})$  associamos a cada elemento da malha, um ponto que tem como propriedades relevantes apenas o código simbólico de localização e o contador 1 ajustado para o número de pontos do elemento. A area formada pelos pontos assim obtidos e  $A_{15}$  U  $A_{12}^{(1)}$  formam um conjunto de areas onde podemos, pelo feixe  $\mathbf{B}_8$ , selecionar as linhas que ligam pontos de mesmo código simbólico de loca-

lização, em cuja relação não exista mais de uma instrução com esse código e que não seja o símbolo "b", isto  $\tilde{\epsilon}$ , branco. Chamamos de  $A_{16}$  a interseção desse feixe com  $A_{15}$  U  $A_{12}^{(1)}$ .

Note-se que em  $A_{16}$  estão reunidas as instruções cujos códigos simbólicos de localização são constituídos por símbolos mutuamente distintos. As demais pertencem ao complemento da interseção de  $B_8$  com  $A_{15}$  U  $A_{12}^{(1)}$ , isto  $\tilde{\epsilon}$ , pertencem a  $I_2^{\prime}(B_8)$ .

Estabelecemos, a seguir, em  $A_{16}$  a malha determinada pela função  $q_3$ . Em cada elemento dessa malha estão reunidas as instruções que têm o mesmo código simbólico de Operação. Chamamos de  $A_{16}^{(1)}$  o elemento que reune as instruções "SYN" e de  $A_{16}^{(2)}$  o conjunto formado pela união dos demais elementos.

$$A_{16}^{(1)} = G(q_3, A_{16})|_{q_3 = "SYN"}$$
 $A_{16}^{(2)} = A_{16} \cap (A_{16}^{(1)})'$ 

Formemos com essas duas áreas um conjunto de áreas  $\{A_{16}^{(1)},A_{16}^{(2)}\}$  no qual pela seleção  $q_{1:5}=q_{2:2}$  formamos o feixe  $B_9$ . Cada linha desse feixe liga uma instrução "SIN" a uma instrução não "SYN" cujo código simbólico de localização é igual ao símbolo que figura no campo de endereço simbólico da instrução "SYN".

Por exemplo as instruções:

q <sub>1</sub>	q <sub>2</sub>	<sup>9</sup> з	<sup>9</sup> 5	<sup>q</sup> 6	<sup>q</sup> 13
173	PEG	ADD	QUA		216
197	ANA	 SYN	PEG+3		253

ficariam ligadas por uma linha de B<sub>9</sub>.

Aplicando a esse feixe a FOB F<sub>9</sub> temos uma área em que cada ponto tem o código de localização simbólica da instrução "SIN" e a localização em binário, determinada pela soma da localização em binário da instrução não "SYN" com o modificador de endereço simb<u>ó</u> lico da instrução "SYN". As demais propriedades são copiadas da instrução não "SYN".

Conseguimos, assim, associar a cada instrução "SYN" uma instrução fictícia, na qual todavia temos em  $q_2$  um símbolo válido e em  $q_{13}$  a correspondente localização em binário, calculada com a definição dada pela instrução "SYN".

As instruções não "SYN", reunidas anteriormente em  $A_{16}^{(2)}$ jã tinham calculada a localização em binário correspondente à localização simbólica (fase 4).

A área  $A_{17}$  reune, assim, todas as informações necessárias a uma tabela de símbolos (possue até informações desnecessárias, que em nada atrapalham, todavia).

$$A_{17} = A_{16}^{(2)} \cup F_{9}(B_{9}) \dots (12)$$

$$B_{9} = (q_{1:5} = q_{2:2}, A_{16}^{(1)}, A_{16}^{(2)})$$

$$F_{9} = (q_{2}' = q_{1:2}, q_{13}' = q_{2:13} + q_{1:6})$$

Cada ponto do complemento da interseção do feixe  $B_9$  com  $A_{16}^{(1)}$ , isto é, de  $I_1'(B_9)$  representa uma instrução "SYN" cujo símbolo colocado no campo de endereço simbólico não tem em correspondên cia nenhum símbolo colocado no campo de localização simbólica de nenhuma instrução não "SYN". Pela seleção l formemos um feixe usando essa interseção como única área que constitui o conjunto de áreas. Então, o feixe engloba todos esses pontos e pela FOB  $F_1$  podemos as sinalá-los como erros, se lembrarmos que

$$F_1 \equiv (q_{15}^{\dagger} = "E")$$

Se lembrarmos ainda (pg.47) que em  $I_2'(B_8)$  estão as instruções cujos códigos simbólicos de localização são "b" (branco) ou não mutuamente distintos, podemos, através da FOB  $F_{10}$ , assinalar estas últimas como erro.

$$F_{10} \equiv (q_{15}^1 = \Omega + (q_{1:2}^2 = b^2) \rightarrow E^2)$$

Então podemos escrever a expressão que representa  $A_{15}$  U  $A_{12}^{(1)}$  com os códigos de erro já atribuídos.

$$A_{18} = A_{16} U F_1(1,I_1'(B_9))UF_{10}(1,I_2'(B_8)) \dots (13)$$

#### 7 - EXPRESSÕES CORRESPONDENTES À FASE 6

Usando  $q_3$  como função que determina a malha, vamos agrupar as instruções, de acordo com o código simbólico de Operação , nos elementos da malha G. Chamamos

$$A_{18}^{(1)} = G(q_3, A_{18})|_{q_3 = \text{"BCD"}}$$

$$A_{18}^{(2)} = G(q_3, A_{18})|_{q_3 = \text{"BSS"}}$$

$$A_{18}^{(3)} = G(q_3, A_{18})|_{q_3 = \text{"DEC"}}$$

$$A_{18}^{(4)} = G(q_3, A_{18})|_{q_3 = \text{"OCT"}}$$

$$A_{18}^{(5)} = G(q_3, A_{18})|_{q_3 = \text{"SYN"}}$$

$$A_{18}^{(6)} = A_{18} \cap (\bigcup_{i=1}^{5} A_{18}^{(i)})^{i}$$

Note-se que  $A_{18}^{(5)}$  é um dos componentes da listagem do programa fonte e que  $A_{18}^{(6)}$  reune as instruções de máquina do programa.

Tomemos, agora, um conjunto de áreas formado por uma única área: A<sup>(1)</sup>. Usando seleções diferentes, formaremos vários feixes, nesse conjunto de áreas.

(1,A<sub>18</sub><sup>(1)</sup>) engloba todos os pontos da area

 $(q_{1:10} > 1,A_{18}^{(1)})$  reune todos os pontos com contador de BCD maior que 1.

 $(q_{1:10} > 2, A_{18}^{(1)})$  reune todos os pontos com contador de BCD maior que 2.

. . .

 $(q_{1:10} > 8, A_{18}^{(1)})$  reune todos os pontos com contador de BCD igual a 9.

Vamos considerar a informação em codificação BCD como um vetor de tantas componentes quantas indique o contador de BCD. Desse modo, o conjunto de informações em codificação BCD irá formar um conjunto de vetores, com número variável de componentes. Pela FOB  $F_{11}$  processamos a primeira componente de todos esses vetores. Pela FOB  $F_{12}$  processamos a segunda componente dos vetores com mais de uma componente. Pela FOB  $F_{12}^{(2)}$  processamos a terceira componente dos vetores com mais de duas componentes, e assim por diante, até que pela FOB  $F_{12}^{(8)}$  processaremos a nona componente dos vetores com nove componentes.

A area A representa, pois, o conjunto de informações em codificação BCD ja montado:

$$A_{19} = F_{11}(1, A_{18}^{(1)}) \bigcup_{j=1}^{8} F_{12}^{(j)}(q_{1:10} > j, A_{18}^{(1)}) \dots (14)$$

em que

$$F_{11} \equiv (q_{14}^{\prime} - q_{1:11}^{(1)})$$

$$F_{12}^{(j)} \equiv (q_{1}^{\prime} - q_{1:1}^{\prime} + j, q_{1}^{\prime} - \Omega, i - 2, 3, ..., 11, q_{13}^{\prime} - q_{1:13}^{\prime} + j, q_{14}^{\prime} - q_{1:11}^{(j+1)})$$

Nessa notação q $^{(K)}_{1:11}$  representa a K-ésima componente do vetor de informação em codificação BCD.

Partindo do conjunto de áreas formado pela única área  $A_{18}^{(3)}$  U  $A_{18}^{(4)}$ , pela seleção 1 tomamos todas as linhas possíveis (formadas de um só ponto). Pela FOB  $F_{13}$  montamos toda a informação em decimal e toda a informação em octal.  $A_{20}$  é a área que representa esse tipo de informação:

$$A_{20} = F_{13}(1, A_{18}^{(3)} \cup A_{18}^{(4)}) \dots (15)$$

$$F_{13} \equiv (q_{14}^{\prime} = q_{1:5}^{\prime})$$

em que

#### 8 - EXPRESSÕES CORRESPONDENTES À FASE 7

Sendo  $V_2$  o Conjunto de valores da Propriedade  $q_2$ ,  $V_2^{-}\{"b"\}$  expressa o conjunto de todos os possíveis códigos simbólicos de  $1_{\underline{0}}$  calização.

Consideremos cada ponto de  $A_{18}^{(6)}$  como sendo uma linha formada de um só ponto. Consideremos a OFOL de seleção  $g_1 = (q_5 \in V_2 - \{"b"\})$ . Utilizando essa função podemos determinar em  $A_{18}^{(6)}$  a malha G e exprimir as instruções de máquina que têm endereço simbólico, como estando reunidas no elemento dessa malha definido por:

$$A_{18}^{(7)} = G(g_1, A_{18}^{(6)})|_{g_1 = (q_5 \in V_2 - \{"b"\})}$$

As instruções de maquina com endereço não simbólico ficam reunidas em

$$A_{18}^{(8)} = A_{18}^{(6)} \cap A_{18}^{(7)}$$

Tomemos o Conjunto de áreas  $\{A_3, A_{18}^{(8)}\}$  em que  $A_3$  é (pg.32) uma tabela que da os equivalentes em binário aos códigos simbólicos de operação. Pela seleção  $q_{1:3}=q_{2:3}$ , vamos selecionar as linhas definidas nesse conjunto de áreas que unem cada instrução de máquina do programa à correspondente entrada na tabela de conversão, formando o feixe  $B_{10}$ .

Para fazer essa conversão é interessante notar que a representação em binário correspondente a uma instrução, depende não so do codigo simbolico de operação mas de outras características da instrução tais como:

- a) O endereço ser direto ou indireto.
- b) O endereço ser ou não afetado pelo conteúdo de um indexador e, em caso afirmativo, qual deles.
- c) O valor do decremento, se existir
- d) O valor do endereço base

A informação sobre (a)  $\tilde{e}$  fornecida pelo indicador de endereço indireto, que faz parte da instrução em ASSEMBLER. Esse indicador constitui a propriedade  $q_4$  do nosso Espaço de Propriedades e seu conjunto de valores  $\tilde{e}$   $V_4$  = {"b",\*} em que "b" significa 'branco".

A informação sobre (b) é fornecida pela TAG simbólica (propriedade  $q_7$ ), sobre (c) por  $q_8$  e sobre (d) por  $q_5$ .

A FOB F<sub>14</sub> é que monta a palavra em binário correspondente a cada instrução de máquina, em linguagem simbólica. Verificamos que a expressão dessa FOB não é afetada pelo formato do código de máquina correspondente, pois lança mão da operação lógica &, que permite o estabelecimento de relações genéricas, entre os fatores que nela influem.

Notando que as instruções em  $A_{13}^{(8)}$  não incluídas em nenhuma das linhas do feixe  $B_{10}$  são aquelas cujo código simbólico de operação não tem em correspondência nenhum código em  $A_3$ , podemos excrever

$$A_{21} = F_{14}(B_{10}) \cup F_{15}(1, I_{2}(B_{10})) \dots \dots \dots (16)$$
em que
$$F_{14} = (q_{14}^{\dagger} = q_{1:14} \oplus q_{2:4} \oplus q_{2:7} \oplus q_{2:8} \oplus (q_{2:13} + q_{2:6} + (q_{2:5} = "*") + q_{2:5})$$

$$F_{15} = (q_{14}^{\dagger} = \Omega, q_{15}^{\dagger} = "E")$$

$$B_{10} = (q_{1:3} = q_{2:3}, A_{3}, A_{18}^{(8)})$$

Na definição de  $F_{14}$  o valor "\*" pertencente a  $V_5$  significa"conteúdo corrente do indicador de localização", isso leva a calcular o endereço base da instrução, a partir da sua localização em binário, levando em conta o modificador de endereço simbólico (propriedade  $q_{\kappa}$ ).

A FOB  $F_{15}$  marca como erro as linhas do feixe que reune to das as instruções de  $A_{18}^{(8)}$  não incluídas em nenhuma das linhas do feixe  $B_{10}$ .

Temos, pois, em  ${\rm A}_{21}$  todas as instruções de maquina, com en dereços não simbólicos, ou montadas ou marcadas como erro.

Tomando, agora, o conjunto de áreas $\{A_3,A_{17},A_{18}^{(7)}\}$  vamos dar igual tratamento às instruções de máquina com endereço simbólico, lembrando que  $A_{17}$  é a tabela de símbolos.

Pela seleção q<sub>1:3</sub><sup>=q</sup><sub>3:3</sub> <sup>A q</sup><sub>2:2</sub><sup>=q</sup><sub>3:5</sub> formamos nesse conjunto um feixe B<sub>11</sub> em que, cada linha, une uma instrução de máquina à entrada correspondente ao seu código simbólico de operação, na tabela de equivalentes em binário e à entrada correspondente ao

seu endereço simbólico base, na tabela de símbolos.

Por um raciocínio análogo ao utilizado para montar as instruções de máquina com endereço não simbólico, vamos analisar a FOB F<sub>16</sub> que faz a montagem das instruções de máquina com endereço simbólico. Essa função, através da operação lógica  $\Theta$  diz que tal montagem deve levar em consideração: o código em binário correspondente ao código simbólico de operação, o indicador simbólico de endereço indireto, a TAG simbólica, o decremento simbólico e a localização em binário obtida da tabela de símbolos, afetada pelo indicador de endereço simbólico.

Podemos, assim, escrever a expressão da área A<sub>22</sub> que reúne todas as instruções de máquina com endereços simbólicos, montadas ou assinaladas como erro (aquelas que não têm código simbólico de operação válido ou cujo código de endereço simbólico não corresponde a nenhum símbolo da tabela de símbolos).

$$A_{22} = F_{16}(B_{11}) U F_{15}(1,I_3(B_{11})) \dots (17)$$
  
em que

$$F_{16} = (q_{14}^{\dagger} = q_{1:14}^{\oplus q_{3:4}} = q_{3:7}^{\oplus q_{3:7}} = (q_{1:3}^{\dagger} = q_{3:3}^{\dagger} \wedge q_{2:2}^{\dagger} = q_{3:5}^{\dagger}, A_{3}^{\dagger}, A_{17}^{\dagger}, A_{18}^{(7)})$$

Como todas as instruções foram processadas, podemos escrever a expressão de  $A_5$  que constitui a listagem do programa fonte:

$$A_{5} = A_{12}^{(2)} U A_{12}^{(3)} U A_{14}^{(1)} U A_{18}^{(2)} U A_{18}^{(5)} U A_{19} U A_{20} U A_{21} U A_{22}..(18)$$

Tomando o conjunto de áreas formado pela única área  $A_5$  podemos estabelecer, pela seleção  $q_{1:14}\neq\Omega$  o feixe que reune todas as instruções que foram montadas em binário. Pela FOB  $F_{17}$  extraimos a informação desse feixe para constituir o programa objeto, que é o conjunto de pares ordenados  $(q_{13}, q_{14})$  correspondentes a cada instrução do feixe.

$$A_4 = F_{17}(q_{1:14} \neq \Omega, A_5)$$
 .....(19)

em que

$$F_{17} \equiv (q_1! = \Omega, i=1,2,...,11,15)$$

#### 9 - CONCLUSÕES

Na proxima pagina apresentamos as expressões todas, em conjunto, que exprimem as relações entre os dados e os resultados num problema de montagem de programas.

Como tivemos a oportunidade de observar, tais expressões foram obtidas sem qualquer consideração quanto à sequência de passos relativa ao "processo" de montagem de programas. A divisão em fases foi devida mais à necessidade de concentração de atenção em determinados aspectos do que por exigência do método. Tal divisão foi da mesma natureza que a utilizada na solução de qualquer problema algébrico, em que o matemático utiliza duas expressões curtas, quando uma comprida bastaria. Poder-se-ia, perfeitamente, apresentar, ao invês de 19, menor número de expressões mais longas ou até uma somente, ainda que essa não viesse a caber na mesa comum de um escritório.

O certo é que não se precisa pensar em termos de sequência das fases, mas pode-se considerar todas ocorrendo simultaneamente, isto é, pensar como se as expressões correspondentes estivessem substituidas na expressão única que representaria todas as relações, en volvidas no problema.

Tais expressões são independentes também do computador em que será feito o processamento da montagem. Dependem, em alguns as pectos, do computador para o qual a montagem será feita porque um ASSEMBLER é sempre dependente do computador para o qual é construído. Ainda nesse aspecto, todavia, as expressões são bem gerais, pois independem das unidades de entrada e saída, memórias auxiliares ou do fato da montagem ser feita em uma ou duas passagens.

Por exemplo, para adaptar essas expressões ao ASSEMBLER do computador IBM1130 bastaria levar em consideração as diferenças quanto à chamada de subrotinas e ao codigo simbolico, de algumas instruções.

```
B_1 = (q_{112} = q_{212} \land q_{1116} = 1 , H_1(q_2, A_1^{(1)}), A_1^{(1)})
                 A_1^{(1)} = G(q_2, A_1)|_{q_3 = \text{"LIB"}}
                  H_1 \equiv (q_2^* - q_2^*, q_{16}^* - E1)
A_7 = H_2(q_1,F(B_2))
         \mathbf{z}_{2} = (\mathbf{q}_{1:12} - \mathbf{q}_{2:2}, A_{2}, A_{6})
         H_2 = (q_1' - q_1, i-1, 2, ..., 11, q_{16}' - E1)
A_{1} = A_{1}^{(2)} \cup A_{7} \cup F_{1}(1,I_{2}^{*}(B_{2})) \cup F_{1}(1,I_{2}^{*}(B_{1}))
         F1 = (q15="E")
        A_1^{(2)} = A_1 \cap (A_1^{(1)})^{\dagger}
A_9 = H_3(q_1, r_2(B_3)) U I_2(B_3)
         B_3 = (q_{1:1} < q_{2:1}, A_7, A_8)
         r_2 \equiv (q_{17}^{\dagger} - q_{1:16}^{\dagger})
         H_3 = (q_1'-q_1+\epsilon_{q_1}, q_1'-q_1, \epsilon_{q_2}, 3, ..., 11, 15)
A_{10} = F_3(q_{1:12} - q_{2:12}, A_9^{(1)}, A_2)
           F_3 = (q_1' - q_{1:1} + q_{2:1} + q_{12}' - \Omega)
           A_9^{(1)} = G(q_3 dq_{15} + A_9)|_{q_3 dq_{15}} = \text{"LIB"}_{d\Omega}
A11 - A9 U A10
A_{12} = R_4(q_1, P_4(B_4)) \cup I_2(B_4)
           B_4 = (q_{1:1} < q_{2:1}, G(q_3, A_{11})|_{q_3 = BGD}, A_{11})
           r_4 = (q_{16}^1 - q_{1:10}^1)
           H_{4} \equiv (q_{1}^{1}-q_{1}+\Gamma(q_{16}-1)), q_{1}^{1}-q_{1}^{2}, i=2,3,...,11,15)
A_{13} = H_5(q_1, r_5(B_5)) \cup r_5(1, I_2^*(B_5))
          B_5 = (q_{1:1} < q_{2:1}, J_{i=1}^{J} A_{12}^{(i)}, A_{12}^{(4)})
                   A_{12}^{(1)} = G(q_3, A_{12}) |_{q_3 = "SYN"}
                   A_{12}^{(2)} = G(q_3, A_{12})|_{q_3^{m'''REH''}}
                   A_{12}^{(3)} = G(q_3, A_{12})|_{q_3 = LIB}
                   A_{12}^{(4)} = A_{12} \cap (\bigcup_{i=1}^{3} A_{12}^{(i)})^{i}
           F<sub>5</sub> = (q'<sub>18</sub> \(\pi\q_{1:1}\)
           H_5 = (q_1' + q_1, i - 1, 2, ..., 11, q_{18}' + q_1' - \Sigma 1)
A_{14} = H_6(q_1, r_6(B_6)) U I_2(R_6)
           B_6 = (q_{1:1} < q_{2:1}, G(q_3, A_{13})|_{q_3 = n_{BSS}}, A_{13})
           r_6 = (q_{16}^1 - q_{1:5}^1)
           H_6 \equiv (q_1' - q_1, i-1, 2, ..., 11, q_1' - q_{18} + \Sigma(q_{16} - 1))
A_{15} = H_7(q_1, r_7(B_7)) \cup r_8(1, I_2(B_7))
           \mathbf{E}_{7} = (\mathbf{q}_{1:1} < \mathbf{q}_{2:1}, \mathbf{A}_{14}^{(1)}, \mathbf{A}_{14}^{(2)})
                   A_{14}^{(1)} = G(q_3, A_{14})|_{q_3 = 0RG}
                   A_{14}^{(2)} = A_{14} \cap A_{14}^{(1)}
           F_7 \equiv (q_{16}^* q_{1:5} + q_{17}^* q_{1:18})
           H_7 = (q_1^i - q_1, i-1, 2, ..., 11, q_{13}^i - q_{16}(max(q_{17})) + q_{18}^i - max(q_{17}) - 1)
           P_8 = (q_{13}^* - K - 1 + q_{1:15} + q_{18}^* - R)
```

```
\mathbf{B_{8}} = (\mathbf{q_{1:2}} - \mathbf{q_{2:2}} \wedge \mathbf{q_{2:16}} - \mathbf{1} \wedge \mathbf{q_{1:2}} \wedge \mathbf{n_{5}} + \mathbf{n_{1}} + \mathbf{q_{2}} , \lambda_{15} \vee \lambda_{12}^{(1)}), \lambda_{15} \vee \lambda_{15}^{(1)})
A_{17} = A_{16}^{(2)} \cup F_9(B_9)
              B_9 = (q_{1:5} - q_{2:2}, A_{16}^{(1)}, A_{16}^{(2)})
                         A_{16}^{(1)} = G(q_3, A_{16})|_{q_3 = "SYN"}
                        A_{16}^{(2)} = A_{16} \cap (A_{16}^{(1)})
             r_9 \equiv (q_2^{\dagger} - q_{1:2}, q_{13}^{\dagger} - q_{2:13}^{\dagger} + q_{1:6}^{\dagger})
A_{18} = A_{16} \cup F_1(1, I_1^{\dagger}(B_9)) \cup F_{19}(1, I_2^{\dagger}(B_8))
             P_{10} = (q_{15}^1 - \Omega + (q_{112}^2 - b^2) + E^2).
A_{19} = F_{11}(1,A_{18}^{(1)}) \bigcup_{j=1}^{8} F_{12}^{(j)}(q_{1:10}>j,A_{18}^{(1)})
             A_{18}^{(1)} = G(q_3, A_{18})|_{q_3^{-1}BCD''}
             Y_{11} = (q_{14}^{\dagger} - q_{1:11}^{(1)})
             r_{12}^{(j)} \equiv (q_1^i - q_{1+1}^i + j, q_1^i - \Omega, i - 2, 3, ..., 11, q_{13}^i - q_{1:13}^i + j, q_{14}^i - q_{1:11}^{(j+1)})
                              A_{20} = F_{13}(1 , A_{18}^{(3)}UA_{18}^{(4)})
             A_{18}^{(3)} = G(q_3, A_{18})|_{q_3 = \text{"DEC"}}
             A_{18}^{(4)} = G(q_3, A_{18})|_{q_3 = 0CT}
             F<sub>13</sub> E (q<sub>14</sub>=q<sub>1:5</sub>)
A_{21} = F_{14}(B_{10}) \cup F_{15}(1, F_{2}^{\dagger}(B_{10}))
              B_{10} = (q_{1:3} - q_{2:3}, A_3, A_{18}^{(8)})
                           A_{18}^{(8)} = A_{18} \cap (A_{18}^{(7)})
                           A_{18}^{(7)} = G(g_1, A_{18}^{(6)})|_{g_1 = (q_5 \in V_2 - \{"b"\})}
                            A_{18}^{(6)} = A_{18} \cap (\bigcup_{i=1}^{5} A_{18}^{(i)})^{i}
                           A_{18}^{(2)} = G(q_3, A_{18})|_{q_3="BSS"}
                           A_{18}^{(5)} = G(q_3, A_{18}) |_{q_3 = "SYN"}
              \mathbf{r}_{14} = (\mathbf{q}_{14}^{\dagger} - \mathbf{q}_{1:14}^{\dagger} + \mathbf{q}_{2:4}^{\dagger} + \mathbf{q}_{2:7}^{\dagger} + \mathbf{q}_{2:8}^{\dagger} + (\mathbf{q}_{2:13}^{\dagger} + \mathbf{q}_{2:6}^{\dagger} + (\mathbf{q}_{2:5}^{\dagger} + \mathbf{q}_{2:6}^{\dagger} + \mathbf{q}_{2:5}^{\dagger})
              F_{15} \equiv (q_{14}^{\dagger} - \Omega , q_{15}^{\dagger} - E^{\dagger})
A_{22} = F_{16}(B_{11}) \cup F_{15}(1,T_3(B_{11}))
              B_{11} = (q_{1:3} - q_{3:3} \wedge q_{2:2} - q_{3:5}, A_3, A_{17}, A_{18}^{(7)})
              \mathbf{r}_{16} = (\mathbf{q}_{14}^{t} - \mathbf{q}_{1:14}^{t} + \mathbf{q}_{3:4}^{t} + \mathbf{q}_{3:7}^{t} + \mathbf{q}_{3:8}^{t} + \mathbf{q}_{3:6}^{t})
A_5 = A_{12}^{(2)} \ U \ A_{12}^{(3)} \ U \ A_{14}^{(1)} \ U \ A_{18}^{(2)} \ U \ A_{18}^{(5)} \ U \ A_{19} \ U \ A_{20} \ U \ A_{21} \ U \ A_{22}
A_4 = F_{17}(q_{1:14} \neq \Omega, A_5)
           F_{17} \equiv (q_i^* = 0, i = 1, 2, ..., i1, 15)
```

#### CAPÍTULO III

#### O MÉTODO GRÁFICO DE YOUNG & KENT

#### 1 - INTRODUÇÃO

Neste capítulo, apresentamos o método desenvolvido em 1958, na Advanced Systems Research Section, Product Specifications Department, The National Cash Register Company, por John W. Young Jr. e Henry K.Kent.

Esse trabalho foi baseado em estudos de Ackerman, S.A., e publicado sob o título de "Formulação Abstrata de Problemas de Processamento de Dados" (38).

Sammet (32) relaciona o método de Young & Kent ao da Álgebra da Informação da CODASYL (1) e aos trabalhos de Lombardi (16) e do SHARE Comitee on Theory of Information Handling, como sendo uma das primeiras tentativas de formalizar alguns conceitos, na área de processamento de dados.

Trata-se de método predominantemente gráfico, que utiliza, todavia, expressões matemáticas envolvendo operações aritméticas e lógicas. É baseado na Teoria dos Conjuntos e tem capacidade de abstração bastante grande, pois as relações obtidas independem do computador a ser utilizado, da linguagem de programação e mesmo da estrutura de informação a ser usada no processamento. As relações não são afetadas, nem mesmo pela orientação que se adote quanto à organização, manutenção e pesquisa, em arquivos.

Acreditamos que seu emprego seja bastante útil ao Analista de Sistemas de Informação, seja na fase de levantamentos relativos ao sistema existente em uma empresa, seja na fase de projeto de novos sistemas, pois o Método permite a visão global do problema, através da qual podem ser detetadas redundâncias ou estudadas soluções alternativas.

Assim, empregado na análise de um sistema existente, permite verificar se toda a informação que nele entra, é efetivamente, utilizada para produzir os resultados, se resultados intermediários estão sendo plenamente aproveitados, tendo em vista os vários objetivos do processamento, etc.

Utilizado no projeto de novos sistemas, permite julgar da

conveniência de apresentação dos resultados em certa forma, tendo em vista as exigências quanto a dados, orienta de maneira inequívo ca o programador, quanto às finalidades e exigências dos programas que devam ser feitos; auxilia na fase de concepção da estrutura de arquivos a ser usada, no processamento, etc.

No capítulo 0, apresentamos os conceitos matemáticos, que julgamos necessários ao entendimento do método, tal como é apresentado, neste capítulo.

# 2 - COMPONENTES BÁSICOS DOS PROBLEMAS DE PROCESSAMENTO DE DADOS

Um problema de processamento de dados pode ser descrito em termos de quatro tipos de componentes básicos: Conjuntos de Informação, Documentos, Relações e Exigências Operacionais.

#### 2.1- CONJUNTOS DE INFORMAÇÃO

Um conjunto de informação, cuja notação é P<sub>i</sub> é uma relação de todos os itens possíveis, pertencentes a mesma classe. Dos itens dessas relações são extraídos os dados que vão fluir, através do sistema. Os elementos de um conjunto de informação podem ser considerados como todas as possíveis inscrições que possam ser feitas em determinado espaço de um documento.



A notação gráfica utilizada, para representar os conjuntos de informação,  $\tilde{\mathbf{e}}$  exemplificada acima. Essa  $\tilde{\mathbf{e}}$  a representação do conjunto de informação  $\mathbf{P}_3$ . A linha desenhada abaixo do quadro, serve, para que ligações que tivessem de ser feitas ao quadro, para maior clareza de desenho, possam ser feitas  $\tilde{\mathbf{a}}$  linha.

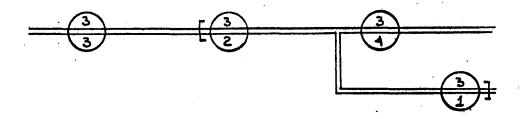
Alguns exemplos de conjuntos de informação seriam: Númerros de peças mantidas em Estoque, Custos unitários das peças, Estoques mínimos a serem mantidos, Nomes dos Almoxarifados, etc.

A efetiva existência de um conjunto de informação ou de uma

coleção de conjuntos de informação correlacionados, no sistema, c<u>o</u> mo um arquivo, vai depender de decisão qu<mark>e se tome em fase pos</mark>terior.

#### 2.2- DOCUMENTOS

Um documento, cuja notação é D<sub>j</sub> é uma coleção de itens de informação correlacionados, D<sub>j-K</sub> em que j é o número do documento e K o do item. Os documentos ou são de entrada ou de saída. Chama se linha de um documento, o conjunto de itens que pode ser repetido várias vezes num único documento. Indica-se uma linha por [ ] ou como novo item.



Acima, está figurado o Documento  $D_3$ , com seus 4 itens: $D_{3-1}$ ,  $D_{3-2}$ ,  $D_{3-3}$  e  $D_{3-4}$ . O ramo inferior do traço duplo foi feito apenas para mostrar que se podem fazer ramificações, caso sejam necessárias para dar maior clareza ao desenho.  $\begin{bmatrix} D_{3-2}, & D_{3-4}, & D_{3-1} \end{bmatrix}$  é exemplo de linha, cuja denominação poderia ser  $D_{3-5}$ .

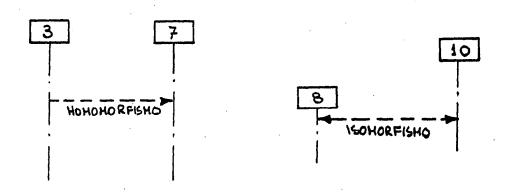
Exemplos de documentos, seriam: Requisição de material, Nota de Recebimento de Material, Inventário do Material em Estoque, etc. Os dois primeiros seriam de Entrada e o último, de Saída.

#### 2.3- RELAÇÕES E CONDIÇÕES

Devem ser descritas as relações entre os conjuntos de informação, documentos e seus itens. Essas relações podem ser: condicionais ou incondicionais.

Uma relação entre conjuntos de informação especifica a correspondência entre os elementos de um conjunto e os de outro. Essa correspondência, se existir, pode ser de dois tipos: unívoca e biu nívoca. Diz-se que é unívoca quando, dado um elemento de um dos conjuntos, pode-se determinar o do outro conjunto que lhe corresponde. Quando essa correspondência funciona, nos dois sentidos, diz-se que ela é biunívoca. Dados dois conjuntos, por exemplo P<sub>3</sub> e P<sub>7</sub>, se existir uma correspondência unívoca entre os elementos de P<sub>3</sub> e os

de  $P_7$  diz-se que existe um homomorfismo entre  $P_3$  e  $P_7$  e utiliza-se a notação  $P_3$ - $P_7$ . Se, dados dois conjuntos, por exemplo  $P_8$  e  $P_{10}$ , existir uma correspondência biunívoca entre seus elementos, isto  $\tilde{e}$ , podendo-se escrever  $P_8$ - $P_{10}$  e  $P_{10}$ - $P_8$  diz-se que, entre eles, existe um isomorfismo e utiliza-se a notação  $P_8$ - $P_{10}$ , uma nomenclatura da Álgebra não muito propriamente, pois nada se exige quanto a operações.

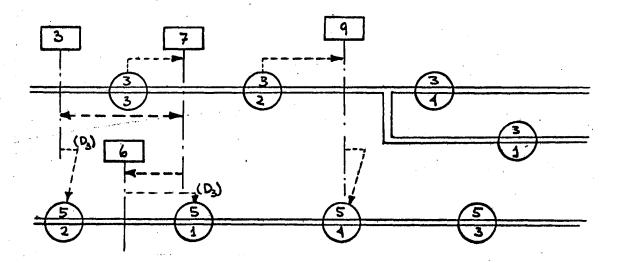


Acima, estão figuradas as relações descritas no parágrafo anterior. Como tais relações são transitivas, se tivessemos P<sub>7</sub><sup>-P</sup><sub>8</sub> poderíamos concluir que as seguintes relações prevaleceriam entre P<sub>3</sub>, P<sub>7</sub>, P<sub>8</sub> e P<sub>10</sub>, além das já mencionadas: P<sub>3</sub><sup>-P</sup><sub>8</sub>, P<sub>3</sub><sup>-P</sup><sub>10</sub>, P<sub>7</sub><sup>-P</sup><sub>10</sub>.

Como exemplo de homomorfismo, podemos citar a relação existente entre o Conjunto dos Números das Peças em Estoque e o dos Cus tos Unitários dessas peças, pois dado um número de peça podemos de terminar seu custo unitário. Dado um custo unitário, não se pode de terminar a peça que a ele corresponde, pois nada impede que mais de uma peça tenha o mesmo custo unitário.

Como exemplo de isomorfismo, podemos citar a relação existente entre o Conjunto dos Números dos Almoxarifados e o dos Nomes desses Almoxarifados, pois não teria sentido dar-se o mesmo número a mais de um Almoxarifado, nem dar-se mais de um número a um mesmo Almoxarifado.

A relação entre um item de um documento e um conjunto de informação mostra que a informação desse item é levada ao conjunto de informação, se o documento é de entrada, ou é tomada do conjunto de informação, se for de saída.



Acima, estão figuradas relações entre itens dos documentos  $D_3$  e  $D_5$  e os conjuntos de informação  $P_3$ ,  $P_6$ ,  $P_7$  e  $P_9$ .  $D_3$  sendo um documento de entrada, a relação entre  $D_{3-3}$  e  $P_7$  mostra que a informação desse item é levada ao Conjunto de Informação  $P_7$ . Analogamente, a informação de  $D_{3-2}$  é levada a  $P_9$ . Por outro lado,  $D_5$  sendo um documento de saída, as relações figuradas mostram que a informação de  $D_{5-1}$  é tomada de  $P_6$ , a de  $D_{5-2}$  de  $P_3$  e a de  $D_{5-4}$  de  $P_9$ .

A relação entre  $D_{5-2}$  e  $P_3$  indica que o valor de  $D_{5-2}$  deve ser tomado de  $P_3$  correspondente aquele que foi levado a  $P_3$  pelo  $D_3$  correspondente. É indicado  $P_3(D_3)$  e, ainda que  $D_3$  não leve nenhuma informação diretamente a  $P_3$ , pelas relações entre os conjuntos de informação, essa informação está bem definida. Em nosso exemplo, como  $P_3 = P_7$  e  $D_{3-3}$  leva informação a  $P_7$ , pode-se determinar  $P_3(D_3)$  como sendo o valor de  $P_3$  correspondente ao valor levado a  $P_7$  por  $D_{3-3}$ . Analogamente,  $P_6(D_3)$  que vai fornecer a  $D_5$  o valor de  $D_{5-1}$ , indica que o valor a ser tomado de  $P_6$  é aquele que corresponde ao levado a  $P_7$  por  $D_{3-3}$ .

Se existisse a relação  $P_9^-P_7$ , ficaria a critério do sistema tomar a informação correspondente à levada a  $P_7$  por  $D_{3-3}$ , como mencionado acima, ou a correspondente à levada a  $P_9$  por  $D_{3-2}$ , conforme fosse julgado mais conveniente.

Quanto a  $D_{3-1}$ ,  $D_{3-4}$  e  $D_{5-3}$ , como está figurado, não têm relação direta com nenhum conjunto de informação.

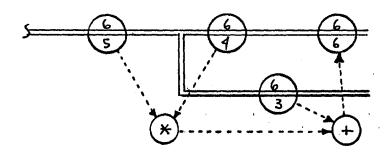
Como exemplo de relações entre itens de documentos e con-

juntos de informação, podemos citar: A informação correspondente ao item da Requisição de Material "Número da Peça requisitada" é levado ao Conjunto dos Números das Peças mantidas em Estoque; a informação correspondente ao item do Inventário do Material em Estoque - "Custo Unitário" é tomada do Conjunto dos Custos unitários das Peças.

Uma relação de definição é aquela que mostra como o item de um documento de saída é derivado, ou de onde deve ser tomado, se for simplesmente copiado de outro documento.

Na pagina anterior, apresentamos dois exemplos de relação de definição: as que definem  $D_{5-1}$  e  $D_{5-2}$  e que são, respectivamente,  $P_6(D_3)$  e  $P_3(D_3)$ .

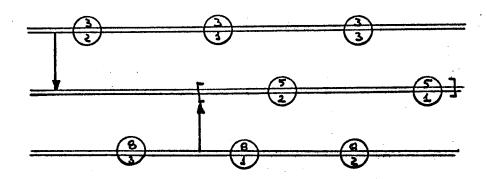
Na expressão das relações de definição são utilizadas operações aritméticas, cuja notação pode ser observada, no quadro da pag. 72.



Acima, está esquematizada a relação de definição de  $D_{6-6}$ , cuja expressão é escrita  $D_{6-5}^{*}D_{6-4}^{+}D_{6-3}$ , isto é, o item  $D_{6-6}^{-}$  é obtido multiplicando-se o valor de  $D_{6-5}^{-}$  pelo de  $D_{6-4}^{-}$  e somando-se o resultado ao valor de  $D_{6-3}^{-}$ . Pressupõe-se que os conjuntos de informação a que estão ligados  $D_{6-5}^{-}$ ,  $D_{6-4}^{-}$  e  $D_{6-3}^{-}$  sejam conjuntos numéricos, a fim de que tenha sentido a aplicação das operações.

Como exemplo de relação de definição, podemos citar: O item "Valor do Material em Estoque" do documento Inventário do Material em Estoque é obtido multiplicando-se os valores dos itens "Quantidade em estoque" e "Custo unitário" de cada linha do documento e fazendo-se a somatória dos resultados para todas as linhas.

Uma relação de produção mostra sob que estímulo um documento ou um conjunto de itens de um documento são produzidos. A notação utilizada  $\tilde{e}$   $D_K^{+D}$ , e  $1\hat{e}$ -se  $D_K$  produz  $D_i$ .



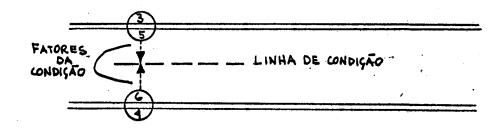
No esquema acima, estão figuradas 2 relações de produção:  $D_3 \rightarrow D_5$  e  $D_8 \rightarrow D_{5-3}$  isto é, todo documento  $D_3$  provoca a emissão de um documento  $P_5$  e todo o documento  $D_8$  produz uma linha de  $P_5$ .

Como exemplo de relação de produção, poderíamos citar que em "todo o semestre é produzido um Inventário de Material em Estoque".

As relações até agora apresentadas, são todas incondicionais. Vejamos, agora, como são expressas as relações condicionais e as condições sob as quais prevalecem.

Na expressão das condições são utilizadas relações usuais matemáticas e operações lógicas, cuja notação pode ser observada no quadro da pg. 72.

Uma condição pode constar apenas de uma relação usual que prevaleça ou não, efetivamente. Caso prevaleça a condição é verdadeira, caso contrário é falsa.

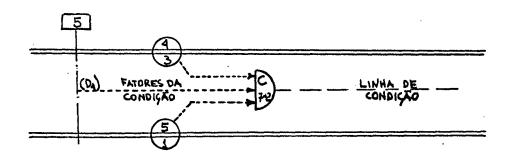


Acima, está figurada a condição formada pela relação usual  $^{\rm D}_{3-5}$   $^{\rm =D}_{6-4}$ , ou seja, a informação contida em  $^{\rm D}_{3-5}$  é igual à contida em  $^{\rm D}_{6-4}$ .

Uma condição pode ser formada por várias relações usuais, ligadas por operações lógicas. Por exemplo, podemos ter a seguinte expressão de uma condição  $P_5(D_4)< D_{4-3}$   $\Lambda D_{5-1}< 8$ . Nesse caso, a condi

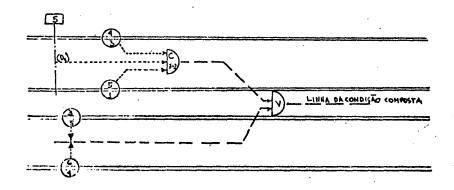
ção é verdadeira se o valor tomado do Conjunto de Informação  $P_5$  correspondente ao documento  $D_4$  for menor que o valor do item  $D_{4-3}$  e se o valor do item  $D_{5-1}$  for menor que 8.

Utiliza-se para expressar uma condição a notação  $C_{m-n}$  sendo m o número do documento a que a condição se refere e n o número da condição referente ao documento. Considera-se que existe uma variável booleana associada a cada condição, que assume valor l.se a condição for Verdadeira e 0 se, Falsa.



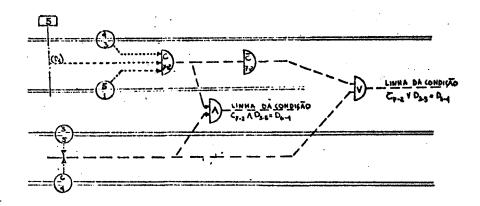
Acima, está figurada a condição  $C_{7-2}$ , isto é, a condição número 2 referente ao documento 7. Conforme está indicado, são fatores dessa condição  $P_5(D_4)$ ,  $D_{4-3}$  e  $D_{5-1}$ . Para que  $C_{7-2}$  fique perfeitamente definida, é necessário que se dê a expressão de  $C_{7-2}$  que poderia ser, por exemplo,  $P_5(D_4) < D_{4-3}$   $\Lambda D_{5-1} < 8$ .

Uma condição pode ser composta de outras. Poderíamos por exemplo, ter a condição  $D_{3-5} = D_{6-4}$  V  $C_{7-2}$  que seria figurada conforme o esquema abaixo:

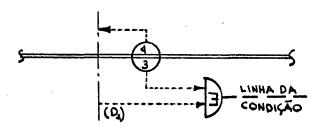


Uma barra sobre a condição  $\tilde{e}$  utilizada para expressar a sua negação. Assim, no exemplo apresentado anteriormente,  $\tilde{c}_{7-2}$  corresponderia a

 $(P_5(D_4) \geqslant D_{4-3} \land D_{5-1} < 8) \lor (P_5(D_4) < D_{4-3} \land D_{5-1} \geqslant 8) \lor (P_5(D_4) \geqslant D_{4-3} \land D_{5-1} \geqslant 8).$  No esquema abaixo, estão figuradas duas condições compostas, uma de las utilizando a condição  $C_{7-2}$  e a outra a negação de  $C_{7-2}$ :



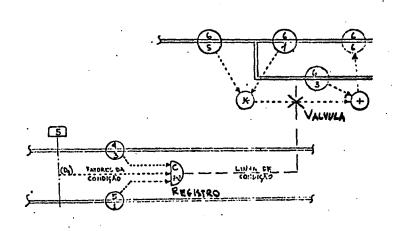
Outro exemplo interessante de condição é a que envolve a existência de informação no sistema, com determinadas características. Esse seria o caso da condição:  $\exists D_4(D_{4-3}(D_{2-1}))$  significando que existe o documento  $D_4$  em que o item  $D_{4-3}$  é função do item  $D_{2-1}$  do Documento  $D_2$ , isto é,  $D_{4-3}$  depende do valor que consta de  $D_{2-1}$ .  $D_4$  poderia ser um documento de entrada e  $D_2$  de entrada ou de saída.



No esquema acima, está figurada essa condição. A linha vertical de traço e ponto corresponderia ao conjunto de informação ao qual estariam relacionados  $D_{4-3}$  e  $D_{2-1}$ .

Vejamos, agora, como são expressas as relações condicionais.
Para podermos comparar os efeitos da utilização de condi-

ções, suponhamos que a relação de definição incondicional de  $D_{6-6}$ , que consta do exemplo apresentado na pg.62 passasse a ser afetada pe la condição  $C_{7-2}$  que consta do exemplo apresentado na pg.64, tornando-se, então, uma relação condicional. Agora, a relação de definição de  $D_{6-6}$  poderia ter a expressão  $D_{6-3}^{+C}$ ,  $D_{6-5}^{+D}$ , isto é, a parcela  $D_{6-5}^{+D}$ , só seria acrescentada a  $D_{6-3}^{-2}$ , se a condição  $C_{7-2}^{-2}$  fosse verdadeira. O ponto que consta da expressão, entre  $C_{7-2}^{-2}$  e  $D_{6-5}^{+D}$ , indica que o valor da variável booleana associada a  $D_{6-5}^{-2}$  deve ser multiplicado por  $D_{6-5}^{+D}$ , Assim, se  $D_{7-2}^{-2}$  for verdadei ra, a variável booleana associada terã valor  $D_{6-5}^{-2}$  a variável booleana associada terã valor  $D_{6-5}^{-2}$  for verdadei nada, enquanto que se  $D_{7-2}^{-2}$  for falsa, a variável terã valor  $D_{6-5}^{-2}$  e adiciona-se  $D_{6-5}^{-2}$  ao invés da parcela  $D_{6-5}^{-2}$ , a  $D_{6-3}^{-3}$ . Abaixo figura o esquema gráfico dessa relação de definição condicional.

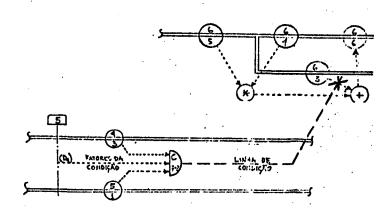


Para que fique bem claro o funcionamento do esquema acima, podemos estabelecer analogia com um esquema hidráulico em que  $C_{7-2}$  seria um registro comandado pelos fatores da condição, que abriria ou não a válvula, conforme a condição fosse verdadeira ou falsa. As sim, se os fatores da condição forem tais, que ela seja verdadeira, a válvula é aberta e a informação passa por ela, que está interceptando a linha correspondente à relação de definição e  $D_{6-6}$  é obtido pela soma de  $D_{6-3}$  com  $(D_{6-5}*D_{6-4})$ . Caso contrário, fica fechada e a informação não passa por ela, com o que  $D_{6-6}$  passa a ser obtido como sendo igual apenas a  $D_{6-3}$ .

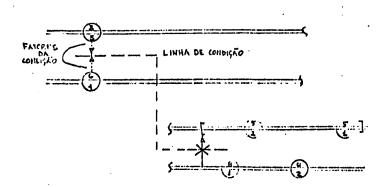
Outra maneira, pela qual a condição  $C_{7-2}$  poderia afetar a relação de definição de  $D_{6-6}$ , seria de acordo com a expressão  $C_{7-2}$ .  $D_{6-3}$ + $D_{6-5}$ \* $D_{6-4}$ , isto E, agora a parcela  $D_{6-3}$  E que seria ou não

adicionada, conforme a condição C<sub>7-2</sub> fosse Verdadeira ou Falsa.

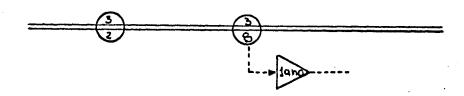
Apresentamos a seguir o esquema gráfico dessa nova configuração.



Concluimos que  $D_{6-6}$  é obtido pela expressão  $D_{6-3}^{+D} = 0.5 \times 0.00$  se  $C_{7-2}$  for verdadeira e pela expressão  $D_{6-5} \times 0.00$  se  $C_{7-2}$  falsa. Ambos os exemplos apresentados são de relação de definição condicional. Vejamos exemplo de relação de produção condicional. Su ponhamos que a relação de produção  $D_{8} \times D_{5-3}$ , apresentada na pg.63 pas sasse a ser afetada pela condição  $D_{3-5} = 0.00$  apresentada na mesma pg. A expressão da relação de produção passaria a  $D_{8} \times 0.00$  apresentada na mesma pg. A expressão da relação de produção passaria a  $D_{8} \times 0.00$  esquema gráfico correspondente seria:



Para finalizar esta apresentação dos detalhes do Método de Young & Kent, referentes às relações e condições, convem cuidar do aspecto tempo. São considerados dois tempos distintos: intrínseco e extrínseco cujas notações são, respectivamente I e E. Tempo intrinseco é aquele que consta do documento, como um de seus itens . Tempo extrínseco é aquele em que o documento é produzido: Por exem plo, uma Ordem de Fornecimento de Material pode ser produzida dia 25, com data de 26, pois precisa ser assinada antes de ser enviada. Nesse caso, temos  $t_1=26$  e  $t_E=25$ . Se  $D_{5-2}$  for o item documento destinado ao registro do dia a que se refere o documento, teremos  $D_{5-2}=26$ . Suponhamos que  $P_8$   $\tilde{\epsilon}$  o conjunto de informação qual D<sub>5-2</sub> está relacionado, isto é, P<sub>8</sub> é o conjunto dos dias que po dem ocorrer. Podemos, então, nos referir a P<sub>8</sub>(t<sub>E</sub>(D<sub>5</sub>)) = 25  $P_8(t_1(D_5)) = 26$  ou, de maneira condensada, tendo em vista que  $P_8$  e. um conjunto de informação formado por unidades de tempo  $P_{8E}(D_5)=25$  $P_{81}(D_5)=26$ . Vejamos outro exemplo, para esclarecer melhor. Seja  $D_3$ um documento emitido todo o ano, por exemplo, o Inventário de Mate rial em Estoque. Seja D<sub>3-2</sub> o item referente ao ano, isto é, inscreve-se em D<sub>3-2</sub> o ano a que corresponde ao Inventário. Seja P<sub>2</sub> o conjunto de informação a que  $D_{3-2}$  está relacionado. Seja  $D_{3-8}$ ítem do Inventário em que se inscreve o valor total do Estoque e s<u>u</u> ponhamos que se queira, para a preparação do Inventário de um ano, fazer referência ao valor total do Estoque do Inventário anterior. Podemos escrever  $D_{3-8}(D_{3-2}-1)$  e graficamente figurar:



## 2.4- EXIGÊNCIAS OPERACIONAIS

Esse componente básico diz respeito a fatores que afetam os problemas de processamento de dados, sem, todavia, influir na lógica desses problemas. Por exemplo: Volume diário de documentos de entrada, número de vias necessário para os documentos de saída, tamanho e cor dos documentos, prazos para sua preparação pelo sistema, etc.

No quadro da pg.72 consta a notação utilizada para a expressão das exigências operacionais, que não são representadas nos esquemas gráficos.

## 3 - MARCHA DE SOLUÇÃO

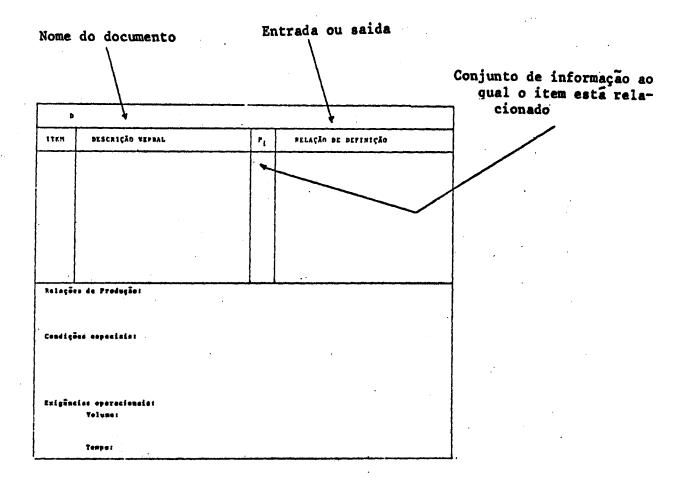
Deve-se iniciar a solução de um problema, pelo método de Young & Kent, identificando os conjuntos de informação e preparando uma lista com as informações abaixo indicadas:

	CONJUNTOS DE INFORMAÇ	ÃO		
P <sub>i</sub>	Nome do Conjunto	N N de eleme <u>n</u> tos	L Nº de cara <u>c</u> teres	Relações

L é o número de caracteres numéricos (N), alfabéticos (A) e alfanuméricos (A/N) necessários para representar cada elemento do conjunto. Especial atenção deve ser dada à especificação das relações entre os conjuntos de informação. Essas, além das jã menciona das em 2.3, podem existir não diretamente entre os conjuntos de informação, mas entre o produto cartesiano de dois conjuntos e um terceiro, entre conjuntos de informação e conjuntos de documentos, etc.

Deve-se também preparar uma lista dos documentos envolvidos, constando para cada um as informações a seguir indicadas:

Para os documentos de entrada, basta mencionar relação de definição para os itens que se referem as linhas dos documentos. Nesse caso, a relação de definição indica, simplesmente, os itens de que a linha é formada. Deve-se também mencionar os volumes que ocorrem, tanto dos documentos, como de suas linhas.



Para os documentos de saída, pode-se deixar de mencionar a relação de definição apenas quanto aos itens que são tomados direta ou indiretamente dos conjuntos de informação a que estão relacionados. Indiretamente, refere-se ao caso em que, pelas relações entre os conjuntos de informação, esteja bem claro como o item é derivado. Em todos os demais casos, deve figurar, para cada item, a relação de definição a ele correspondente, listando-se também as condições especiais que afetam essas relações de definição.

Deve-se ainda deixar bem claro quais as relações de produção existentes. Para tanto, é bom lembrar que, em geral, um documen to é produzido em uma das seguintes circunstâncias:

- 1- Periodicamente, isto é, diaria ou semanal ou quinzenal mente, etc.
- 2- Irregularmente, isto é, toda vez que há ocorrência de um documento de entrada, ou de uma condição que depende da entrada de informação, no sistema.

As relações de produção dizem respeito à existência, ao inves de ao conteúdo, do documento a que se referem. Em geral, todavia, os itens dos documentos de saída são derivados do documento

que os produz.

O traçado gráfico paralelo às providências anteriormente enumeradas auxilia a formação de uma visão global do problema, e <u>a</u> juda a chamar a atenção para as relações que devem ser definidas.

Inicia-se o traçado, marcando-se as linhas duplas correspondentes aos documentos e depois marcando os conjuntos de informa
ção e os itens dos documentos, da maneira mais conveniente para as
ligações a serem feitas. Desenham-se, a seguir, essas ligações, bem
como as relações de produção. Em seguida, cuida-se de cada relação
de definição incondicional. Finalmente, desenham-se as condições
especiais e as relações condicionais.

Completo o traçado, analisa-se a solução quanto à redundância de dados, aproveitamento de resultados intermediários e pos sibilidade de modificações para maior rendimento no aproveitamento da informação.

Pode-se, a essa altura, melhor estimar custos de produção dos documentos de saída, para julgar da conveniência de produzi-los, estudar a substituição de informação na entrada, para tornar mais econômica a coleta de dados, enfim, tomar medidas para otimizar o funcionamento do sistema.

Uma vez atingida a solução desejada, a apresentação final deve incluir os quadros com as características dos sistemas de informação; dos documentos, inclusive quanto às exigências operacionais e o esquema gráfico, que mostrem claramente as relações entre os dados e os resultados do problema de processamento de dados.

Na pagina seguinte, apresentamos um resumo dos símbolos e da notação gráfica que devem ser utilizados.

No próximo capítulo, apresentaremos a solução completa de dois problemas de processamento, pelo método aqui exposto, de modo a que se possa observar o funcionamento do método, em problemas típicos daqueles, para cuja solução, foi desenvolvido.

- Tempo intrinseco

- Operação aritmética de Adição

- Operação aritmética de subtração

- Operação aritmética de multiplicação

- Operação aritmética de divisão - Lista de todas as informações possíveis pertencentes à mesma classe. - Um membro específico da classe. - "E" lógico - Um Documento "OU" logico - Documento específico pertencente à classe D;. "NÃO" logico D<sub>ik</sub> - Coleção de inscrições no documento D<sub>i</sub> "SE" lógico - Linha de um documento Relação "Igual" C<sub>man</sub>- n-ésima condição relativa ao m-ésimo documento Relação "Diferente" ( ) - Punção - Relação "Maior" - Relação "Menor" - Isomorfo - Homomorfo - Relação "Maior ou igual" - Produto cartesiano - Relação "Menor ou igual" - Contido - Produz - Tempo extrinseco

- Quantidade

- Quantidade media

- Condiçãotde igualdade

REFAÇÃO DE PRODUÇÃO LINHA REPRESENTATIVA DE UN CONSUNTO DE INFORMAÇÃO RELAÇÃO ENTRE CONSUNTOS DE INFORMAÇÃO bloco representativo de um conjunto de THEORHAM LIÇÃO ENTRE UM ITEM DE UM DOCUMENTO E VH CONSUNTO DE INFORHAÇÃO -[3 ATORES DE UHA CONDIÇÃO LINHA DA CONDIÇÃO ITEM DE DOCUHENTO IHITE DE LI-DOCUMEN-RETARDAHENTO RELAÇÃO DE DEFINIÇÃO CONDICIONAL VALVULA CORRESPONDENTE À CONDIÇÃO RELAÇÃO DE DEFINIÇÃO INCONDICIONAL LINHA DA CONDIÇÃO REGISTRO CORRES PONDENTE À CONDIÇÃO

## CAPITULO IV

# COMPARAÇÃO DOS METODOS

## 1 - INTRODUÇÃO

Neste capítulo, apresentamos a solução de dois problemas típicos de processamento de informação, pelos dois métodos estudados, a fim de que se possa compará-los.

O primeiro, referente ao processamento da folha de pagamento de uma indústria, é apresentado em (1) para ilustrar a utilização da Álgebra da Informação da CODASYL. Assim, a solução algébrica por nos apresentada é uma compilação da solução lá formulada. Já a solução gráfica é de nossa responsabilidade.

O segundo, cuida do sistema de faturamento, controle de vendas e de cobrança de uma indústria e foi apresentado em (38) para ilustrar o Método Gráfico de Young & Kent para a formulação abstrata de problemas de processamento de dados. A solução algébrica que apresentamos é de nossa responsabilidade, enquanto que a solução gráfica foi por nos, transcrita, dessa publicação.

Trata-se, por conseguinte, de problemas típicos, cuja solução foi visada ao serem concebidos ambos os métodos estudados. A
ideia de compará-los surgiu da forma pela qual são citados por Sam
met em (32). Miss Jean Sammet faz essa citação, ao abordar as lin
guagens especiais de programação, e considera que o Método Gráfico
de Young & Kent foi um dos precursores da Álgebra da Informação da
CODASYL.

De fato, não devemos esquecer que o Método de Young & Kent não se limita aos esquemas gráficos, mas monta também expressões, tanto para as condições, como para as relações de definição, de produção e para as exigências operacionais.

Miss Sammet cita, paralelamente, o trabalho de Lionello Lombardi (16) referente a uma álgebra de arquivos. Estudamos esse trabalho, mas não o incluimos nesta dissertação, porque Lombardi es tá mais preocupado com linguagem para a manipulação de arquivos, do que com o objetivo dos dois métodos, por nos apresentados. Assim é que a álgebra de arquivos que apresenta, figura no final de seu trabalho e o exemplo de sua aplicação não é muito ilustrativo. Pre-

tendemos, todavia, no prosseguimento de nossos estudos, voltar a <u>a</u> nalisar o trabalho de Lombardi, já com outra preocupação que dê continuidade à pesquisa de que trata esta dissertação.

Miss Sammet cita ainda os trabalhos do SHARE Comitee on Theory of Information Handling, aos quais não tivemos acesso, apesar dos esforços empregados em tentar consegui-los.

# 2 - UM PROBLEMA DE PAGAMENTO DE PESSOAL

#### 2.1- ENUNCIADO

Uma indústria tem todos os seus operários contratados na base de salário-hora. O período normal de trabalho é de 8 horas por dia e 5 dias por semana. O trabalho além da jornada de 8 horas é considerado extraordinário e pago na base de 1.5 vezes o salário normal. O trabalho além dos 5 dias so é considerado extraordinário quando ultrapassa a jornada normal completa, isto é, quando o operário trabalhou em horário normal 8 horas por dia, todos os 5 dias, ou, em caso de haver, em algum dia, trabalhado menos, tenha compensado, além dos 5 dias, as horas não trabalhadas.

Vejamos alguns exemplos para ilustrar o critário adotado.

ноі	RAS	TR	ABAL	HAD	AS	TOTAL	HORAS NORMAIS	HORAS EXTRAS
8	8	8	8	8		40	40	0
8	12	8	10	8		46	40	6
8	8	8	8	8	4	44	40	4
8	12	8	10	8	4	50	40	10
6	8	7	8	8		37	37	0
6	12	7	10	8		43	37	6
6	12	7	10	8	3	46	40	6
6	12	7	10	8	5	48	40	8

Cada operário tem um número de identificação e o saláriohora á individual.

O pagamento é feito semanalmente e mantém-se, para cada operário, registro da importância total a ele paga, desde o início do ano.

As características das informações envolvidas nesse problema constam do quadro abaixo.

CARACTERÍSTICAS DAS INFORMAÇÕES ENVOLVIDAS	QUANTIDADE	NHO DO
INFORMAÇÃO	QUAN	TAMANHO
Número de identificação do operário	1200	. 5N
Nome do operário	1200	20A
Salārio-hora	47	4 N
Quantidade de horas		2 N
Dia do período	7	1 M
Período de pagamento	52	2 N

Ha três documentos nesse sistema: dois de entrada (Cartão de Ponto e Boletim de Admissão) e um de saída (Folha de pagamento).

, D	D <sub>1</sub> - CARTÃO DE PONTO				
ITEM	DESCRIÇÃO VERBAL				
D <sub>1-1</sub>	Número do período				
D <sub>1-2</sub>	Número de identificação do operário				
[P <sub>1-3</sub>	Número do dia do período				
01-4]	Hores trabalhadas nesse dia				
01-5	Linha				

D	D <sub>2</sub> - Boletin de Admyssão				
ITEM	DESCRIÇÃO VERBAL				
D <sub>2-1</sub>	Número do período				
[D <sub>2-2</sub>	Número de identificação do operário				
D <sub>2-3</sub>	Nome do operário				
D <sub>2-4</sub> ]	Salārio-hora				
D <sub>2-5</sub>	Linha				

D	D <sub>3</sub> - FOLHA DE PAGAMENTO			
ITEM	DESCRIÇÃO VERBAL			
D <sub>3-1</sub>	Número do período			
[D <sub>3-2</sub>	Número de identificação do operário			
D <sub>3-3</sub>	Nome do aperário			
D <sub>3-4</sub>	Salārio-hora			
D <sub>3-5</sub>	Salārio do período			
D <sub>3-6</sub> ]	Salário acumulado			
P <sub>3-7</sub>	Linha			

Cada operário tem um cartão de ponto, em cada período, no qual, em cada uma das sete linhas, consta o total de horas traba-lhadas, no dia correspondente.

Em cada período são feitas admissões para completar o quadro de pessoal, considerando-se as demissões que, eventualmente, tenham ocorrido. Essas admissões constam do boletim correspondente ao período, que, em média, relaciona vinte novos operários.

A folha de pagamento é produzida dentro de três dias, a partir do encerramento do período a que corresponde. Em média, constam da folha 1200 operários. Um operário entra em folha, quando co meça trabalhar e só sai, quando é demitido. Enquanto pertencer ao quadro de pessoal da indústria, em todo o período tem um cartão de ponto. Se faltou em todo o período, ainda assim entra em folha com salário zero. Quando em férias, seu cartão é apontado pela seção de pessoal, com as horas correspondentes.

## 2.2- SOLUÇÃO ALGEBRICA

O Espaço de Propriedades no qual vamos estudar o problema tem o seguinte conjunto de coordenadas:

	CONJUNTO DE COORDENADAS DO ESPAÇO DE PROPRIEDADES	)	<b>A</b> 1	^2	A 3	A4
	PROPRIEDADE	CONJUNTO DE VALORES	PACAMENTO ANTERIOR	PONTO DO PESSOAL	NOVAS ADMISSÕES	NOVO PACAMENTO
1	Número de identificação do operário	00000	×	×	×	×
2	Nome do operário	20 caracteres alfabéticos	×	ព	×	×
3	Salario-hora	00.0099.99	×	ß	×	×
4	Quantidade de horas	000232	Ω	×	Ω	Ω
5	Dia do período	07	Ω	×	Ω	Ω
6	Salário acumulado	00000.00	×	Ω	ຄ	*
7	Período de pagamento	0052	×	Ω	×	×
8	Salário do período	000.00	×	Ω	Ω	×

Os arquivos de entrada são:

 $A_1$  - Pagamento anterior, no qual, cada registro, engloba as informações referentes a um operário, tais como: número de iden

tificação, nome e salário-hora, que são os dados fixos e salário a cumulado, que é atualizado em cada processamento.

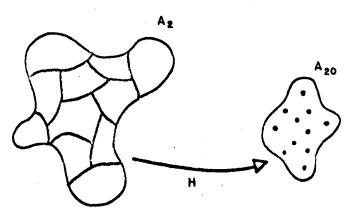
A<sub>2</sub> - Ponto do pessoal, no qual, cada registro, corresponde a uma linha do cartão de ponto de um operário, referente ao período.

A3 - Novas admissões, no qual, cada registro, corresponde aos dados de novo operário. Como um operário pode ser admitido em um período e só começar a trabalhar em outro, esse arquivo não con têm apenas os dados operários admitidos no período a que se refere a folha em processamento, mas também dos admitidos em períodos anteriores, que, por algum motivo, não tenham ainda entrado em folha.

A<sub>4</sub> - Novo pagamento, no qual, cada registro, corresponde aos dados de um operário que trabalha na indústria. Os registros são criados quando aparece cartão de ponto em um período e são removidos quando deixa de aparecer cartão, referente ao operário, no arquivo de Ponto do Pessoal, no processamento relativo a certo período.

Vamos, inicialmente, calcular, para cada operário, o total de horas trabalhadas no período. Para isso, vamos considerar cada hora extraordinária como valendo 1.5 da hora normal; assim, o total de horas será expresso em equivalente a horas normais.

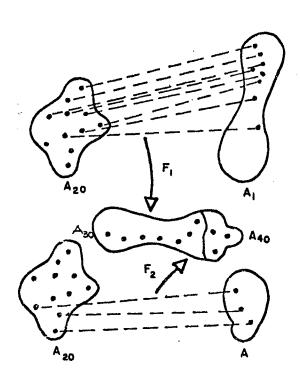
Estabeleçamos em  $A_2$  a malha determinada pela função  $q_1$ , na qual cada elemento reunirá os pontos relativos a determinado operário.



 $A_{20} = H(q_2, A_2)$   $H = (q_1^{\dagger} = q_1, q_4^{\dagger} = \Sigma(q_4 + (q_4 \leq 8) + 1.5 + q_4 - 4) + f_1)$   $f_1 = 0 + (f_2 \leq 40) + .5 + f_2 - 20$   $f_2 = \Sigma(q_4 + (q_4 \leq 8) + 8)$ 

Pela FOG H, podemos associar a cada elemento da malha um ponto cujas propriedades relevantes são o número de identificação do operário e o total de horas trabalhadas no período. Na expressão de H, estão envolvidas operações "se-senão" e as FOA f<sub>1</sub> e f<sub>2</sub> que permitem montar algebricamente o critério, descrito, no enunciado de atribuição de horas extraordinárias.

Agora, precisamos procurar em  $A_1$  ou  $A_3$  os demais dados referentes a cada operário cujo trabalho, no período, é representado por um ponto de  $A_{20}$ . Faremos isso através de dois feixes, um esta belecido no conjunto de áreas  $\{A_{20}, A_1\}$  e outro no conjunto  $\{A_{20}, A_3\}$ , ambos determinados pela seleção  $q_{1:1}=q_{2:1}$ 



Com os dois feixes cobrimos as duas possibilidades: o operário que trabalhou no período, isto é, que teve suas horas apontadas, ou já estava em folha ou consta do arquivo de novas admissões.

Assim, podemos, em resumo, escrever a expressão que relaciona  $^{\rm A}_4$  a  $^{\rm A}_1$ ,  $^{\rm A}_2$  e  $^{\rm A}_3$ , isto é, a nova folha aos dados necessários à sua preparação.

$$A_{4} = F_{1}(q_{1:1} = q_{2:1}, H(q_{2}, A_{2}), A_{1}) \cup F_{2}(q_{1:1} = q_{2:1}), H(q_{2}, A_{2}), A_{3})$$

$$H = (q'_{1} = q_{1}, q'_{4} = \sum (q_{4} + (q_{4} \le 8) + 1.5 * q_{4} - 4) + f_{1})$$

$$f_{1} = 0 + (f_{2} \le 40) + 5 * f_{2} - 20$$

$$f_{2} = \sum (q_{1} + (q_{4} \le 8) + 8)$$

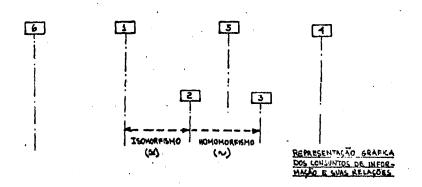
$$F_{1} = (q'_{6} = q_{2:6} + q_{1:4} * q_{2:3}, q'_{7} = q_{2:7} + 1, q'_{8} = q_{1:4} * q_{2:3})$$

$$F_{2} = (q'_{6} = q_{1:4} * q_{2:3}, q'_{7} = q_{2:7} + 1, q'_{8} = q_{1:4} * q_{2:3})$$

## 2.3- SOLUÇÃO GRÁFICA

Identifiquemos, inicialmente, os conjuntos de informação envolvidos no problema. No quadro abaixo, estão descritas suas características que foram obtidas do quadro da pg. 75 do enunciado, bem como as relações existentes entre os conjuntos.

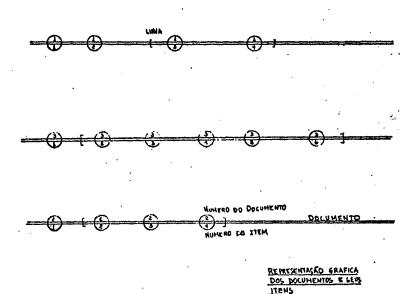
	ONJUNTOS DE INFORMAÇÕES NO de		L Nº de	Relações
Pi	NOME DO CONJUNÇO	elemen tos	carac teres	Kelaçoea
Pi	Número de identificação do operário	1200	5 <b>%</b>	P <sub>1</sub> = P <sub>2</sub>
P <sub>2</sub>	Nome do operário	1200	20A	P1 = P2 - P3
P 3	Salārio-hore	47	4N	P2 - P3
P <sub>4</sub>	Quantidade de horas		211	
P <sub>5</sub>	Dia do paríodo	7	1N	
P 6	Período de pagamento	52	2 N	



A cada operário corresponde certo número de identificação. Por outro lado, a cada número de identificação corresponde certo o perário. Existe, assim, uma correspondência biunívoca, entre o conjunto dos números de identificação e o conjunto dos nomes dos operários, isto  $\tilde{e}$ , há um isomorfismo, entre os dois conjuntos. Podemos pois escrever  $P_1 \simeq P_2$ .

Já quanto aos salários-hora, a cada operário corresponde um salário-hora mas, a cada salário hora vão corresponder um ou mais operários, pois para 1200 operários existem apenas 47 salários hora diferentes. Dado um operário, obtemos imediatamente seu salário, mas não reciprocamente, isto  $\tilde{e}$ , o conjunto dos operários  $\tilde{e}$  homomorfo ao conjunto dos salários-hora. Daí escrevermos:  $P_2-P_3$  ou,  $p_2$  la transitividade  $P_1-P_3$ .

Vamos, a seguir, cuidar dos documentos do sistema. A partir das informações constantes do enunciado (quadros da pg.75)cons truimos o esquema figurado abaixo:

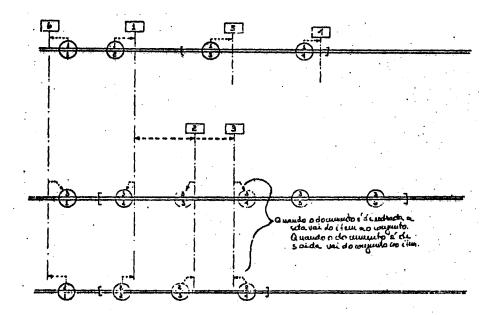


Vejamos quais as relações existentes entre os itens dos do cumentos e os conjuntos de informação. Nos quadros a seguir, estão indicadas essas relações, cuja indicação gráfica é feita como se mostra no esquema adiante.

D	1 - CARTÃO DE PONTO	
2729	DESCRIÇÃO VERBAL	Pi
D <sub>1-1</sub>	Número do período	7.
01-2	Rúmero de identificação do operário	P.
[D <sub>1-3</sub>	Número do dia da período	١,
P1-4]	Horas trabalhadas nossa dia	
D <sub>1-5</sub>	Linha	`

D	1 - BOLETIN DE ADNISSÃO	
ITEN	DESCRIÇÃO VERBAL	Pi
D <sub>2-1</sub>	Hünero do perlado	76
[P <sub>2-2</sub>	Munero de identificação do operário	Px
92-3	Nome do operário	7,
02-4]	Salarie-hore	۲,
D <sub>2-5</sub>	Linhe ·	

D	D <sub>3</sub> - FOLHA DE PAGAMENTO .		
ITEH	DESCRIÇÃO VERBAL	7,	
D <sub>3-1</sub>	Número do período	76	
[0,-2	Número de (dentificação do operário	P,	
D <sub>3-3</sub>	Nome de aperérie	7,	
D <sub>3-4</sub>	Salário-hore	7,	
b <sub>3-5</sub>	Salário do porfedo	1	
n <sub>3-6</sub> }	Salário acumulado	1	
03-7	Links	1	



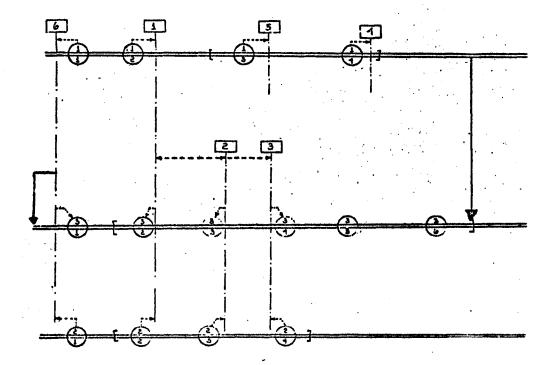
Vejamos quais as relações de produção existentes. O único documento produzido pelo sistema é a folha de pagamento. É produzida uma folha em cada período. Escrevemos então

$$P_6 \rightarrow D_3$$

Por outro lado, cada operário que pertence à empresa, teve, no período, um cartão de ponto correspondente. Assim, cada car tão de ponto, produz uma linha da folha de pagamento, isto é,

$$D_1 \rightarrow D_{3-7}$$

Graficamente essas relações são expressas como mostra o esquema a seguir.



Vemos, nos quadros da página anterior que os itens D<sub>3-5</sub> e D<sub>3-6</sub> da folha de pagamento, não têm relação direta com nenhum dos conjuntos de informação. Vamos analisar minuciosamente cada um deles.

Quanto a  $D_{3-5}$ , que é o salário do operário correspondente ao período, é obtido a partir de valores pertencentes a  $P_4$  e  $P_3$ , is to é, horas trabalhadas e salário-hora.

Quanto ao salário-hora, trata-se de uma mera transcrição, ou da folha de pagamento correspondente ao período anterior, ou do boletim de admissão (se for operário novo).

Ja quanto ao total de horas trabalhadas, é necessario explicitar os critérios adotados, para que o sistema efetue os calculos. Voltemos ao enunciado do problema, para analisar esses critérios. Podemos observar que existem condições envolvidas.

A primeira é referente à jornada diária de trabalho de 8 horas. Se o operário trabalha mais que 8 horas, apenas 8 são consideradas como trabalho normal, as demais contam como trabalho extra ordinário. Vamos então denominar C<sub>3-1</sub> a condição respectiva, isto é,

A segunda condição é referente à jornada semanal de trabalho de 40 horas normais. Se o operário trabalha mais que 40 horas

normais, significa que também faz jus ao pagamento de extraordinário. Na expressão dessa segunda condição, vamos utilizar a expressão de C<sub>3-1</sub>, pois o cálculo do número de horas normais trabalhadas é feito considerando que, se o operário trabalhou mais que 8 horas em um dia, apenas 8 são de trabalho normal.

Considerando que  $C_{3-1}$  pode ser Verdadeira ou Falsa, podemos associá-la a uma variável booleana que assume respectivamente os valores 1 e 0.  $\bar{C}_{3-1}$  que  $\bar{e}$  a condição complementar de  $C_{3-1}$  teria em correspondência, os valores 0 e 1. Assim, podemos exprimir a quantidade de horas normais trabalhadas em um dia, como:

$$c_{3-1} \cdot D_{1-4} + \overline{c}_{3-1} \cdot 8$$

Vejamos como isso funciona. Se o operário trabalhou 5 horas  $C_{3-1}$  tem valor 1 e  $\overline{C}_{3-1}$  tem valor 0; a expressão dá como resultado 5.1+0.8=5, isto é, 5 horas normais. Se trabalhou 8 horas temos 8.0+1.8=8 horas normais. Se,12 horas, temos 12.0+1.8=8 horas normais.

Podemos, pois escrever, a expressão de  $C_{3-2}$ , que  $\tilde{e}$ , ter o operario trabalhado em um período, mais que 40 horas normais:

$$C_{3-2}:\Sigma(C_{3-1}.D_{1-4}+\overline{C}_{3-1}.8)>40$$

Como uma hora extra equivale a 1.5 horas normais, escreve mos a quantidade equivalente a horas normais relativa ao trabalho de um dia, como sendo:

$$C_{3-1} \cdot D_{1-4} + \overline{C}_{3-1} \cdot (D_{1-4} * 1.5-4)$$

No período todo, devemos fazer a soma dessas quantidades referentes a todos os dias do período. Falta, para obter o total de horas, acrescentar a essa soma a parcela referente ao extraordinário a que faz jus o operário que, eventualmente tenha trabalhado, no período mais que 40 horas normais, isto é.

$$c_{3-2}$$
.(( $\Sigma(c_{3-1},D_{1-4}+\bar{c}_{3-1},8)$ )\*.5-20)

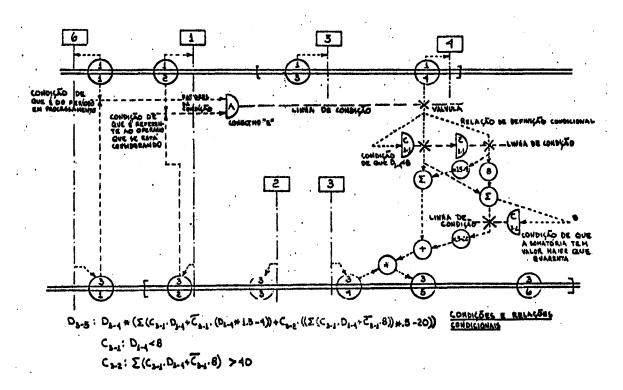
Podemos, finalmente, escrever a expressão do total de horas relativo ao período, como sendo:

$$\Sigma(c_{3-1}, D_{1-4}+\bar{c}_{3-1}, (D_{1-4}*1.5-4))+c_{3-2}, ((\Sigma(c_{3-1}, D_{1-4}+\bar{c}_{3-1}, 8))*.5-20)$$

A expressão correspondente à relação de definição de D<sub>3-5</sub>

 $\tilde{\mathbf{a}}$  simplesmente o produto desse total pelo salário-hora, ou seja, por  $\mathbf{D}_{\mathbf{3-4}}$ .

Graficamente, a relação de definição de D<sub>3-5</sub>, que é, como vimos, uma relação condicional, é esquematizada como se segue:



Cuidemos, agora, de D<sub>3-6</sub>, isto é, do salário acumulado. Se o operário constava da folha de pagamento referente ao período anterior, basta acrescentar, ao salário acumulado que lá figurava; o do período em processamento. Caso contrário, o salário acumulado é igual ao do período, pois o operário é recem admitido. A condição de que o operário constava da folha anterior pode ser escrita:

$$c_{3-3}: 3p_{3-7}(p_{3-1}-1)|p_{3-2}=p_{1-2}$$

Então podemos escrever a relação de definição de D3-6;

$$D_{3-5}+C_{3-3}\cdot(D_{3-6}(D_{3-2},D_{3-1}-1))$$

Graficamente, essa relação é figurada como se pode obser-

var no esquema da pg. 86.

Resta, apenas, para completar a solução do problema, exprimir as exigências operacionais.

Pelo enunciado, vemos que cada operário tem um cartão de ponto em cada período, no qual, em cada linha, são registradas as horas correspondentes a cada dia do período. Assim, podemos escrever as expressões quanto ao volume de informações correspondente ao cartão de ponto:

$$E_{1/P_{6}}=1200$$

$$D_{1-5}/D_{1}=7$$

Na pagina seguinte, constam dos quadros as expressões relativas a volume, referente ao Boletim de Admissão e a Folha de pagamento, determinadas de maneira semelhante.

Quanto a tempo, a unica exigência operacional é, pelo enunciado, que a folha de pagamento seja produzida dentro de três dias, a partir do encerramento do período a que corresponde. Pode mos, então, dizer que a diferença entre o tempo extrínseco relativo à folha de pagamento, isto é, a data em que é produzida e o tempo intrínseco, isto é, a data do ultimo dia do período, deve ser menor ou igual a três dias, e escrevemos

$$t_E(D_3)-t_I(D_3)=3$$
 dias

Na página seguinte, apresentamos a solução completa do problema, pelo método gráfico, descrito no capítulo III.

B <sub>2</sub> - CARTÃO DE PONTO			ENTRADA		
ITEM	DESCRIÇÃO VEPRAL	7,	PFLAÇÃO DE DEFINIÇÃO		
P1-1	Número da pariado	Pa			
71-2	Hümero de Identificação do operário	P <sub>1</sub>			
(*,-,	Rúnoro do dio da pariodo	١,			
*1-4]	Horas trubalhados massa dis	124			
	Linba		31-3,4		

	2 - BOLETIM DE ADMISSÃO		ENTRADA
TTEN	DESCRIÇÃO VERBAL	PE	RELAÇÃO DE DEFERÇÃO
D <sub>2-1</sub>	Húnero de Pertodo	7.	
[P2-7	Mûnero de identificação do operácio	71	
22-3	Nome de operârio		İ
02-4]	Salāria-hota	7,	
11-5	Links	1	02-2.6

D	3 - FOLHA DE PAGAMENTO	SATDA					
ITEM	DESCRIÇÃO VERBAL	Pi	RELAÇÃO DE DEFINIÇÃO				
D <sub>3-1</sub>	Número do período	P					
[D <sub>3-2</sub>	Número de identificação do operário	P <sub>1</sub>					
D <sub>3-3</sub>	Nome do operário	P <sub>2</sub>					
D <sub>3-4</sub>	Salário-hora	P3					
D <sub>3-5</sub>	Salario do període		$\begin{array}{c} D_{3-4} * (\Sigma(C_{3-1}, D_{1-4} * \overline{C}_{3-1}, (D_{1-4} * 1.5-4)) + \\ + C_{3-2} \cdot ((\Sigma(C_{3-1}, D_{1-4} * \overline{C}_{3-1}, 8)) * \cdot 5-20)) \end{array}$				
D <sub>3-6</sub> ]	Salário acumulado		D <sub>3-5</sub> +C <sub>3-3</sub> .D <sub>3-6</sub> (D <sub>3-2</sub> ,D <sub>3-1</sub> -1)				
D <sub>3-7</sub>	Linha		D <sub>3-2,6</sub>				

Relações de Produção:

P<sub>6</sub> + D<sub>3</sub>

D<sub>1</sub> + D<sub>3-7</sub>

Condições especiais:

 $C_{3-1}:D_{1-4} \in \mathbb{S}$  O operário não fez extraordinário no dia.

 $C_{3-2}:\Sigma(C_{3-1},D_{1-4}+\overline{C}_{3-1},8)>40$  O operário fez extraordinário no fim da semana.

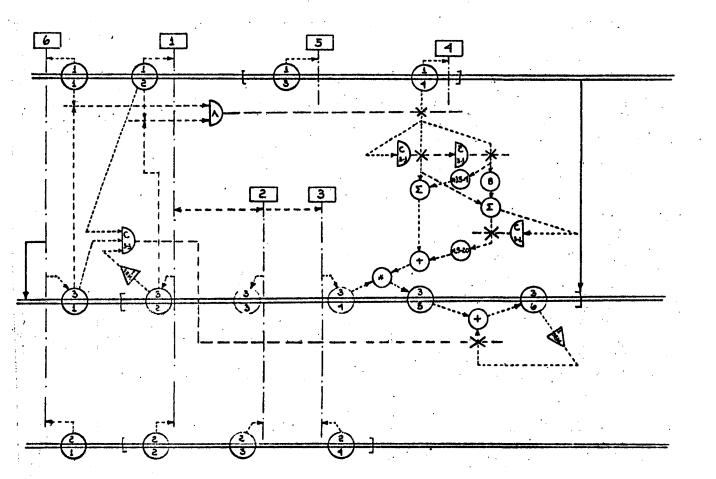
 $c_{3-3}: ^{3}D_{3-7}(D_{3-1}-1)|_{D_{3-2}-D_{1-2}}$  O operario estava incluído na folha de pagamento anterior.

Exigências operacionais:

Volume: [ D3/P6=1 & emitida uma folha em cada período.

Î D<sub>3-7</sub>/D<sub>3</sub>-1200 Em média constam de cada folha 1200 operários

Tempo:  $t_{\mathbf{g}}(D_3) - t_{\mathbf{I}}(D_3) \leq 3$  dias



# 3 - UM PROBLEMA DE FATURAMENTO, CONTROLE DE VENDAS E DE COBRANÇA

I Thinks

#### 3.1- ENUNCIADO

Uma industria mantém estoque em varios armazens. A cada um deles estão ligados diversos viajantes, cada um dos quais atende a varios clientes.

As vendas são efetuadas e comunicadas aos armazens, que providenciam a remessa das mercadorias e, ao mesmo tempo, enviam in formações ao escritório central, para efeito de faturamento.

O escritório central providencia a emissão e remessa das faturas aos clientes, recebe pagamentos, fornece extratos de conta mensal e prepara um relatório diário de vendas acumuladas, para informação gerencial.

Cada cliente é atendido por um determinado viajante, que, por sua vez, trabalha ligado a certo armazém. Cada cliente pode possuir mais de um depósito. A cada depósito vai corresponder certo endereço de remessa.

Cada peça é oferecida em uma ou mais cores. Uma peça de determinada cor constitui um modelo. A lista de preços é organizada de modo que, a cada modelo, corresponda um preço unitário para cada área de preço.

Cada armazém pertence a uma determinada área de preço.Uma área de preço pode abranger mais de um armazém.

Existe, em funcionamento, um sistema de numeração e codificação.

Cada cliente tem um número de identificação constituído de cinco algarismos. A cada um de seus depósitos está associado um código de remessa constituído de um algarismo, de modo que, dado certo número de identificação de cliente e certo código de remessa, se saiba para qual endereço se deve fazer a remessa.

Cada viajante tem um número de dois algarismos que o ide $\underline{\mathbf{n}}$  tifica.

Cada peça á identificada por um código, constituído de três caracteres alfa-numéricos. A cada cor está associado um código de dois caracteres alfabéticos. Assim, cada modelo é identifica do por um código de cinco caracteres alfa-numéricos.

Cada área de preço é identificada por um código constitu $\underline{\tilde{I}}$ 

do de um caracter alfabético.

As quantidades de clientes, viajantes, armazens, modelos, etc. estão indicadas, no quadro da pg. 89. Nele constam também indicações quanto ao tamanho máximo dos nomes de viajantes, nome e endereço de clientes, nomes de armazens, etc.

Ha cinco documentos nesse sistema: dois de entrada (nota de remessa e nota de pagamento) e três de saída (fatura, relatório diário de vendas acumuladas e extrato de conta mensal).

DESCRIÇÃO VERBAL
••
mero de nota de Remessa
mero de identificação do cliente digo de Temessa
mero do viajante antidade encomendada
mero do modelo

D	2 - FATURA
ITEM	DESCRIÇÃO VERBAL
D <sub>2-1</sub>	Data
D <sub>2-2</sub>	Número da Fatura
D <sub>2-3</sub>	Número de identificação do cliente
D <sub>2+4</sub>	Nome e endereço do cliente
D <sub>2-5</sub>	Endereço de remessa
D <sub>2-6</sub>	Armasem remeteste
D <sub>2-7</sub>	Quantidade encomendada
P <sub>2-8</sub>	Numero do modelo
D <sub>2-9</sub>	Preço unitário
P <sub>2-10</sub> ]	Preço total
D <sub>2-11</sub>	Total de fatura
D <sub>2-12</sub>	Linha

D <sub>3</sub> - NOTA DE PAGAMENTO								
ITEM	DESCRIÇÃO VERBAL							
D <sub>3-1</sub>	Date							
[D <sub>3-2</sub> ]	Número da fatura Importância							
D <sub>3-4</sub>	Limba							

D <sub>A</sub> - EXTRATO DE CONTA MENSAL								
ITEM	DESCRIÇÃO VERBAL							
P <sub>4-1</sub>	None e endereço do cliento							
D <sub>4-2</sub>	Data							
D <sub>4-3</sub>	Seldo enterior							
[P <sub>4-4</sub>	Número da fatura							
D4-5	Data da fatura							
P4-6]	Importância da fatura							
D <sub>4-7</sub>	Linha							
P4-8	Kovo Saldo							

D <sub>5</sub> - RELATÓRIO DIÁRIO DE VENDAS ACUMULADAS								
TTEM	DESCRIÇÃO VERBAL							
D <sub>5-1</sub>	Data							
[3 <sub>5-2</sub>	Número do Viajante							
<sup>3</sup> 5-3	Nome do Visjante							
D <sub>5-4</sub>	Vendes nesta Data							
D <sub>5-5</sub> ]	Vendas acumuladas meste mês							
35-6	Linha							
25-7	Total de vendas mesta Data							

Trezentas notas de remessa são recebidas, diariamente, pe lo escritório central e cada uma tem, em média, cinco linhas.

É preparada uma fatura correspondente a cada nota de remessa. Após a determinação do preço unitário, o preço total de cada item e o total da fatura são calculados. A remessa da fatura de ve ser feita, dentro de dois dias, a contar do recebimento da nota de remessa.

CARACTERÍSTICAS DAS INFORMAÇÕES	IDADE	HO DO
INFORMAÇÃO	QUANTIDADE	TAMANHO CAMPO
Data	•	6 N
Número de identificação do cliente	2000	5N
Codigo de remessa	9	1 N
Numero do viajante	50	2 N
Número do modelo	1500	5A/N
Quantidade encomendada		2 N
Dia	31	2 N
Mês	12	2A
Ano	10	2 N
Nome e endereço do cliente	2000	50/N
Nome do armazém	10	12A/N
Número da peça	800	3A/N
Cor	20	2A
Endereço de remessa	6000	50A/N
Area de preço	8	1A
Nº da fatura (Nº da nota de remessa)		5 N
Preço unitário	12000	5 N
Nome do viajante	50	15A

Cerca de duzentos pagamentos são recebidos, diariamente , cada um cobrindo, em média,  $1.5~{\rm faturas}$ . O cliente menciona o  $n\bar{\rm u}$ -mero das faturas, em cada pagamento

Os extratos de conta mensal são enviados entre os dias 10 e 15 de cada mês, a todos os clientes que tenham saldo em aberto no dia 10 desse mês. As faturas de data posterior so dia 10 do mês anterior, que não tenham sido pagas, são relacionadas no extrato. Faturas anteriores, não pagas, são englobadas e constam apenas no montante do saldo anterior. Em média, são emitidos quinhentos extratos por mês e são relacionadas, em média, quatro faturas por extrato.

O relatório diário de vendas acumuladas é produzido dentro de dois dias, após o encerramento do movimento, do dia a que se refere. Inclui o total de vendas na data e uma relação, por via jante, do total de vendas por ele efetuadas na data e acumuladas desde o início do mês.

## 3.2- SOLUÇÃO ALGÉBRICA

Os problemas de faturamento, controle de vendas e de cobrança podem ser encarados como constituídos de dois casos separados de processamento de dados:

Problema 1 - Processamento diário, destinado a produzir o facuramento das vendas do dia anterior e o Relatório Diário de vendas acumuladas até o dia anterior.

Problema 2 - Processamento mensal, destinado a produzir o Extrato de Conta Mensal.

Vamos, inicialmente, expressar o Problema 1, através da Álgebra da Informação.

Nele os arquivos de entrada são:

- A<sub>1</sub> Cadastro de Clientes, no qual cada registro reune to das as características relativas a um depósito de cliente, quais sejam código e endereço de remessa correspondente ao depósito, número de identificação, nome e endereço do cliente, número e nome do viajante que o atende, nome do armazém que o serve e área de preço, em que está situado.
- A Cadastro de Preços, em que cada registro corresponde ao preço unitário de determinado modelo de peça, isto é, uma peça com certo número e determinada cor, relativo a uma área de preço.

A3 - Mercadoria Remetida, onde cada registro corresponde a uma linha da Nota de Remessa datada do dia anterior. Nesse regis tro constam, além das informações que constituem a linha, todas as demais informações relativas à Nota de Remessa, à qual a linha per tence. Esse arquivo deve incluir todas as Notas de Remessa, datadas do dia anterior, e apenas essas notas.

Através de uma série de transformações que serão expressas pela Álgebra da Informação, deve-se obter os seguintes arquivos de saída:

A<sub>4</sub> - Faturamento, com finalidade de, a partir dele, imprimir-se as faturas a serem enviadas aos clientes.

 $A_{600}$  - Resumo do faturamento para preparação do Extrato Mensal.

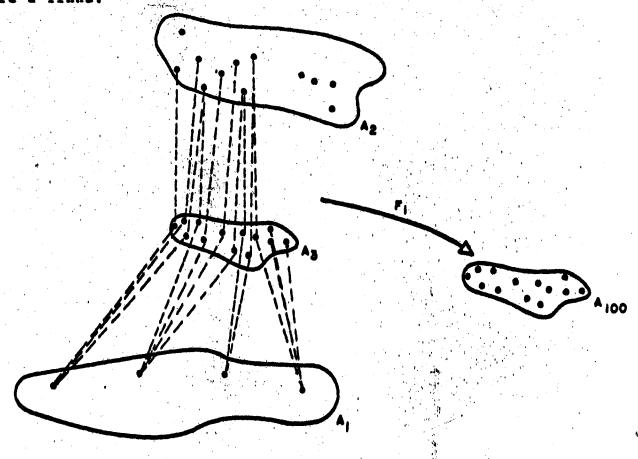
A, - Novo Relatório Diário de Vendas Acumuladas.

Na tabela da página 92 apresentamos o conjunto de propriedades relativas ao problema, que vai constituir o conjunto de coordenadas do Espaço de Propriedades em que vamos estudá-lo. Nessa tabela, figuram as áreas correspondentes a cada arquivo, com indicação da relevância ou não de cada propriedade, para cada área.

COMJUNTO DE COORDENADAS DO ESP				<b>^</b> 1	A <sub>1</sub> A <sub>2</sub> A <sub>3</sub>		A	A 5	A6	A, A,00		A <sub>20</sub>	
DE PROPRIEDADES UTILIZADO				CONJUNTO DE VALORES	CABASTRO DE CLIEBTES	CADASTRA BE PREÇOS	MERCABORIA REMETIDA	PATURAMENTO	PELATÓRIO DIÁRIO VENDAS ACUN.	KYTRATO DE COSTA HENSAL A	PACAMENTOS	RESUNO PATURAMENTO	RESUNO EXTRATO DE CONTA
41	Data			000000 311299	n n	o o	*	*	N X	×	*		i i
۹2	Número de	identificação	do cliente	00000	×	្ន	×	*	n	G	រា	×	×
۹,	Codigo de	Temesta _		0	×	Ω	×	Ω	Ω	ñ	Ü	Ω	Ω
94	Número do	viajante		0099	×	·Ω	×	Ω	*	ภ	Ω	·ß	Ω
۹5	Número do	medelo		5 caracteres alfa-numéricos	ធ	×	×	я	Ω	Ω	Ω	Ů	Ω
96	Quantidad	e enconeudade		0099	·υ	ຄ	×	*	O.	Ω	n	Ω	Ω
97	Dia			0031	Ω	U	×	Ω	Ω	×	Ω	Ω	×
ч <sub>в</sub>	Hēs	•		0012	Ω	Ω	×	Ω	ຄ	×	Ω	Ω	×
9	Ano .			0099	Ω	Ω	×	Ω	Ω	×	Ω	Ω	×
<sup>q</sup> 10	Nome a en	dereço do elie:	ate .	30 caracteres alfa-numéricos	×	Ω	Ω	×	s .	*	Ω	ິດ	×
911	Nome do d	rmasém	•	12 caracteres alfa-muméricos	×	Ω	ũ	*	a	Ω	Ω	Ω	្ត្
912	Númeto de	Paga -		3 caracteres alfa-numéricos	ū	ж.	U	U	ß	Ω	រា	Ω	3
q <sub>13</sub>	Cor			2 carecteres alfabáticos	Ω	·×	Ω	n	Ω	מ	Ω	n	is ,
q <sub>14</sub>	Endereço	de remessa		50 caracteres alfa-numéricos	×	ល	Ω	*.	ល	Ω	Ω	Ω	n i
<sup>q</sup> 15	Āres de p	reço		l caracter alfabético	×	×	Ω	ព	Ω	Ω	Ω	Ω.	£,
<sup>q</sup> 16	NY da Tat	ura (NY da mot	a de remessa)	00000	Ω	ព	¥	×	Ω	×	¥	*	Ω
417	Praço uni	tário		000.00	Ω	×	Ω	×	Ω	Ω	Ω	Ω	Ω
9 <sub>18</sub>	Nome do v	risjante		15 caracteres alfabéticos	×	Ω	Ω	a	*	Ω	Ω	Ω	Ω
q <sub>19</sub>	Totalized	or 1		00000000.00	Ω	ŭ	Ω	я	×	Ħ	Ω	Ω	*
q <sub>20</sub> Totalisador 2			00000000.00	Ω	Ω	Ω	×	*	¥.	*	×	, *	
921	Totalizad	or 3		00000000.00	ດ	រ	Ω	Ω	×	*	Ω	ũ	
UTILIZAÇÃO DOS A		A <sub>5</sub>	^6	A7		A700			A <sub>20</sub>		<u></u>		
Tota	líz, i	Preço Total	Vendas do via jante mesta data	- Importância da Fature				,	•		0		
Tota	1iz. 2	Total de Paturs	Vendas acumul das do viajan neste mês			ortân Fetu			al da tura			o Sale	
Totaliz. 3 - Total de ven						-			4			o Sale	

## 3.2.1- EXPRESSÕES CORRESPONDENTES AO PROBLEMA 1

Os dedos necessários à preparação do faturamento estão se paradom em três áreas distintas, isto é, alguns elementos constam do Cadatro de Clientes, outros do de Preços e outros do Arquivo de Mercadoria Remetida. Para reuni-los precisamos estabelecer um feixe no conjunto de áreas {A1,A3,A2} em que a função que determina o feixe é constituída pela seleção que permite contruir cada linha do feixe, ligando os elementos constantes de uma linha da Nota de Remessa, às características do cliente a que se refere a Nota e so Preço unitário, correspondente ao Modelo de Peça a que se refere a linha.



A<sub>100</sub> = Y<sub>1</sub>(q<sub>1:2</sub>-q<sub>2:2</sub> A<sub>1:3</sub>-q<sub>2:3</sub> A<sub>2:5</sub>-q<sub>3:5</sub> A<sub>q<sub>1:15</sub>-q<sub>3:15</sub> A<sub>1:A<sub>3</sub></sub> A<sub>2</sub>)

Y<sub>1</sub> = (q<sub>1</sub>-q<sub>2:1</sub>, q<sub>2</sub>-q<sub>2:2</sub>, q<sub>4</sub>-q<sub>1:4</sub>, q<sub>6</sub>-q<sub>2:6</sub>, q<sub>7</sub>-q<sub>2:7</sub>, q<sub>10</sub>-q<sub>1:10</sub>,

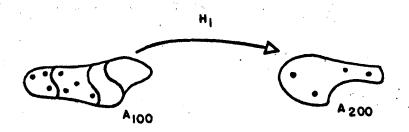
q<sub>1</sub>-q<sub>1:11</sub>, q<sub>12</sub>-Ω, q<sub>13</sub>-Ω, q<sub>14</sub>-q<sub>1:14</sub>, q<sub>15</sub>-Ω, q<sub>16</sub>-q<sub>2:16</sub>,

q<sub>19</sub>-q<sub>2:6</sub>\*q<sub>3:17</sub>)</sub>

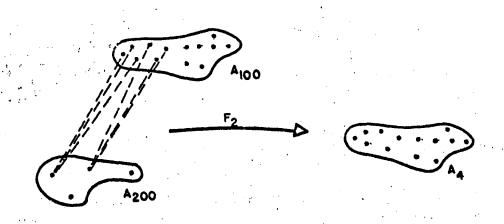
Pela função  $F_1$  desse feixe reunimos na area  $A_{100}$  todas as informações necessárias tanto ao faturamento quanto ao relatório de vendas. Cada ponto, nessa area, corresponde a um ponto de  $A_3$  enriquecido, todavia, com as informações colhidas de  $A_1$  e  $A_2$ .

Calculemos, agora a importância de cada fatura. Para isso estabelecemos em A<sub>100</sub> a malha determinada pela função q<sub>16</sub>, isto é, na qual cada elemento reune os pontos correspondentes às linhas de uma Nota de Remessa ou seja, que produzem uma fatura.

Pela FOG H, obtemos a importância desejada.



 $A_{200} = H_1(q_{16}, A_{100})$   $H_1 = (q_1' - q_1, q_4' - q_4, q_7' - q_7, q_{16}' - q_{16}, q_{20}'' - Eq_{19})$ 



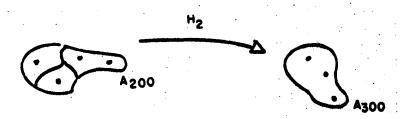
A4 = 72 (41:16 42:16 + A200 + A100) 72 = (44 91.420 41:20)

Pela FOB F<sub>2</sub> incorporamos a importância de cada fatura aos pontos a ela correspondentes de  $A_{100}$  e obtemos, assim, a área  $A_4$  que corresponde ao faturamento completo.

Procuremos o total de vendas de cada viajante, nesse dia. Para isso, estabelecemos em  $A_{200}$  a malha determinada pela função

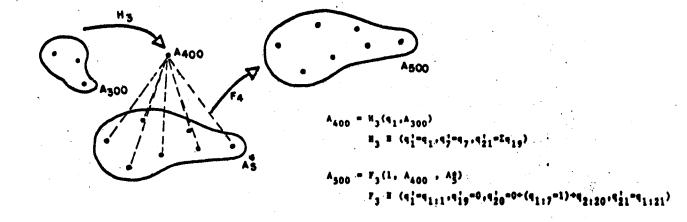
q<sub>4</sub>, isto, e na qual cada elemento reune as faturas relativas a determinado viajante.

Pela FOG H<sub>2</sub>, obtemos a importância desejada. Cada ponto de A<sub>300</sub> representa, assim, um viajante que teve mercadorias por ele vendidas, remetidas naquele dia. No totalizador 1 temos o valor total dessas mercadorias.

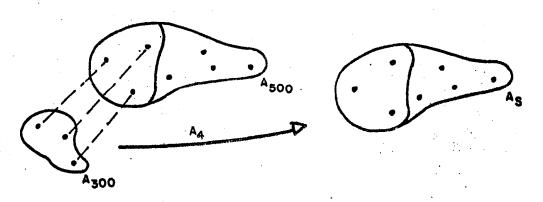


A<sub>300</sub> = H<sub>2</sub>(q<sub>4</sub> , A<sub>200</sub>) H<sub>2</sub> = (q;-q<sub>1</sub>,-q;-q<sub>2</sub>,-q;-Eq<sub>20</sub>)

Falta apenas calcular o total de vendas na data do Relatorio. Para isso, estabelecemos em  $A_{300}$  a malha determinada por  $q_1$ , isto é, que será constituída por um só elemento, pois todos os pontos tem mesmo valor para  $q_1$  e pela FOG  $H_3$  calculamos o valor desejado, obtendo a área  $A_{400}$  constituída de um só ponto. Depois, no conjunto de áreas  $\{A_{400},A_5^*\}$  construimos pela seleção 1 um feixe que liga esse ponto a todos os pontos de  $A_5^*$  e pela FOB  $F_3$  incorporamos o total de vendas, na data, a todos os pontos de  $A_5^*$ , bem como a data do novo relatório e, conforme seja início de mês, ou não, zera-se o total acumulado de vendas de cada viajante ou transporta-se esse total para cada ponto de  $A_{500}$ .



Com as providências anteriormente citadas, preparamos  $_{\rm A}$   $_{\rm a}$  rea  $_{\rm A_{500}}$  para constituir o novo Relatório de Vendas, pois inclusive zeramos o total de vendas, na data, para todos os viajantes, a fim de prever o caso dos viajantes que não fizeram vendas nesse dia. Agora, no conjunto de áreas  $_{\rm A_{300},A_{500}}$  formamos o feixe determinado pela seleção  $_{\rm 1:4}$  através do qual vamos lançar o total de vendas dos viajantes que efetuaram vendas nesse dia e ajustar o total de vendas acumulado, utilizando, para isso, a FOB  $_{\rm A}$ .



$$A_5 = P_A(B_1)UT_2^{\dagger}(B_1)$$
 $B_1 = (q_{1:4}^{\dagger}q_{2:4}^{\dagger}, \Lambda_{300}^{\dagger}, \Lambda_{500}^{\dagger})$ 
 $P_A = (q_{1:9}^{\dagger}q_{1:19}^{\dagger}, q_{20}^{\dagger}q_{1:19}^{\dagger}q_{2:20}^{\dagger})$ 

A partir da malha determinada em A<sub>100</sub> por q<sub>16</sub>, podemos, a-través da FOG H<sub>4</sub> escrever a expressão do resumo do faturamento, que será usado como um dos dados necessários à solução do Problema 2, isto é, do processamento mensal.

$$A_{600} = H_4(q_{16}, A_{100})$$

$$H_4 = (q_1' = q_1, q_2' = q_2, q_{10}' = q_{10}, q_{16}'' = q_{16}, q_{16}'' = q_{19})$$

Resumindo, podemos escrever as expressões relativas ao Problema 1, isto é, ao processamento diário:

$$A_{5} = F_{4}(B_{1}) UI_{2}^{1}(B_{1})$$

$$F_{4} = (q_{19}^{1} = q_{1:19}, q_{20}^{1} = q_{1:19} + q_{2:20})$$

$$B_{1} = (q_{1:4} = q_{2:4}, A_{300}, F_{3}(1, H_{3}(q_{1}, A_{300}), A_{5}^{2}))$$

$$F_{3} = (q_{1}^{1} = q_{1:1}, q_{19}^{1} = 0, q_{20}^{1} = 0 + (q_{1:7} = 1) + q_{2:20}, q_{21}^{2} = q_{1:21})$$

$$H_{3} = (q_{1}^{1} = q_{1}, q_{7}^{1} = q_{7}, q_{21}^{1} = \mathbb{Z}q_{19})$$

$$A_{300} = H_{2}(q_{4}, H_{1}(q_{16}, A_{100}))$$

$$H_{2} = (q_{1}^{1} = q_{1}, q_{4}^{1} = q_{4}, q_{7}^{1} = q_{7}, q_{19}^{1} = \mathbb{Z}q_{20})$$

Cuidemos agora do Problema 2, isto 4, do processamento mensal.

Aqui, os arquivos de entrada são:

A700 - Resumo do Faturamento do Período. Essa área é obtida juntando-se todos os resumos de faturamento relativos ao período, isto é, cujas datas são posteriores ao dia 10 do mês anterior e vão até o dia 10 do mês corrente.

A\* - Resumo do Extrato de Conta Mensal anterior.

A<sub>7</sub> - Pagamentos Recebidos. Essa área é obtida juntando-se todas as Notas de Pagamento, relativas aos pagamentos recebidos no período.

Esses arquivos vão ser submetidos a transformações para que se obtenha os seguintes arquivos de saída:

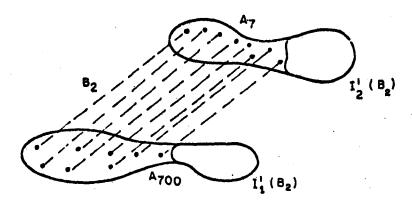
A6 - Extrato de Conta Mensal.

A20 - Resumo do novo Extrato de conta Mensal.

Na tabela da pg. 92 indicamos a relevância ou não das propriedades em cada uma dessas áreas.

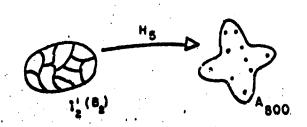
#### 3.2.2- EXPRESSÕES CORRESPONDENTES AO PROBLEMA 2

Pelo enunciado do problema, verificamos que no extrato de conta mensal não entram as faturas emitidas e pagas, durante o período. Vamos, através do feixe  $B_2$ , estabelecido no conjunto de árreas  $\{A_{700},A_7\}$  pela seleção  $q_{1:16}=q_{2:16}$  identificar essas faturas. Os pagamentos não abrangidos por esse feixe  $(I_2'(B_2))$  referem-se a faturas emitidas, no período anterior e não abrangidas  $(I_1'(B_2))$  são faturas não pagas, emitidas no período.



 $B_2 = (q_{1:16} - q_{2:16}, A_{700}, A_7)$ 

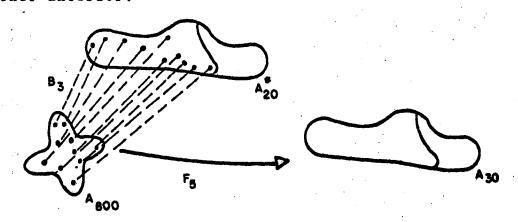
Vamos agora calcular o total de pagamentos relativos a faturas emitidas, em períodos anteriores, por cliente. Fazemos isso, através da FOG  $\rm H_5$  da malha estabelecida em  $\rm I_2^*(B_2)$  pela função  $\rm q_2$ .



 $A_{800} = H_5(q_2, I_2^+(B_2))$   $H_5 = (q_2^+ q_2^-, q_1^+ = Iq_{20}^-)$ 

Cada ponto de  $A_{800}$  refere-se a 1 cliente que fez, no período, pagamento de faturas emitidas, no período anterior. No tota lizador 1 de cada ponto consta o total desses pagamentos.

Vamos começar a preparar o novo Extrato. Para isso, calculemos o saldo anterior do novo Extrato, que é igual ao valor do totalizador 3 dos pontos que constam do Resumo do Extrato Anterior, do qual se diminui o valor do totalizador 1 dos pontos de  $A_{800}$ , para os clientes que fizeram pagamentos durante o período, relativos a faturamento anterior.



 $B_3 = (q_{1:2} - q_{2:2} + A_{800} + A_{20}^*)$ 

Fazemos isso através da FOB F $_5$  do feixe B $_3$  estabelecido no conjunto de áreas  $\{A_{800},A_{20}^{*}\}$ , pela seleção  $q_{1:2}=q_{2:2}$ .

Podemos, assim, escrever a expressão de  $\mathbb{A}_{30}$  que é uma primeira aproximação do resumo do novo Extrato.

$$A_{30} = F_5(B_3) UI_2'(B_3)$$

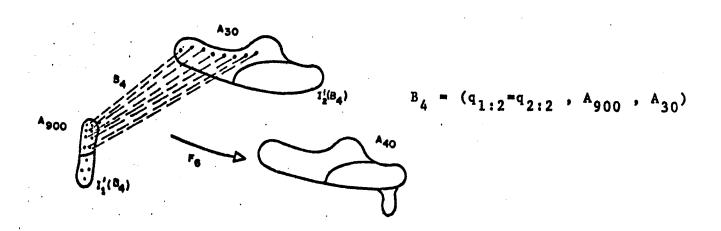
$$F_5 = (q'_{20} = q_{2:20} - q_{1:19}, q'_{21} = q_{2:20} - q_{1:19})$$

Vamos agora verificar, por cliente, o total relativo as faturas não pagas, emitidas no período. Isso é feito através da FOG  $\rm H_6$  da malha determinada em  $\rm I_1'(B_2)$  pela função  $\rm q_2$ .

Passemos a uma segunda aproximação do resumo do novo Extrato, calculando o novo saldo, para cada cliente. Teremos três ca

- a) Clientes que constavam do resumo do extrato anterior e que tiveram faturamento não pago, no período.
- b) Clientes que constavam do resumo do extrato anterior e que não tiveram faturamento não pago, no período.
- c) Clientes que não constavam do resumo do extrato anterior e que tiveram faturamento não pago, no período.

Com o feixe  $B_4$ , determinado no conjunto de áreas  $\{A_{900},A_{30}\}$  pela seleção  $q_{1:2}=q_{2:2}$  vamos delimitar os três casos, isto é, os pontos ligados pelo feixe representam (a), os pertencentes a  $I_2'(B_4)$  representam (b) e os pertencentes a  $I_1'(B_4)$  representam (c).

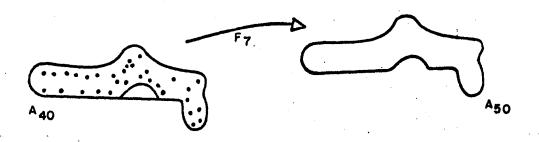


$$A_{40} = F_6(B_4) UI_1'(B_4) UI_2'(B_4)$$

$$F_6 = (q_{19}'' - q_{1:19}, q_{21}'' - q_{2:20}' + q_{1:19})$$

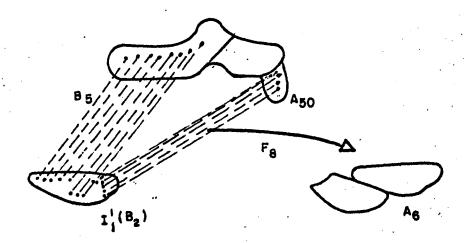
Nessa área A<sub>40</sub> há para cada cliente, um ponto em que constam: Número de identificação, nome e endereço, total de faturamento emitido no período e não pago, novo saldo e saldo anterior.

Agora, no conjunto de áreas  $\{A_{40}\}$  determinemos um feixe pela seleção  $q_{1:21}\neq 0$  e pela FOB F<sub>7</sub> formemos a área  $A_{50}$  que reune os clientes com novo saldo não nulo.



$$A_{50} = F_7(q_{1:21} \neq 0, A_{40})$$
 $F_7 = ()$ 

Finalmente, transportemos os elementos do resumo para as faturas que devem constituir as linhas do novo Extrato de conta mensal. Fazemos isso, através da FOB  $F_R$ , do feixe  $B_5$ .



$$B_{5} = (q_{1:2} q_{2:2}, I_{1}^{1}(B_{2}), A_{50})$$

$$A_{6} = F_{8}(B_{5})UI_{2}^{1}(B_{5})$$

$$F_{8} = (q_{1}^{1} q_{1:1}, q_{16}^{1} q_{1:16}, q_{19}^{1} q_{1:20})$$

A partir de  ${\rm A}_{50}$  podemos produzir o resumo do novo Extrato que vai ser utilizado, no próximo processamento mensal, pela FOB  ${\rm F}_9$ :

$$A_{20} = F_9^{(1,A_{50})}$$

$$F_9 = (q_1' - \Omega, q_{19}'' - 0, q_{20}'' - q_{1:21}, q_{21}'' - q_{1:21})$$

Resumindo, podemos escrever as expressões relativas ao Problema 2, isto é, ao processamento mensal.

$$A_{6} = F_{8}(B_{3})UI_{2}^{1}(B_{5})$$

$$F_{8} \equiv (q_{1}^{1} = q_{1:1}, q_{16}^{1} = q_{1:16}, q_{19}^{1} = q_{1:20})$$

$$B_{5} = (q_{1:16}^{1} = q_{2:16}, A_{50})$$

$$B_{2} = (q_{1:16}^{1} = q_{2:16}, A_{700}, A_{7})$$

$$A_{50} = F_{7}(q_{1:21} \neq 0, A_{40})$$

$$F_{7} \equiv ()$$

$$A_{40} = F_{6}(B_{4})UI_{1}^{1}(B_{4})UI_{2}^{1}(B_{4})$$

$$F_{6} \equiv (q_{19}^{1} = q_{1:19}, q_{21}^{1} = q_{2:20}^{1} + q_{1:19})$$

$$B_{4} = (q_{1:2}^{1} = q_{2:2}, A_{900}, A_{30})$$

$$A_{900} = H_{6}(q_{2}, I_{1}^{1}(B_{2}))$$

$$H_{6} \equiv (q_{2}^{1} = q_{2}, q_{10}^{1} = q_{10}, q_{19}^{1} = Eq_{20}, q_{21}^{1} = Eq_{20})$$

$$A_{30} = F_{5}(B_{3})UI_{2}^{1}(B_{3})$$

$$F_{5} \equiv (q_{10}^{1} = q_{2:20}^{1} - q_{1:19}, q_{21}^{1} = q_{2:20}^{1} - q_{1:19})$$

$$B_{3} = (q_{1:2}^{1} = q_{2:2}, A_{800}, A_{20}^{2})$$

$$A_{800} = H_{5}(q_{2}, I_{2}^{1}(B_{2}))$$

$$H_{5} \equiv (q_{2}^{1} = q_{2}, q_{19}^{1} = Eq_{20})$$

$$A_{20} = F_{9}(1, A_{50})$$

$$F_{9} \equiv (q_{1}^{1} = 0, q_{19}^{1} = 0, q_{20}^{1} = q_{1:21}, q_{21}^{1} = q_{1:21})$$

Na solução apresentada tanto para o problema 1 como para o problema 2 foram adotadas as seguintes premissas:

- 1- A inclusão, no sistema, de novos clientes, novos modelos de peças ou novos viajantes, bem como alterações em suas carac terísticas, são feitas em um processamento separado.
- 2- Não são indicados pelo sistema erros tais como "A mercadoria remetida não consta do Cadastro de Preços" ou "O cliente a tendido não figura no Cadastro de Clientes" ou ainda "Viajante não consta no Relatório anterior", o que pressupõe uma verificação pré

via, em separado, para que tais erros não venham a ocorrer.

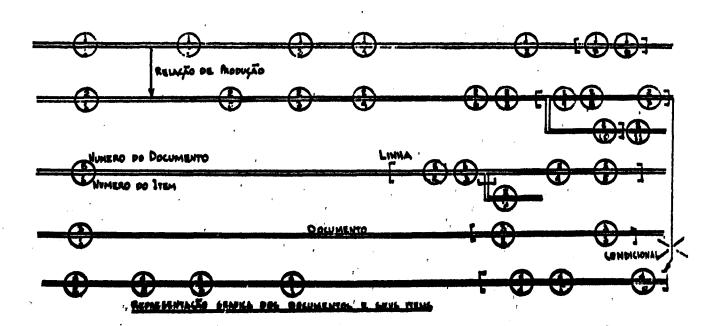
3- A data colocada nos documentos produzidos pelo sistema é introduzida, independente das relações apresentadas, com exceção da data das faturas e do Relatório diário de vendas, que é a mesma das Notas de Remessa correspondentes. No Extrato de Conta Mensal,  $q_1$  é reservada para a data das faturas emitidas no período e não pagas, colocando-se a data do Extrato, em  $q_7,q_8,q_9$ .

## 3.3- SOLUÇÃO GRÁFICA

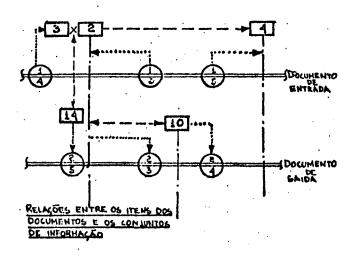
Inicialmente, identifiquemos os conjuntos de informação en volvidos no problema, bem como as relações existentes entre eles.

Atraves das informações constantes do quadro da pg. 89 podemos construir o quadro da pg. 105 onde estão perfeitamente identificados os conjuntos de informação e as relações que entre eles existem. Na parte inferior da mesma página, consta a representação gráfica correspondente.

Agora, em relação a cada documento que constitui o sistema, identifiquemos seus itens e verifiquemos a que conjunto de informação cada um pertence. Com as informações dos quadros da pg.88 construimos os quadros da pg.106. Podemos assim efetuar a representação gráfica dos documentos e seus itens, de acordo com o esquema abaixo:



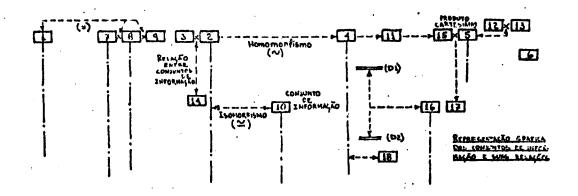
Quanto à representação gráfica das relações entre os itens dos documentos e os conjuntos de informação vejamos, por exemplo , como esquematizar as relações sobre os itens 3, 4 e 5 da Nota de Remessa e os conjuntos  $P_2$ ,  $P_3$  e  $P_4$ . Como se trata de um documento de entrada, as setas indicando essas relações vão do item do documento para o conjunto correspondente.



Quanto ao esquema das relações entre os itens 3, 4 e 5 da fatura e os conjuntos  $P_2$ ,  $P_{10}$  e  $P_{14}$ , como a fatura é documento de saída, as setas vão do conjunto para o item do documento.

Usando essa mesma sistemática, desenhamos, no esquema constante da pg.105. as linhas ligando cada item de documento ao conjunto de informação correspondente. Quanto à Nota de Remessa, isso é conseguido sem maiores problemas, mas quanto à Nota de Pagamento, D<sub>3-3</sub> não pertence a nenhum dos conjuntos de informação, pois é uma informação obtida das outras, quando do processamento da fatura correspondente. Pelos quadros da pg.106 pode-se observar que em situação análoga a essa, estão D<sub>2-10</sub>, D<sub>2-11</sub>, D<sub>4-3</sub>, D<sub>4-6</sub>, D<sub>4-8</sub>, D<sub>5-4</sub>, D<sub>5-5</sub>, e D<sub>5-7</sub>.

Pi	NOME DO CONJUNTO	N de N9 de elemen- tos	L Nº de carac- teres	RELAÇÕES
P	Data .		6N	P1=P7×P8×P9
P 2	Número de identificação do Cliente	2000	5ห	P2*P10,P2~P4
P 3	Codigo de remessa	9	111	P14*P2*P3
P <sub>4</sub>	Número do Viajante	50	· 2N	P2~P4~P11
P <sub>5</sub>	Número do Modelo	1500	5A/N	P5*F12*F13
P <sub>6</sub>	Quantidade encomendada		2 N	, ·
P <sub>7</sub>	Dia	31	2 N	P1=F7×P8×P9
P <sub>8</sub>	Mês	12	2 A	P1=P7×P8×P9
P <sub>9</sub>	Ano	10	2 N	P1=P7×P8×P9
P <sub>10</sub>	Nome e endereço do Cliente	2000	50A/N	P10*P2
P <sub>11</sub>	Nome do Armazém	10	12A/N	P4~P11~P15
P <sub>12</sub>	Número da Peça	800	3A/N	P5*P12*P13
P <sub>13</sub>	Cor	20	· 2A	P5*P12*P13
P <sub>14</sub>	Endereço de remessa	6000	SOA/N	P14"P2"P3
P <sub>15</sub>	Área de preço	8	1.6	P5 <sup>×P</sup> 15 <sup>~P</sup> 17
P <sub>16</sub>	Nº da Fatura (Nº da Nota de Remessa)		5 N	P16 <sup>=D</sup> 2
P <sub>17</sub>	Preço unitário		5 N	P5 <sup>×P</sup> 15 <sup>-P</sup> 17
P <sub>18</sub>	Nome do Viajante	50	15A	P18*P4



	1 - NOTÀ DE REHESSA	
ITEH	DESCRIÇÃO VERBAL	Pi
D <sub>1-1</sub>	Data	P1
D <sub>1-2</sub>	Número da nota de Remessa	P <sub>16</sub>
D <sub>1-3</sub>	Número de identificação do cliente	P <sub>2</sub>
D <sub>1-4</sub>	Código de Temassa	P <sub>3</sub>
D <sub>1-5</sub>	Número do viajante	7.
[b <sub>1-6</sub>	Quantidade encomendada	Pe
D <sub>1-7</sub> ]	Número do modelo	7,
D <sub>1-8</sub>	Linha	

D	3 - NOTA DE PAGAMENTO	
ITEM	DESCRIÇÃO VERBAL	Pi
D <sub>3-1</sub>	Data	P <sub>1</sub>
D <sub>3-2</sub>	Número da fatura	P <sub>16</sub>
n <sub>3-3</sub> ]	Importância	
D <sub>3-4</sub>	Linha	

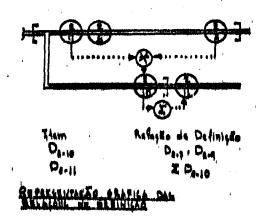
D	2 - PATURA	
ITEH	DESCRIÇÃO VEPBAL	P
D <sub>2-1</sub>	Data	P
D <sub>2-2</sub>	Húmero de Fatura	7
D <sub>2-3</sub>	Mumero de identificação do cliente	7
D <sub>2-4</sub>	Nome e endereço do cliente	
D <sub>2-5</sub>	Endereço de remessa	
D <sub>2-6</sub>	Armazem remetents	P
[02-7	Quantidade encomendada	P
D <sub>2</sub> -8	Número do modelo	•
D <sub>2-9</sub>	Preço unitário	
D <sub>2-10</sub>	Preçe total	
D <sub>2-11</sub>	Total da fatura	
D <sub>2-12</sub>	Linha	1.

D <sub>4</sub> - EXTRATO DE CONTA MENSAL				
ITEM	DESCRIÇÃO VERBAL	Pi		
D <sub>4-1</sub>	Nome e endereço do cliente	P <sub>10</sub>		
D <sub>4-2</sub>	Data	P <sub>1</sub>		
D <sub>4-3</sub>	Saldo anterior			
[D4-4	Número da fatura	P <sub>16</sub>		
D <sub>4-5</sub>	Data de fature	P <sub>1</sub>		
D4-6]	Importância da fatura			
D4-7	Linha	1		
D <sub>4-8</sub>	Nove Salde			

D <sub>5</sub> - RELATÓRIO DIÁRIO DE VENDAS ACUMULADAS			
TTEM	DESCRIÇÃO VERBAL	Pi	
D <sub>5-1</sub>	Data	P	
D <sub>5-2</sub>	Número do Viajante	P4	
D <sub>5-3</sub>	Nome do Viajante	P <sub>18</sub>	
D 5-4	Vendas nesta Data		
D <sub>5-5</sub> ]	Vendes acumulades neste mes		
D 5-6	Linha		
D <sub>5-7</sub>	Total de vendas nesta Data		

Analisemos cada um desses casos e formulemos a relação de definição correspondente.

Quanto a  $D_{2-10}$ , é obtida pelo produto da quantidade encomendada, pelo preço unitário correspondente. Então podemos escrever a relação de definição de  $D_{2-10}$  como sendo  $D_{2-7}$ . $D_{2-9}$ . Ora, $D_{2-9}$  não é simplesmente copiado de nenhum documento de entrada, nem obtido diretamente de  $P_{17}$ . Lembremos que, pelo enunciado, o preço unitário é dado em função do número do modelo e da área de preço em que o cliente está enquadrado. Assim, podemos dar como relação de definição de  $D_{2-9}$ :  $P_{17}$  ( $D_{2-8} \times P_{15}$ ). Não precisamos especificar o modo de obtenção do valor de  $P_{15}$ , pois, tanto pode ser obtido através de  $P_2$ , isto é, utilizando-se  $D_{2-3}$ , de acordo com o quadro da  $P_{2}$ 0.  $P_{2}$ 0.  $P_{2}$ 0.  $P_{2}$ 0.  $P_{2}$ 0.  $P_{2}$ 1.  $P_{15}$ 0. como através de  $P_{11}$ 0, utilizando-se  $P_{2-6}$ 0, pois  $P_{11}$ 1.



Nele vemos, também, a relação de definição de  $D_{2-11},D_{3-3}$   ${\bf Z}$  o  $D_{2-11}$  da fatura a que o pagamento corresponde; como  ${\bf Z}$  uma informação em documento de entrada, não há necessidade de representar a relação de definição correspondente pois essa informação não  ${\bf Z}$  obtida pelo sistema e sim a ele formacida.

Por reciocínio enflogo obtém-se a releção de definição de  $D_{4-8}$  que é  $D_{4-3}+D_{4-6}$  e que está representada no gráfico de pg. 113, bem como a referente a  $D_{5-7}$  que é  $D_{5-6}$ .

Vamos agora nos concentrar em D<sub>4-3</sub>, isto 6, saldo anterior

de determinado cliente. Pelo enunciado, verificamos que o saldo an terior  $\tilde{\mathbf{e}}$  obtido, deduzindo-se do novo saldo do extrato anterior os pagamentos relativos a faturamento anterior, ao período.

Depreendemos daí que um pagamento so deve ser deduzido se for referente a faturas emitidas, fora do período, isto é, se essa condição se verificar. Vejamos como expressar essa condição.

Lembremos que o período engloba datas posteriores a 10 do mês anterior e até o dia 10 do mês corrente. Em outras palavras, um pagamento não é deduzido se o mês constante da data da fatura aque se refere for o mês do extrato e o dia respectivo for menor ou i-gual a 10, isto é:

$$P_8(D_2) = P_8(D_4) \wedge P_7(D_2) \leq 10$$

ou se o mês da fatura é o anterior ao do extrato e o dia respectivo é maior que 10, ou seja

$$P_8(D_2) = P_8(D_4) - 1 \land P_7(D_2) > 10$$

chamando de  $C_{4-1}$  a condição composta das duas alternativas acima, podemos dizer que um pagamento deve ser deduzido se  $C_{4-1}$  for falsa ou seja se  $\bar{C}_{4-1}$  for verdadeira.

Podemos, agora, escrever a relação de definição de D<sub>4-3</sub>:

$$D_{4-8}(D_{4-1}, P_8-1) - \Sigma \bar{C}_{4-1}D_{3-3}(D_{4-1})$$

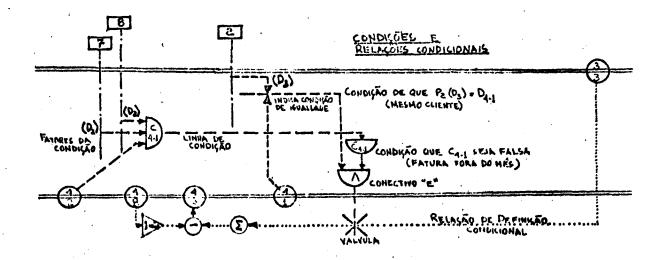
em que

$$C_{4-1} : P_8(D_2) = P_8(D_4) \land P_7(D_2) \le 10 \lor P_8(D_2) = P_8(D_4) - 1 \land P_7(D_2) > 10$$

isto é, D<sub>4-3</sub> é calculado tomando-se o novo saldo do extrato de conta anterior do cliente e dele subtraindo os pagamentos relativos a faturas emitidas, anteriormente ao período.

Essa expressão pode ser criticada dizendo que, atendendo a ela seriam também deduzidos pagamentos relativos a faturas posterio res ao dia 10 do mês corrente. Se lembrarmos, todavia, que os dados fornecidos ao sistema são sempre até o dia 10, verificamos que isso não pode ocorrer.

A representação gráfica dessa condição e dessa relação condicional é:



Observando melhor essa condição C<sub>4-1</sub> notamos que ela valditar não somente o critério para a dedução de um pagamento do saldo do extrato anterior, mas também para a inclusão de uma fatura, no extrato em preparação, desde que ela não tenha sido paga pelo cliente.

Essa última condição que chamamos de C<sub>4-2</sub> pode ser escrita:

$$c_{4-2}: \bar{J}D_3[D_{3-2}(D_{2-2})]$$

e significa que não existe nenhuma nota de pagamento, referente à fatura.

Então a composta dessas condições, isto é,  $C_{4-1}$   $\Lambda$   $C_{4-2}$  vai dar o critério de inclusão de uma fatura no extrato, isto é, vai dizer se uma fatura vai ou não produzir uma linha do extrato. Temos aí uma relação de produção condicional que pode ser escrita:

$$D_2 + D_{4-7} | C_{4-1} \wedge C_{4-2}$$

que da origem as relações de definição dos itens que constituem a linha e que são também condicionais, isto é, a relação

de definição de  $D_{4-4}$  que é  $D_{2-2}$ , de  $D_{4-5}$  que é  $D_{2-1}$  e de  $D_{4-6}$  que é  $D_{2-11}$ .

No gráfico da pg.113, podemos ver a representação gráfica completa da condição para que uma fatura seja incluída no extrato, isto  $\tilde{\epsilon}$ , que se refira ao cliente, tenha sido emitida no período e não tenha sido paga.

Outra relação de produção que influi na preparação do extrato é a de que só é emitido extrato para clientes com saldo não nulo, no fim do período a que êle se refere. Essa relação pode ser escrita:

$$P_2 \times P_8 \rightarrow D_4 \mid D_{4-8} \neq 0$$

No gráfico da pg.113 podemos ver a representação gráfica dessa condição e dessa relação.

Na pg.112, apresentamos, com referência a  $D_2$  e  $D_4$  todas as relações de produção, relações de definição e condições corres pondentes às relações condicionais. Tudo isso está esquematizado, no gráfico da pg.113.

Vamos agora cuidar de  $D_{5-4}$  e verificar como podemos obter o total de vendas nesta data, referente a certo viajante. Podemos escrever a seguinte relação de definição para  $D_{5-4}$ :  $\Sigma D_{2-11}$   $D_{5-1}$ ,  $D_{5-2}$ . Ela indica que uma fatura é considerada para calcular o total de vendas de certo viajante, se ela é dessa data e se é referente a esse viajante.

Quanto a  $D_{5-5}$ , se o extrato é referente ao primeiro dia do mês, o total de vandas acumulado é calculado considerando-se, apenas as vendas do dia, caso contrário deve levar em conta o total acumulado do relatório anterior. Podemos escrever a seguinte relação de definição para  $D_{5-5}$ 

$$D_{5-4}+C_{5-1}$$
,  $D_{5-5}$  ( $D_{5-1}-1$ )

em que

$$C_{5-1}: P_7(D_{5-1}) \neq 1$$

Nos quadros da pg.112 estão registradas as relações de definição e de produção, referentes ao relatório diário de vendas a cumuladas, bem como as condições correspondentes às relações condicionais. Nessa mesma página, em relação a todos os documentos do sistema, constam dados quanto a volume, dos documentos de entrada e exigências operacionais dos documentos produzidos pelo sistema, isto é, volume e prazos.

Assim, por exemplo, quanto ao volume das notas de remessa, temos pelo enunciado que, em média são recebidas trezentas notas por dia, no escritório central, cada uma com cinco linhas. Isso é expresso:

$$\bar{L} D_1/P_{7E}=300$$

Quanto ao prazo para a emissão do relatório de vendas, temos, pelo enunciado, que  $\tilde{\mathbf{e}}$  de dois dias, a contar do encerramento do movimento do dia a que se refere. Escrevemos então:

$$t_{E}(D_{5})-t_{I}(D_{5}) < 2 \text{ dias}$$

isto é, a data em que o relatório é produzido deve diferir de menos de dois dias da data que nele consta.

As exigências operacionais não são representadas grafica mente, mas o esquema da pg.113 apresenta a representação gráfica dos conjuntos de informação, documentos e seus itens, relações entre os itens dos documentos e os conjuntos de informação, relações de definição e produção, condicionais ou não, e condições envolvidas, no problema.

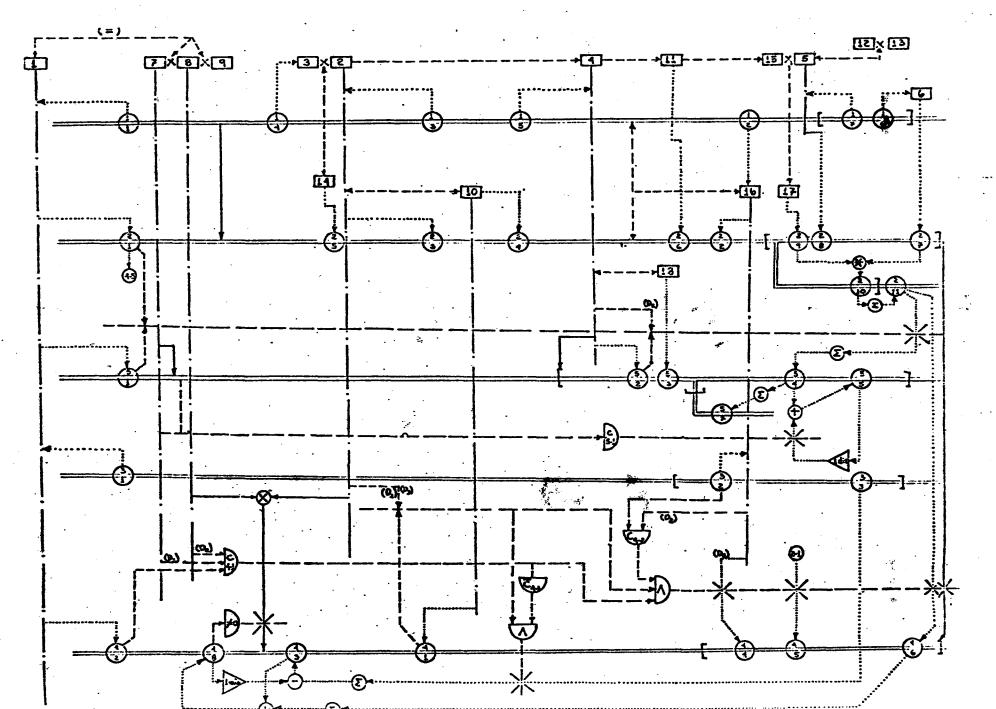
D <sub>L</sub> - MOTA DE BRHERSA			ENTRADA	
TEN	DESCREÇÃO VERBAL	P4	RELAÇÃO DE DEPIRIÇÃO	
1-1	Data	7,		
1-2	Mûnere de vote de Roncose	716	j	
1-3	Minere de identificação de allante	7,	}	
1-4	Cidigo de Tamesto	٠,		
1-3	Bémero do viajento	124	'	
1-6	Quintlidade encomendade	14		
1-1	Núsero de modeto	١٠,	ļ	
1-1	Linha	1	P1-4.7	

₽,	- PATURA		SATDA
TTEM	BESCEIÇÃO TERML	7,	telação de depidição
72-1	Pota	7,	
92-2	Rémeto da Fatura	716	'
92-3	Número de identificação de eliente	7,	
92-4	None e andereço de alleman	Pto	
P2-5	Indareço de remessa	714	
92-6	Armates tenatonts	P11	
92-7	Quantidada encouandeda	7.	ļ
	Número de medalo	۲,	· ·
D2-9	Proço unitério	7,,	P <sub>17</sub> (D <sub>2-8</sub> =P <sub>15</sub> )
P2-10]	Progo sotal .	1	13-7.72-9
9-11	lotal de feture	1	20 <sub>2-10</sub>
D <sub>2-12</sub>	Liuhe		P2-7 e 10
Polação	de Produção D <sub>1</sub> + D <sub>2</sub>	-	
trigën:	tias operacioneis		
	Valumer C 0 <sub>2</sub> /P <sub>7E</sub> = 200		
	Tougat tg(#g)-tg(#g) 4 E dias		

•	, - NOTA DE PAGAMENTO	ENTRAGA		
tten.	- PESCHIÇÃO VERBAL	7,	PELAÇÃO DE DEPINIÇÃO	
93-1	Pete	7,		
P3-2	Himero de Estura	710	,	
*3-3]	Importância	"		
91-4	Links		83-1,3	
Yelune	£ 3,/772 = 200	····	<del></del>	
	£ #3+4/93 + 1.5		•	

•	4 - EXTRATO DE CONTA MEMBAL		, ENTPADA
ITEN	BESCRIÇÃO VETBAL	1	PELAÇÃO DE DEFINIÇÃO
74-1	Nome e endereço da Eliente	110	
74-2	Dete	12	Os extratos deven our datados de 18 de esde mis.
4-5	Soldo enterior	ľ	94-6(P4-1.P6-1)-164-193-3(P4-1)
[04-4	Pipero da Estats	714	*2-2
	Date de fetura	1 -	*2-1
14-4]	Importância da fatuta		3 <sub>2-11</sub> 3 <sub>4-4</sub> a s
4-7	Links	-	
94-4	Pave Salde	L	P4-3*E94-6
Poloçõ	salde não mute	•	s cuda mõn para esda ellants com sa G <sub>4-1</sub> a G <sub>4-2</sub> õão anhas vorda-
Condig		_	_
	C <sub>4-1</sub> :[P <sub>8</sub> (P <sub>2</sub> )-P <sub>8</sub> (P <sub>4</sub> )AP <sub>7</sub> (P <sub>2</sub> )410]V A deca de fetero É até o'd 10 de mis esterior.	(a 10 40 wi	e servente, mas pesseries e
	C4-2: 303[032(022)] Não fei re	cobido pagá	uenço de fatura.
Erigia	Volume: C P4/Fgg-500 En midia		o 300 aperatos por uão. Madas à faturas (linhas) por Extre
	Toupon 10 < F <sub>7E</sub> (S <sub>A</sub> ) < 15 On extr.		ser emicidos estra es diss 10 e 15

,	3 - METULOMIO DIUNTO DE ARADAS VCAM	FLADAS	SATDA
ITEM	DESCREÇÃO VERDAL	PA	RELAÇÃO DE BEFESEÇÃO
*3-1	Bata	71	
[3,-2	Hűmere de Viajanta	74	ļ
D <sub>5-3</sub>	Jone do Viajesta	710	
93-4	Vendes mesta Data	į.	(02-11(03-1103-2)
93-3]	Tandas saunulades naste tils	j j	3 <sub>5-6</sub> +c <sub>3-1</sub> , 3 <sub>5-5</sub> (3 <sub>5-1</sub> -1)
9-4	Linto	- 1	3-1 . 3
95-7	Total de vendes sesta Deta	- [	295-4
teleçõ	es da Produçãos		
	P <sub>1</sub> *P <sub>3</sub>		
	74 - 23-4		
Condiç	io topociati		
	c <sub>5-1</sub> : P <sub>7</sub> (0 <sub>5-1</sub> )#1		•
trigia	cias operacionais:		
	Volume: C 9 <sub>5-6</sub> /9 <sub>5</sub> = 50		
	Tenpe: 0, (0,)-0, (0,) < 2 4445		



- 11

# 4 - CONCLUSÕES

Como tivemos a oportunidade de observar, os dois métodos são eficientes para resolver problemas do tipo dos apresentados. O método gráfico permite mais facilmente expressar relações, mesmo en tre informações referentes a processamentos não simultâneos, como é o caso do processamento diário e do processamento mensal, no problema de faturamento, controle de vendas e cobrança. Na solução al gébrica fomos levados a separar o problema em duas partes, uma referente a cada processamento, para exprimir as relações com maior facilidade.

Ambos os métodos consideram a informação como pertencendo a conjuntos, mas, enquanto o método gráfico considera esses conjuntos, individualmente, o método algébrico trabalha com o produto cartesiano deles. Isso permite que as expressões do método algébrico se refiram aos documentos como um todo, enquanto que, no método gráfico, há expressões referentes aos documentos e linhas de documentos: as de produção, e referentes aos itens dos documentos; as de definição.

O método algébrico exige a definição explícita de conjuntos de informação para abrigar os resultados derivados do processa mento, enquanto que o gráfico não. Assim, por exemplo, no problema da folha de pagamento, o método algébrico exigiu a definição de dois conjuntos de informação, além dos utilizados para o método gráfico, e, no problema de faturamento, controle de vendas e cobrança, de mais três conjuntos de informação.

A exigência do método gráfico de que se defina, no enunciado do problema, quais os documentos e seus itens, equivale à do método algébrico, de que se defina as áreas a serem utilizadas e a relevância de cada propriedade, em cada área. Assim, as relações obtidas pelo método gráfico são entre documentos e itens de documentos, ao passo que as obtidas pelo algébrico são entre áreas ou seja, entre arquivos. Como mencionamos, no primeiro parágrafo, nem sempre, essas relações funcionam correspondentemente.

O nível de abstração do método gráfico é maior que o do algébrico, no que se refere à maneira pela qual o Sistema de Informação armazena ou tem acesso à informação armazenada. Assim, na solução algébrica do problema da folha de pagamento, foi necessário

explicitar que o sistema tomava as informações quanto ao nome e sa lário-hora de certo operário, através de feixes estabelecidos, ora com a área referente ao pagamento anterior, ora com a referente às novas admissões. Já na solução gráfica, só foi necessário explicitar quais as relações entre os itens correspondentes e os conjuntos de informação. Também na solução gráfica do problema de faturamento, controle de vendas e cobrança, não foi necessária, ao contrário do que ocorreu na solução algébrica, a definição explícita do cadastro de clientes, nem do cadastro de preços. Quanto ao acesso a essa informação, ao contrário da solução algébrica, que exigiu a u tilização de feixes para reuni-la a solução gráfica limitou-se a mencionar as relações entre os conjuntos de informação correspon dentes.

As principais ferramentas do método algébrico são as malhas, feixes, funções e operações. Já no método gráfico são as con dições e as operações utilizadas para o estabelecimento das relações. Não se pode dizer que haja uma correspondência efetiva entre essas ferramentas, mas, pela observação comparativa das soluções apresentadas, pode-se dizer que, em geral, se obtem, no metodo gráfico,os efeitos correspondentes à utilização de uma malha no método brico, utilizando-se condições. Sobre os efeitos da utilização feixes no método algébrico, já esclarecemos, no parágrafo anterior, que, tendo em vista a maior capacidade de abstração do metodo gráfico, em geral é desnecessária a obtenção, quando se utiliza último método, de efeitos análogos. Quando a utilização de feixes, no metodo algebrico, e feita com o objetivo de selecionar informação, consegue-se, no método gráfico, efeito análogo, com a utiliza ção de condições. Reciprocamente, podemos dizer que, no método algébrico, se conseguem efeitos análogos aos obtidos com a utilização, no metodo gráfico, de condições, não somente com o emprego de malhas ou feixes, mas, em alguns casos, com a utilização da operação "SE-SENÃO". Quanto aos efeitos do uso de funções, no método al . gébrico, são em geral obtidos efeitos análogos, no metodo gráfico, com a utilização de operações.

Finalmente, quanto às operações utilizadas nos dois metodos, são conceitualmente distintas, pois, enquanto no metodo algebrico são operações com funções de linhas, definidas pelas tabelas constantes do Apêndice I (pg.142), no metodo gráfico são operações aritméticas ou lógicas, usuais.

	·	

#### CAPÍTULO V

## PESQUISA DE INFORMAÇÃO EM TABELAS

#### 1 - INTRODUÇÃO

Neste capítulo apresentamos, como exemplo de aplicação dos métodos de relacionamento entre dados e resultados discutidos anteriormente, um problema de pesquisa de informação em tabelas, que é a determinação do ponto de sela de matrizes. Tal aplicação permitiu evoluir rápida e eficientemente, da solução natural do problema, para uma outra, que é consideravelmente mais rápida e nem por isso ocupa mais memória.

Escolhemos a determinação do ponto de sela de matrizes como exemplo, por ser problema típico da pesquisa da informação, em tabelas, que é assunto de grande interesse da Informática.

Como o termo "Informática" tem sido utilizado com vários significados, vamos especificar o sentido em que é por nos empregado. Entendemos por Informática o ramo da Teoria Geral dos Sistemas voltado para os Sistemas de Informação.

Acreditamos que o estabelecimento de expressões ou esquemas, que relacionem os dados aos resultados de um problema de proces samento de informações, seja uma arma de grande valor para os estudiosos da informática.

Charles T.Meadow, um dos pioneiros desse novo ramo da Ciên cia, em 1970, ao descrever o estágio atual de desenvolvimento (State of the art), em que se encontrava a Informática, numa renomada série, que é o "Annual Review of Information Science and Technology" (25) p.169, afirmava que um dos mais importantes avanços da Organização, Manutenção e Pesquisa em Arquivos de Computadores, no a no de 1969, foi o reconhecimento da importância da separação entre os conceitos de organização física e de organização lógica dos arquivos e que, apesar da "grande variedade de sistemas de informação acessíveis aos usuários de computadores, não há, aparentemente, ne nhum modo de analisar essa multidão de produtos, em termos de seu desempenho comparativo, ou mesmo de sua capacidade de realizar de terminado objetivo. O atual comprador de um sistema de informação é um marinheiro, em mares não cartografados."

O volume de dados nas aplicações de computadores à informática é sempre elevado, o que obriga o uso de memórias auxiliares, por exemplo, discos, fitas magnéticas etc. Nesse caso, o tempo de acesso fica bastante aumentado e as técnicas de organização e manipu lação de arquivos influenciam ponderavelmente o tempo de processamento para a solução de determinado problema.

A determinação do ponto de sela foi assim executada para uma matriz, armazenada em disco, de modo que cada linha da matriz constituia um registro do arquivo.

Essa determinação foi feita, utilizando-se tres modos distintos de abordagem para o problema, cada um deles dando origem à elaboração de uma subrotina.

No primeiro modo, o problema foi abordado como julgamos que o seria por um programador que não tivesse conhecimento dos métodos expostos. Chamamos a solução obtida de solução natural do problema.

No segundo, estabeleceram-se, inicialmente, as relações entre os dados e os resultados, utilizando-se tanto o método algébrico quanto o gráfico

No terceiro, utilizou-se a abordagem sugerida por Knuth (12), lançando mão de propriedades demonstradas dos pontos de sela.

Para comparar o desempenho das tres subrotinas foi elaborado um programa que, lendo a matriz de cartões de dados, efetuava
seu armazenamento no disco e, posteriormente, chamava cada uma das
subrotinas.

Esse programa foi aplicado a quatro matrizes diferentes, anotando-se os tempos gastos pelas subrotinas para efetuar a pesquisa dos pontos de sela. Essas matrizes são apresentadas, no anexo II.

A matriz A é constituida, exclusivamente, de uns e zeros (pois branco é lido como zero, no formato utilizado) e tem grande número de pontos de sela e um número ainda maior de candidatos a pontos de sela, pois, o número de vezes em que o mínimo de uma linha ocorre, na linha, bem como o número de vezes em que o máximo de uma coluna ocorre, na coluna, é bastante grande.

A matriz B e formada por elementos que podem ser zero, um ou dois e tem reduzido número de pontos de sela, mas grande número de candidatos a pontos de sela.

A matriz C, cujos elementos variam entre zero e quatro, tem poucos pontos de sela e o número de candidatos a pontos de sela ja não é tão grande.

A matriz D é alfabética, apresentando poucos pontos de sela e um número de candidatos a pontos de sela menor que o das ou-

## 2 - SOLUÇÃO NATURAL DO PROBLEMA

Chama-se ponto de sela de uma matriz a um de seus elementos que seja, ao mesmo tempo, o mínimo da linha e o máximo da coluna a que o elemento pertence.

A solução natural desse problema seria tomar uma linha da matriz, determinar o mínimo dessa linha, e pesquisar, em cada coluna em que esse mínimo ocorre, se o seu valor é o máximo da coluna, repetindo esse procedimento para todas as linhas da matriz.

Foi essa a solução utilizada, na elaboração da subrotina SOL 1, documentada no Anexo II.

Nessa solução, foi utilizada uma estrutura de informação denominada pilha, para guardar os índices das colunas em que ocorre o mínimo de determinada linha, de modo a facilitar a pesquisa para verificação se o valor do mínimo é máximo, nessas colunas.

Para alojar a pilha na memoria central, utilizamos uma variavel indexada e, uma variavel simples, para guardar o índice do topo da pilha.

Essa solução é de processamento lento, pois exige o trânsito de todas as linhas da matriz pela memória central para a formação de cada coluna, todas as vezes em que se efetua a pesquisa, para verificar se o mínimo de determinada linha é também máximo, nes sa coluna.

## 3 - FORMULAÇÃO MATEMÁTICA DO PROBLEMA

Vamos estabelecer as relações entre os dados e os resultados do problema de determinação do ponto de sela, utilizando os dois métodos estudados.

Para isso, consideremos a matriz, como sendo uma aplicação definida no produto cartesiano de dois conjuntos finitos, que toma valores em outro conjunto, com a definição de elemento da matriz, apresentada no capítulo 0.

Particularizando os conjuntos dos quais se efetua o produto cartesiano, como sendo o conjunto dos inteiros no intervalo [1,80] e, tomando o outro conjunto, como sendo {0,1}, {0,1,2}, {0,1,2,3,4} e {A,B,C,D,E,F,G,H,I,J,K} temos as matrizes A, B, C,D

apresentadas no Anexo II, como exemplos dessa aplicação.

#### 3.1- APLICAÇÃO DO METODO ALGEBRICO

Consideramos cada elemento da matriz como sendo uma entidade e vamos estudar essas entidades num espaço de propriedades em
que o conjunto de coordenadas é formado pelas propriedades: "Linha
a que o elemento pertence", "Coluna a que o elemento pertence" e
"Valor do elemento".

Precisamos tomar cuidado com o nome dessa última propriedade pois, por coincidência, o valor da propriedade "Valor do elemento" coincide com o valor do elemento.

Podemos generalizar o conjunto de valores da propriedade "Linha a que o elemento pertence" como sendo o conjunto dos inteiros, no intervalo [1,m], em que m é um inteiro positivo, o conjunto de valores da propriedade "Coluna a que o elemento pertence" como sendo o conjunto dos inteiros no intervalo [1,n], em que n é um inteiro positivo e o conjunto de valores da propriedade "Valor do elemento" como sendo qualquer conjunto ordenado.

Nesse problema, o arquivo de entrada é:

A - Matriz dada, no qual cada registro reune todas as características correspondentes a cada elemento da matriz, isto  $\hat{e}, l\underline{i}$  nha, coluna e valor do elemento.

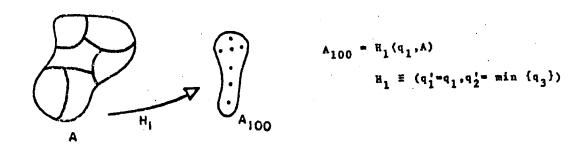
Através de uma série de transformações, deve-se obter o arquivo de saída:

A<sub>1</sub> - Pontos de sela, no qual cada registro reune todas as características correspondentes a cada ponto de sela, isto é, li-nha, coluna e valor do elemento que constitui o ponto de sela.

Na tabela abaixo, reunimos essas considerações, indicando as areas correspondentes a cada arquivo.

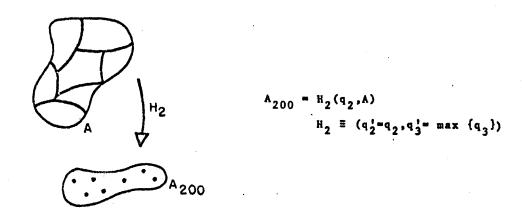
CONJUNTO DE COORDENADAS DO ESPAÇO DE PROPRIEDADES UTILIADO				^1
qį	NOME DA PROPRIEDADE	CONJUNTO DE VALORES	MATRIZ	PONTOS DE SELA
91	Linha a que o elemento pertence	1,2,,	. *	×
92	Columa a que o elemento pertence	1,2,,u	×	×
۹ <sub>3</sub>	Valor de clemento	Conjunto ordenado	×	×

Vamos inicialmente estabelecer em A a malha que reune em um dos seus elementos, todos os elementos da matriz pertencentes a uma determinada linha. Com a função H<sub>1</sub> dessa malha determinamos os mínimos das várias linhas.



Assim, na área  $A_{100}$  temos tantos pontos quantas são as 1inhas da matriz, cujas propriedades relevantes são apenas  $q_1$  e  $q_3$ , isto é, linha do mínimo e valor desse mínimo.

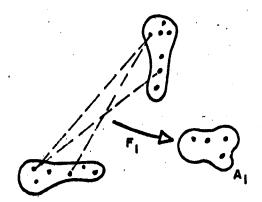
Estabeleçamos em A a malha que reune em um mesmo de seus elementos, todos os elementos da matriz, pertencentes a uma determinada coluna. Com a função H<sub>2</sub> dessa malha determinaremos os máximos das várias colunas.



Assim, na area  $A_{200}$  temos tantos pontos quantas são as colunas da matriz, cujas propriedades relevantes são apenas  $q_2$  e  $q_3$ , isto é, coluna do máximo e valor desse máximo.

Vamos agora estabelecer um feixe, no conjunto de areas  $\{A_{100}, A_{200}\}$ , determinado pela seleção  $q_{3:1} = q_{3:2}$ , isto  $\tilde{e}$ , que liga

a coluna e a linha onde o valor do máximo da coluna coincide com o valor do mínimo da linha. Com a função  $F_1$  desse feixe obtemos a área  $A_1$ .



 $A_1 = Y_1(q_{3:1}=q_{3:2}, A_{100}, A_{200})$  $Y_1 = (q_1'=q_{1:1}, q_2'=q_{2:2})$ 

Poderíamos, nessa solução, apenas ter dúvida sobre que os pontos de A, não correspondessem a todos os pontos de sela.

Vejamos, inicialmente, um aspecto em que poderia se funda mentar essa dúvida. Poder-se-ia pensar que, apesar de o máximo de uma linha e o mínimo de uma coluna serem iguais, tanto o máximo da coluna poderia ocorrer em outra linha como o mínimo da linha poderia ocorrer em outra coluna, ou ambas as coisas poderiam ocorrer.

Vamos mostrar que essa dúvida não se justifica, isto é, que quando o máximo de uma coluna é igual ao mínimo de uma linha, é também igual ao valor do elemento que pertence, simultâneamente, à linha e à coluna correspondentes.

Seja m o valor mínimo de certa linha i. Seja a ik um elemento dessa linha. Podemos escrever

Seja M o valor maximo de certa coluna j. Seja a l j um ele mento dessa coluna. Podemos escrever

Seja  $a_{ij}$  o elemento que pertence, simultaneamente, a linha i e  $\bar{a}$  coluna j.

$$a_{ij} \ge m_i$$

Por (2) podemos escrever

$$a_{ij} \leq M_{j}$$

οú

$$m_i \leq a_{ij} \leq M_i$$

$$C \leqslant a_{ij} \leqslant C \Rightarrow a_{ij} = C \dots CQD$$

Então, não paira dúvida sobre se cada ponto de A<sub>1</sub> corresponde a um ponto de sela.

Resta, agora, saber se os pontos de  $A_1$  correspondem a to-dos os pontos de sela. Pela propria definição de feixe e de ponto de sela verificamos que isso a verdade.

Resumindo, temos as seguintes expressões como solução do problema da determinação do ponto de sela de matrises:

# 3.2- SOLUÇÃO GRÁFICA

Temos, nesse problema, apenas dois documentos, isto  $\tilde{a}$ , a matrix que  $\tilde{a}$  documento de entrada e a relação dos pontos de sela, que  $\tilde{a}$  documento de saída. Vamos chamar a matrix de  $D_1$  e a relação dos pontos de sela de  $D_2$ .

No quadro a seguir, são apresentados os conjuntos de informação envolvidos no problema de determinação dos pontos de sela das matrizes, aquí consideradas. A generalização para qualquer matris á imediata, uma vez resolvido o problema particular.

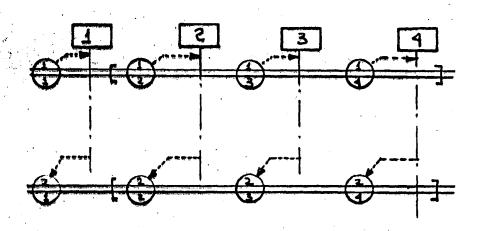
7,	NOME DO CONJUNTO	N N9 de elemen- tos	L NO de carac- teres	Relações
Pı	Nome de metris	4	1A	P1 = D1
P2	Múmero da linha	80	2 N	
P3	Número de colune	80	2 W	
P4	Valor do elemento	16	K\A£	

Nos quadros abaixo são apresentadas as relações entre os itens dos documentos e os conjuntos de informação.

D <sub>1</sub> - MATRIE					
TTEM	DESCRIÇÃO VERBAL	Pi			
D <sub>1-1</sub>	None de matrix	P,			
[D <sub>1-2</sub>	Númere da linha	P2			
P <sub>1-3</sub>	Número da colume	23			
P <sub>1-4</sub> ]	Valey de elemente	PA			
1-5	Linha				

D <sub>2</sub> - RELAÇÃO DOS PONTOS DE SELA				
ITEM	DESCRIÇÃO VERBAL	Pi		
D <sub>2-1</sub>	Nome de matris	P <sub>1</sub>		
D <sub>2-2</sub>	Número da linha	P <sub>2</sub>		
D <sub>2-3</sub>	Número de celuns	P <sub>3</sub>		
D <sub>2-4</sub> ]	Valor do ponte de sela	P4		
D <sub>2-5</sub>	Linha			

Não se deve confundir linha de documento com linha da matris, pois, no caso, cada linha de documento corresponde a um elemento da matris.



No gráfico acima, estão figuradas as relações existentes entre os itens dos documentos e os conjuntos de informação.

Quanto as relações de definição, nada há a observar, de Vez que os itens do documento de saída são, simplemente copiados do documento de entrada. A chave da solução do problema está nas relações de produção.

Ha duas relações de produção a considerar: a primeira é in condicional e diz que cada documento de saída é produzido por um do cumento de entrada e pode ser escrita  $D_1+D_2$ . A segunda é condicional e diz que uma linha do documento de entrada produz uma linha do documento de saída, se o elemento correspondente é ponto de sela e pode ser escrita  $D_{1-5}+D_{2-5}$   $C_{2-1}$ .

Vejamos, então, como exprimir a condição de que o elemento é ponto de sela. Pela definição de ponto de sela temos uma condição dupla, isto é, o elemento deve ser mínimo da linha e máximo da coluna. A condição de que é mínimo da linha pode ser escrita:

$$P_4(D_{1-5}) = \min D_{1-4} V D_{1-5}|D_{1-2} = D_{2-2}$$

A condição de que é máximo da coluna pode ser escrita:  $P_4(D_{1-5})=\max D_{1-4} \ V \ D_{1-5}|D_{1-3}=D_{2-3}$ 

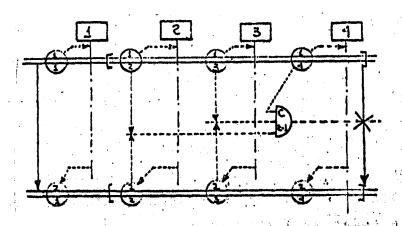
 $P_4^{(D_{1-5})=\max D_{1-4}}$   $P_{1-5}^{(D_{1-5})D_{1-3}=D_{2-3}}$ Então podemos escrever a expressão de  $\bar{C}_{2-1}$ :

 $P_4(D_{1-5}) = \min D_{1-4} V D_{1-5} D_{1-2} = D_{2-2} A P_4(D_{1-5}) = \max D_{1-4} V D_{1-5} D_{1-3} = D_{2-3}$ 

Obtivemos, assim, a solução gráfica do problema, apresentada a seguir e os quadros abaixo de que constam, para cada documento, as relações e condições correspondentes.

•	- MATERE		ENTRAGA
1120	sesenição versal	1.	MELAÇÃO DE DEFINIÇÃO
1-1	Jone de metris	7,	,
1-1	Moore de liebs Moore de coluce	7,	
*1-; •1-c]	Valor de plemets	7,	
1-3	Links		P1-2 = P1-4
	1 to 1 to 2		·
		٠.	•

P <sub>2</sub> - straffe pos pouros de dela			SATOA		
1787	JAGUST OKS143830	74	RELAÇÃO DE DEFINIÇÃO		
*2-1 [*2-2 *2-3 *2-6] *2-6	Pone de metris Sinero de Sinho Dimero de cojum Taior de pouso de selá Linho	P1 P2 P4	9 <sub>3+3</sub> 4 9 <sub>3+4</sub>		
Palogão	o de frotogles				
	0, • 0,				
•	01-5 + 22-5   C2-1	٠.			
Couligi	ine Reportator		17 18 18 18 18 18 18 18 18 18 18 18 18 18		
	€2-11F4(0{-3)mola 03-4 70	1-5 10	-1-01-1 <sup>h</sup>		
	94/9 <sub>1-9</sub> )-see \$ <sub>1-4</sub> v1		.,000,		



# 4 - SOLUÇÃO SUGERIDA PELA FORMULAÇÃO MATEMÁTICA DO PROBLEMA

Ao aplicar o método algébrico para a formulação matemática do problema, de determinação do ponto de sela, observamos que basta determinar o mínimo de cada linha e o máximo de cada coluna, pois os pontos de sela estarão situados em todo o cruzamento de linha e coluna em que o mínimo da linha for igual ao máximo da coluna. Para isso, basta transitar uma única vez com cada linha da matriz pela memória central, pois, nessa oportunidade, podemos determinar o mínimo da linha e verificar se, em alguma coluna precisamos corrigir os máximos provisórios, que foram inicialmente tomados, como os valores dos elementos da la. linha.

Foi essa a solução utilizada na elaboração da subrotina SOL 2, documentada no Anexo II.

Se observarmos bem, nem seria necessário o armazenamento da matriz no disco, para que se pudesse determinar seus pontos de sela. Bastaria que cada linha da matriz fosse lida de cartões de dados, para que, ao transitar pela memória central, se pudesse extrair dela toda a informação necessária.

Essa solução, assim, além de extremamente mais rápida, pois basta que cada linha da matriz transite uma única vez pela memória central, ocupa menos memória que a solução natural do problema.

# 5 - SOLUÇÃO CONSIDERADA ÓTIMA PARA EFEITO DE COMPARAÇÃO

Baseado em teorema demonstrado em (12), Knuth sugere ser possível obter-se uma solução ainda melhor que a apresentada em seu livro, para a determinação do ponto de sela de matrizes.

Esse teorema afirma que o valor do elemento que é ponto de sela é o máximo dos mínimos que ocorrem nas linhas e o mínimo dos máximos que ocorrem nas colunas, se ambos coincidirem.

Assim, ao transitar com uma linha pela memória central podemos extrair dela toda a informação para corrigir os máximos provisórios das colunas, tomados, inicialmente, como os valores dos elementos da la. linha, e determinar seu mínimo. Nem precisamos guardar todos os mínimos das linhas. Basta guardar em uma pilha os índices das linhas em que o máximo dos mínimos ocorre. Ao terminar o trânsito de todas as linhas da matriz pela memória central, de-

terminamos o mínimo dos máximos das colunas e armazenamos, em uma pilha, as colunas em que ocorre.

Essa solução e, efetivamente melhor que a tomada como base, pois, dentro de um unico loop são feitas as comparações necessárias à verificação das implicações do valor de cada elemento, na determinação do máximo e do mínimo.

Foi essa a solução utilizada na elaboração da subrotina SOL 3, documentada, no Anexo II e, sem duvida, foi a solução mais eficiente que pôde ser dada para o problema, seja quanto à rapidez de processamento, seja quanto ao aproveitamento da memória.

#### 6 - RESULTADOS DO PROCESSAMENTO

O processamento comparativo das três subrotinas foi feito no computador IBM1130 do Centro de Processamento de Dados da Escola de Engenharia de São Carlos. Esse Computador apresenta a seguinte configuração:

- Unidade Central IBM-1131, modelo 2D com 32K. Capacidade de 32 768 palavras de 16 bits.
- Leitora Perfuradora de cartões IBM -1142, modelo 7.Lê 400 cartões, por minuto e perfura 160 colunas, por segundo.
- Impressora IBM-1132. Imprime dados alfa-numéricos à velocidade de 110 linhas por minuto. A linha de impressão compreende 120 caracteres.
- Impressora IBM-1403. Imprime dados alfa-numéricos à velocidade de 600 linhas, por minuto.
- Canal Multiplex 1133.
- Traçador de gráficos Plotter IBM-1627, modelo 2. Área de traçado: 29 1/2 polegadas por 120 pes.
  Incremento: etapas de 1/100 de polegada a velocidade de 200 etapas por segundo.
- Leitora de fita de papel IBM-1134.
- Perfuradora de fita de papel IBM-1055.
- Memoria auxiliar IBM-2310- Modelo BO.2. Capacidade: 1.024.000 palavras de 16 bits.
- O Sistema de operação do CPD é feito em "closed shop" e "batch processing", razão pela qual giremes que nos contentar com medidas de tempo feitas pelos operadores do Centro, baseadas no

marcador de tempo da unidade central de processamento, do computador. A precisão desse marcador é de centésimo de hora, existindo, porém, uma escala que permite a leitura, ainda que não muito precisa, de milésimo de hora.

Os resultados do processamento são apresentados, no quadro a seguir:

RESUMO DO PROCESSAMENTO COMPARATIVO	PROCESSAMENTO PONTO DE CETA				
MATRIZ	SOL.1	SOL.2	SOL.3	DE SELA EN- CONTRADOS	
A	409	. 8	. 8	80	
В	315	varia entre .5 e .4	varia	4	
С	167		entre .4 e .3	2	
D	114			4	

OBSERVAÇÃO - Os tempos estão expressos em centésimos de hora e correspondem a diferenças de leitura do marca dor de tempo da UCP do IBM1130 do CPD da EESC.

Como podemos observar no quadro acima,os resultados do processamento confirmaram, plenamente, a grande eficiência das subrotinas Sol.2 e Sol.3, em relação à subrotina Sol.1.

Quanto à comparação entre o desempenho da subrotina Sol.2 e Sol.3, ainda que dificultada, porque seus tempos de processamento situam-se na faixa não muito precisa dos milésimos de hora, pode-se observar que são de eficiência bastante semelhante. A subrotina Sol.3 mostrou ser efetivamente a mais eficiente, pois em todos os casos apresentou tempo de processamento da ordem de um milésimo de hora menos que a subrotina Sol.2, exceto na pesquisa efetuada, na matriz A, quando o tempo de impressão, tendo em vista a grande quantidade de pontos de sela encontrados, dominou, fazendo com que fossem iguais os tempos de processamento das duas subrotinas.

Verificamos pelo quadro comparativo que a vantagem da solução elaborada após a formulação matemática do problema, em relação a solução natural é tão maior quanto maior for o número de can
didatos a ponto de sela, pois, enquanto que na solução natural, esse número tem grande influência no tempo de processamento, em nada
influi, na solução elaborada.

## BIBLIOGRAFIA BÁSICA

- (1) AN INFORMATION algebra-Phase I Report of the Language Structure Group of the CODASYL Development Committee. Communications of the ACM 5,4(Apr.1962),190-204. Computing Reviews, Rev.2621. (Resumo)
- (2) BOSAK,R. An information algebra. 16th Nat.Meeting Association Computing Machinery, Los Angeles, (1961), paper 6B-1.

  Communications of the ACM, 4(7) jul.1961 (Resumo); Information Processing Journal, Rev.3719 (Resumo).
- (3) BOYCE, Raymond Francis Topological reorganization as an aid to program simplification. (PhD Thesis), Purdue University Lafayette, Ind., June 1972. Computing Reviews, Rev. 24, 395 (Resumo)
- (4) DATA Base Task Group Report to the CODASYL Programming Language Committee. (Oct.1969 report) ACM, New York, N.Y.1969. Computing Reviews, Rev.19,080 (Resumo)
- (5) DETAB-X: Preliminary specifications for a decision table Structured Language, CODASYL Systems Group, September, 1962.
- (6) EVANS, O.Y. Advanced analysis method for integrated Electronic data processing. General Information Manual F20-8047, Intern. Bus. Machines Corp. New York.
- (7) GENERAL Data Files and Processing Operations, SHARE Committee on Theory of Information Handling, Report TIH-1,SSD-71, Item C-1663 (1959).
- (8) GILDERSLEEVE, Thomas R. Decision tables and their practical application in data processing. Prentice-Hall, Inc. Englewood Cliffs, N.J., 1970, 206p. Computing Reviews, Rev. 20,655. (Resumo)
- (9) GORN,S. Introductory remarks. In: Working Conference on Mechanical Language Structures, August. 14-16, 1963. Princenton, N.J. Communications of the ACM, 7(2):51,Feb.1964.
- (10) IBM 7090 Programming Systems: SHARE 7090 9PAC; Part 1, Introduction and General Principles (J28-6166); Part 2, The File Processor (J28-6167); Part 3, The Reports Generator (J28-6168), Intern.Bus.Machines Corp., New York, 1961.
- (11) KATZ, Jesse H. & McGEE, William C. An experiment in non-procedural programming. Proc. AFIPS 1963 Fall Joint Comput.

  Conf., Las Vegas, Nov. 1963, 1-13. Spartan Books, Baltimore, Md. Computing Reviews, Rev. 6323 (Resumo)
- (12) KNUTH, Donald E. The art of computer programming. London, Addison Wesley, 1969. v.I.
- (13) LANGEFORS, B. Some approaches to the theory of information systems. BIT 3,4(1963), 229-254. Computing Reviews, Rev. 6399 (Resumo)

- (14) LIST of Conferees, Working Conference on Mechanical Language Structures, August 14-16, 1963. Communications of the ACM, 7 (2):88, feb.1964.
- (15) LOMBARDI, Lionello A. A general business-oriented language based on decision expressions. <u>Communications of the</u> <u>ACM</u> 7,2(Feb.1964),104-111. <u>Computing Reviews</u>, Rev. 9013 (Resumo)
- (16) LOMBARDI, Lionello A. Mathematical structure of non-arith metic data processing procedures.

  <u>ciation for Computing Machinery</u>, 9:136-159, 1962. <u>Information Processing Journal</u> Rev. 2123 (Resumo); <u>Computing Reviews</u> Rev. 3163 (Resumo)
- (17) LOMBARDI, Lionello A. Nonprocedural data system languages.
  Invited paper, 16th National Meeting of ACM, 1961.
- (18) LOMBARDI, Lionello A. On table operating algorithms. Proc.

  IFIP, Congress 62 Munich, 1962, 230-232. North Holland
  Pub.Co.Amsterdam. Computing Reviews Rev. 4174 (Resumo)
  Information Processing Journal Rev. 2396 (Resumo)
- (19) LOMBARDI, Lionello A. Theory of files. Proc.1960 EJCC,
  New York, N.Y., Dec.1960, 137-141. Computing Reviews,
  Rev. 3165 (Resumo)
- (20) McDANIEL, Herman, ed. Aplications of decision tables.

  Brandon/Systems Press, Princeton, N.J., 1970, 226p.

  Computing Reviews, Rev. 20,775, (Resumo)
- (21) McGEE, William C. File structures for generalized data management. Proc. IFIP Congress 1968, Applications 1, Booklet F, 68-73. Computing Reviews Rev. 16,557 (Resumo)
- (22) McGEE, William C. Generalized file processing. Annual review in automatic programming, vol.5, 1969, 77-149.

  Computing Reviews Rev. 18,049 (Resumo)
- (23) McGEE, William C. The formulation of data processing problems for computers. Advances in Computers, New York, Academic Press. v.4, 1973.
- (24) McGEE, William C. The property classification method of file design and processing. Communications of the ACM, 5 (8):450-453, Aug. 1962. Computing Reviews, Rev. 3464 (Resumo); Information Processing Journal Rev. 2769 (Resumo)
- (25) MEADOW, Charles T. Organization, maintenance and search of machine files. Annual Review of Information Science and Technology, 5:169-191, 1970.
- (26) NAMIAN, Paul Algebra of management information. <u>Proc. IFIP</u>

  <u>Congress 1968</u>, Applications 1, Booklet F, 63-67. <u>Computing</u>

  Reviews, Rev. 18,211 (Resumo)

- (27) NAMIAN, Paul Approche théorique du traitement des informations administratives. Revue Automatisme 10 (1966)
- (28) NAMIAN, Paul Etude théorique du traitement des fichiers se quentiels en automatisme administratif. Revue Metra 10 (1966)
- (29) NOONAN, Robert E. Computer programming with a dynamic algebra. (PhD Thesis), Purdue University, Lafayette, Ind., June, 1971. Computing Reviews, Rev. 21,939 (Resumo)
- (30) PATTERSON, A.C. Requirements for a generalized data base management system. Proc. AFIPS 1971 Fall Joint Computer Conf., AFIPS Press, Montvale, N.J.,515-522: Computing Reviews, Rev. 23,148 (Resumo)
- (31) ROBERTS, David C. File organization techniques. Morris Rubinoff, Advances in Computers. Academic Press, N.Y., 1972 p.115-174.
- (32) SAMMET, Jean Programming languages: history and fundamentals.

  Prentice-Hall. Englewood Cliffs, N.J., 1969, 785p. Computing Reviews, Rev.17,682 (Resumo)
- (33) SAMMET, Jean Programming languages: history and future.

  <u>Communications of the ACM</u>, 15,7(July 1972),601-610. <u>Computing Reviews</u>, Rev. 24,398 (Resumo)
- (34) STEEL Jr, T.B. Beginnings of a theory of information handling. Communications of the ACM, 7(2):97-103, feb. 1964.
- (35) STEEL Jr, T.B. Formal language description languages for computer programming. North-Holland Publ.Co., Amsterdam, The Netherlands, 1966, 330p. Computing Reviews, Rev. 10,465, (Resumo)
- (36) STEEL Jr, T.B. The foundations of a theory of information processing. FN-6341, System Development Corp., Santa Monica, Calif., 1962.
- (37) STEEL Jr, T.B. The foundations of a theory of data processing. Preprints of Papers Presented at the 16th National Meeting of the ACM, Sept. 5-8, 1961, paper 6B-2. Communications of the ACM, 4(7) jul. 1961 (Resumo); Information Processing Journal, Rev. 3714 (Resumo)
- (38) YOUNG Jr, J.W. & KENT, H. Abstract formulation of data processing problems. Journal of Industrial Engineering, 9, 471-479 (1958)

,	

#### APÊNDICE I

#### UMA ÁLGEBRA DA INFORMAÇÃO

Language Structure Group of the CODASYL Development Committee.\*

## 1- INTRODUÇÃO

Esta Álgebra apresenta uma estrutura para a descrição de problemas de processamento da informação. Considerando as generalizações adotadas, pretende-se tornã-la capaz de possibilitar a des
crição de problemas que se situam num amplo espectro de processamento de dados. As regras da Álgebra espelham as características
comuns desses problemas e são formuladas, em linguagem formal mate
mática, particularmente lançando mão de Álgebra Moderna e Teoria
dos Conjuntos.

# 2- ESPAÇOS DE PROPRIEDADES

- 2.1 CONCEITOS NÃO DEFINIDOS

  Propriedade (q)

  Valor (v)

  Entidade (e)
- 2.2 POSTULADO: Cada propriedade tem um e somente um conjunto de valores a ela associado. O conjunto de valores assim associado será chamado Conjunto de Valores da Propriedade (V).
- 2.3 POSTULADO: Cada conjunto de valores de propriedade contém pelo menos dois valores:
  - $\Omega$  (não definido, irrelevante)
  - θ (desconhecido, relevante, porem não conhecido)
- 2.4 POSTULADO: Cada entidade tem um e somente um valor a ela associado de cada conjunto de valores de propriedade.

<sup>\*</sup> Os conceitos apresentados neste apêndice foram extraídos do Phase I Report do LSG da CODASYL. (1)

- 2.5 DEFINIÇÃO: Um Conjunto de Coordenadas (Q) é um conjunto finito de propriedades distintas.
- 2.6 DEFINIÇÃO: O Conjunto de Coordenadas Nulo é aquele que não contem nenhuma propriedade.
- 2.7 DEFINIÇÃO: Dois conjuntos de coordenadas são equivalentes se e somente se contêm, exatamente, as mesmas propriedades.
- 2.8 DEFINIÇÃO: O Espaço de Propriedades (P) de um conjunto de coordenadas Q  $\in$  o produto cartesiano:

$$P = V_1 \times V_2 \times V_3 \times \ldots \times V_n$$

em que  $V_i$  é o conjunto de valores de propriedade associado à i-ésima propriedade de Q. Cada ponto (p) do espaço de propriedades será representado por uma n-upla  $p=(a_1,\ a_2,\ a_3,\ \ldots,\ a_n)$  em que  $a_i$  é algum valor de  $V_i$ . A ordem das propriedades no conjunto é apenas por conveniência. Se n=1 então  $P=V_1$ . Se n=0 então P é o espaço nulo.

- 2.9 DEFINIÇÃO: O ponto representativo (d) de uma entidade (e) num espaço de propriedades (P) é um ponto de (P) tal que se d = (a<sub>1</sub>, a<sub>2</sub>, a<sub>3</sub>, ..., a<sub>n</sub>), então a<sub>i</sub> é o valor associado a e do i-ésimo conjunto de valores de propriedade de P para i = 1,2,3, ...,n. (d) é a representação de (e) em (P).
- 2.10 TEOREMA: Cada entidade tem um e somente um ponto representa tivo, num determinado espaço de propriedades.

Esse teorema é uma consequência imediata de 2 (4) e 2 (9). Note-se que várias entidades podem ser representadas por um mes mo ponto representativo.

- 2.11 DEFINIÇÃO: Um espaço de propriedades discriminatório para um conjunto de entidades é um espaço de propriedades em que nenhum ponto representativo representa mais que uma entidade. O conjunto de coordenadas correspondente será chamado conjunto de coordenadas discriminatório.
- 2.12 DEFINIÇÃO: Se
  - (a) Q e Q' são conjuntos de coordenadas
  - (b) Q é discriminatório e
  - (c) ¥ Q'⊂ Q, Q' e não discriminatório,

Então Q é um conjunto de coordenadas discriminatório básico.

#### 3- - LINHAS E FUNÇÕES DE LINHAS

- 3.1 DEFINIÇÃO: Uma linha (L) é um conjunto ordenado de pontos escolhidos de P. A ordem de grandesa (n) da linha é o número de pontos abrangidos pela linha.
- 3.2 NOTAÇÃO: Uma linha L de ordem de grandeza n  $\tilde{\mathbf{e}}$  escrita da seguinte maneira:

$$L = (p_1, p_2, ..., p_n).$$

3.3 DEFINIÇÃO: Uma função de linhas (FOL) é uma aplicação que associa um e somente um valor a cada linha de P. O conjunto de valores distintos associados por uma FOL é o Conjunto Imagem da FOL.

O conjunto imagem de uma FOL pode estar, mas não estã, necessáriamente, contido no conjunto de valores de propriedade de qualquer das propriedades de P.

- 3.4 NOTAÇÃO: É conveniente escrever uma FOL na forma de funcional f(X) em que f é a FOL e X é a linha.
- 3.5 DEFINIÇÃO: Duas FOLs são iguais se e somente se associam o mesmo valor a mesma linha para todas as linhas em P.
- 3.6 DEFINIÇÃO: Uma FOL é independente de ordem se associa o mesmo valor a todas as linhas que contém, exatamente, os mesmos pontos, independentemente da ordem dos pontos na linha.
- 3.7 DEFINIÇÃO: A FOL Nula  $\tilde{\mathbf{e}}$  aquela cujo conjunto imagem contem apenas  $\Omega$ .
- 3.8 DEFINIÇÃO: Uma FOL é de ordem de grandesa n se não é a FOL nula, e se associa o valor  $\Omega$  a todas as linhas cuja ordem de grandeza não é n.

Essa definição torna possível distinguir-se FOLs que são definidas apenas para linhas com um número específico de pontos. Note-se que a FOL não é necessariamente definida para todas as linhas de dimensão n.

3.9 DEFINIÇÃO: Uma FOL de ordem (OFOL) é uma FOL cujo conjunto imagem está contido em um campo de variação para o qual e-xiste uma relação de ordem R que é não reflexiva, antissimétrica e transitiva.

#### Alem disso:

a) Para as relações definidas nos campos de variação abaixo, foram atribuídos nomes específicos à OFOL.

CAMPO DE VARIAÇÃO

NOME DA OFOL

Ω < Θ < -∞ < conjunto dos números reais < +∞

Real (a relação R é "menor que")

Ω < θ < conjunto dos inteiros negativos < 0 < conjunto dos inteiros positivos Inteira (a relação R é "menor que")

Ω R Θ R Falso(F) R Verdadeiro(V)

Booleana

Falso(F) R Verdadeiro(V)

Seleção

b) Diz-se que uma FOL de ordem e de mesmo tipo que seu cam po de variação.

Note-se que uma FOL pode ser de mais de um tipo, tendo em vista que certos campos de variação são sub-conjuntos de outros. Por exemplo, uma OFOL Inteira é também uma OFOL Real, mas não, reciprocamente. Uma OFOL de Seleção é também uma OFOL Boole ana, mas não, reciprocamente.

10. DEFINIÇÃO: Uma FOL que não seja uma FOL de ordem serã chamada uma FOL não de ordem.

# 4- ÁREAS E FUNÇÕES DE ÁREAS

- 4.1 DEFINIÇÃO: Uma Āreα é qualquer sub-conjunto de P.
  Note-se que tanto linhas como áreas são formadas por conjuntos de pontos. A diferença é que os pontos de uma linha são ordenados.
- 4.2 DEFINIÇÃO: Uma Função de Áreas (FOA) é uma aplicação que associa um e somente um valor a cada área. O conjunto de valores distintos associados por uma FOA é o Conjunto Imagem da FOA.
- 4.3 NOTAÇÃO: É conveniente escrever uma FOA na forma de funcional f(X) em que f é a FOA e X é a área.
- 4.4 TEOREMA: Se
  - a) B é uma área de P e

b) M é uma linha que contém todos os pontos de B e nenhum outro.

então para cada FOA f existe uma FOL f' que  $\acute{e}$  independente de ordem tal que f(B)=f'(M) para todas as linhas M que satisfazem (b).

PROVA: Seja f uma FOA e B uma área. Por definição f associa a B um e somente um valor. Seja  $v_B$  esse valor. Consideremos, agora, a linha M, que contém todos os valores de B e nenhum outro e definamos uma aplicação g que associa a M o valor  $v_B$ . Pela definição de FOL, g é uma FOL. Seja M' uma linha que difere de M apenas pela ordem dos pontos. Nada nos impede de fazer por definição  $g(M') = v_B$ .

Mostramos então que pudemos definir uma FOL g, com as características desejadas, donde podemos concluir que para dada FOA f existe a f' tal como enunciada. CQD

# 5- FEIXES E FUNÇÕES DE FEIXES

5.1 DEFINIÇÃO: Um Conjunto de Áreas  $\phi$  de ordem n  $\tilde{e}$  uma n-upla de áreas  $(A_1, A_2, \ldots, A_n) = \phi$ .

Conjunto de áreas é uma denominação genérica para uma coleção de áreas em que a ordem é significante.  $\phi$  consiste de uma ou mais áreas que são consideradas, simultaneamente.

- 5.2 DEFINIÇÃO: O Feixe  $B=B(b,\phi)$  de um conjunto de áreas  $\phi$  para uma OFOL de Seleção b  $\hat{\mathbf{e}}$  o conjunto de todas as linhas L tais que, se
  - a)  $\phi = (A_1, A_2, ..., A_n)$
  - b) L = (p<sub>1</sub>, p<sub>2</sub>,..., p<sub>n</sub>) em que p<sub>i</sub> e um ponto de A<sub>i</sub> para i = 1,2, ..., n

então b(L) = Verdadeiro. A função b é chamada função que determina o Feixe B. Diz-se que o Feixe é de mesma ordem que o conjunto de áreas.

Note-se que cada linha do feixe deve conter um e apenas um ponto de cada área. O conceito de feixe nos dá um método de ligar pontos de várias áreas, de modo a poder considerá-los, em conjunto.

5.3 DEFINIÇÃO: Uma Função de um feixe (FOB) é uma aplicação que

associa uma área a um feixe, tal que:

- a) ha uma correspondência entre as linhas do feixe e os pontos da área;
- b) o valor de cada propriedade de cada ponto da área é definido por uma FOL da linha do feixe correspondente ao ponto;
- c) o conjunto imagem dessa FOL deve ser um sub-conjunto do correspondente conjunto de valores da propriedade.

Note-se que cada linha do feixe é levada em um ponto ( não necessariamente diferente) cujos valores das coordenadas são determinados por um conjunto de FOLs.

- 5.4 DEFINIÇÃO: A FOB nula é aquela que associa a área que contém o único ponto  $(\Omega,\Omega,\ldots,\Omega)$  a todos os feixes.
- 5.5 NOTAÇÃO: Uma FOB pode ser expressa de três modos, equivalentes

a) 
$$F \equiv (f_1, f_2, \dots, f_k, \dots, f_m)$$
b)  $F \equiv (q_1' = f_1, q_2' = f_2, \dots, q_k' = f_k, \dots, q_m' = f_m)$ 

$$\begin{cases} q_1' = f_1 \\ q_2' = f_2 \end{cases}$$
c)  $F \equiv \begin{cases} q_k' = f_k \\ \vdots \\ q_m' = f_m \end{cases}$ 

em que F é a FOB,  $q'_k$  é a k-ésima propriedade dos pontos da área associada pela FOB, m é o número de propriedades e  $f_k$  é uma função de  $q_{ij}$  para  $i=1,2,\ldots,n$  e  $j=1,2,\ldots,m$ , onde  $q_{ij}$  é a j-ésima propriedade da i-ésima área do conjunto de áreas; n é a ordem de grandeza das linhas compreendidas pe lo feixe;

Na forma (a) a f correspondente a cada q' deve ser especificada.

Na forma (b) e (c) qualquer das expressões  $q_k^i = f_k^i$  pode ser omitida. Nesse caso entende-se que  $q_k^i = q_{nk}^i$  onde  $q_{nk}^i$  é a

propriedade  $q_k$  dos pontos da n-ésima área, isto é, da área que contém o último ponto de cada linha. Se nenhum dos fs for especificado, isto é, F = ( ) entende-se que  $q_k^* = q_{nk}$  para k = 1, 2, ..., m.

5.6 NOTAÇÃO: A área A associada por uma FOB F ao feixe B pode ser expressa das seguintes maneiras:

A = F(B)

 $A = F(b, \phi)$ 

 $A = F(b, A_1, A_2, ..., A_n)$ 

5.7 NOTAÇÃO: Entende-se por A(s) uma area que é determinada por A(s) = F(s,P) em que F = (

s e chamada função de seleção para A.

A area A(s) contem pontos selecionados de todo o espaço de propriedades P, pelo critério de seleção s.

# 6- MALHAS E FUNÇÕES DE MALHAS

- 6.1 DEFINIÇÃO: Se A é uma área e g é uma FOL definida sobre linhas de ordem de grandeza l de A, então a MALHA G=G(g,A) de uma área A para uma FOL g, é uma partição de A por g tal que rada classe de equivalência da partição consiste de todos os pontos de A que são levados por g no mesmo valor. A função g será chamada função que determina a Malha G.

  O conceito de Malha não implica em nenhuma relação de ordem
  - O conceito de Malha não implica em nenhuma relação de ordem quer entre os pontos de um elemento da malha (classe de equivalência da partição), quer entre os elementos da malha.
- 6.2 NOTAÇÃO: O elemento da malha que  $\tilde{\mathbf{e}}$  o sub-conjunto de A definido pelos pontos para os quais o valor de g  $\tilde{\mathbf{e}}$  c,  $\tilde{\mathbf{e}}$  expresso por  $G(g,A)|_{g=c}$

Note-se que diferentes elementos da malha não contêm, necessariamente, o mesmo número de pontos.

- 6.3 DEFINIÇÃO: Uma Função de uma Malha (FOG) é uma aplicação que associa uma área a uma malha tal que
  - a) Existe uma correspondência entre os elementos da malha e os pontos da área associada.
  - b) O valor de cada propriedade de um ponto da area associada é definido por uma FOA do elemento da malha correspon dente.

- c) O conjunto imagem da FOA deve ser um sub-conjunto do cor respondente conjunto de valores da propriedade.
- 6.4 NOTAÇÃO: Uma FOG pode ser expressa de 3 modos equivalentes:

a) 
$$H \equiv (f_1, f_2, ..., f_k)$$
  
b)  $H \equiv (q_1' = f_1, q_2' = f_2, ..., q_k' = f_k)$   
 $q_1' = f_1$   
 $q_2' = f_2$   
 $\vdots$   
 $q_k' = f_k$ 

em que H é a FOG,  $f_i$  é uma FOA,  $q_i'$  é a i-ésima propriedade dos pontos da área associada, k é o número de propriedades. Na forma (a) a f correspondente a cada q' deve ser especificada. Nas formas (b) e (c) qualquer expressão  $q_i' = f_i$  pode ser omitida. Nesse caso, entende-se que  $q_i' = \Omega$ .

6.5 NOTAÇÃO: A área A associada por uma FOG H à malha G=G(g,A<sub>1</sub>) tando pode ser expressa por A=H(G) como por A=H(g,A<sub>1</sub>)

# 7- ORDENAÇÃO DE ÁREAS

7.1 DEFINIÇÃO: Uma Ordenação O(f,A) de uma area A por uma OFOL f de ordem de grandeza 1 é um conjunto de pontos  $(p_1,p_2,\ldots,p_n)$  escolhidos de A de tal modo que o conjunto esgota A e ou  $f(p_i) < f(p_{i+1})$  ou  $f(p_i) = f(p_{i+1})$ .

Nos conceitos de área, feixe e malha a noção de "ordem"foi deliberadamente ignorada, por não ser necessária. Todavia, em algumas aplicações práticas sua introdução é necessária.

- 7.2 DEFINIÇÃO: Diz-se que uma ordenação é estrita para uma determinada OFOL se e somente se não existe nenhum ponto tal que f(p<sub>i</sub>)=f(p<sub>i+1</sub>). Diz-se que uma ordenação é não estrita para uma determinada OFOL se e somente se existe pelo menos um par de pontos (p<sub>i</sub>, p<sub>i+1</sub>) para o qual f(p<sub>i</sub>)=f(p<sub>i+1</sub>).
- 7.3 TEOREMA: Se f é uma OFOL de ordem de grandeza 1 definida so bre uma área A e  $G(f,A)=(G_1,G_2,\ldots,G_m)$  é uma malha determinada por f em A, em que o número de pontos em  $G_i$  é  $n_i$ ,

 $i=1,2,\ldots,m$ , então existem exatamente  $n_1!$   $n_2!$   $\ldots n_m!$  or de nações de A por f.

PROVA: Por definição G(f,A) é uma malha em A por f tal que cada elemento da malha contém pontos para os quais  $f(p_i) = f(p_{i+1})$ , isto é, f tem o mesmo valor para todos os pontos de um elemento. Se um determinado elemento  $G_k$  da malha contém  $n_k$  pontos então existem  $n_k$ ! permutações possíveis desses pontos, isto é, permutações de  $n_k$  elementos  $n_k$  a  $n_k$ . O número total de ordenações de A por f é, assim, o produtório do número de ordenações, de cada elemento da malha ou seja  $n_1$ !  $n_2$ :... $n_m$ ! CQD

7.4 COROLARIO: Existe no máximo uma ordenação estrita de uma área A por uma OFOL f.

PROVA: Pela definição de ordenação estrita segue-se que não existe nenhum ponto A para o qual  $f(p_i)=f(p_{i+1})$ . Assim G(f,A) particiona A em áreas, cada uma das quais contêm exatamente 1 ponto. Daí  $n_1=n_2=\ldots=n_m=1$ . Então  $n_1!$   $n_2!$   $\ldots$   $n_m!$  = 1 CQD

## 8- OPERAÇÕES SOBRE FOL's

- 8.1 DEFINIÇÃO: Concatenação (é) é uma operação binária com propriedades que
  - a) Se  $f_1 \notin f_2 = f_3$  então para todo L,  $f_3(L)$  é o par  $(f_1(L), f_2(L);$
  - b) Se f<sub>1</sub> & (f<sub>2</sub> & f<sub>3</sub>)=f<sub>4</sub> então f<sub>4</sub>(L) e a terna (f<sub>1</sub>(L), f<sub>2</sub>(L), f<sub>3</sub>(L));
  - c)  $f_1 \notin (f_2 \notin f_3) = (f_1 \notin f_2) \notin f_3 = f_1 \notin f_2 \notin f_3$ ;
  - d) Em geral  $f_1 \notin f_2 \neq f_2 \notin f_1$

A função concatenação  $\tilde{\mathbf{e}}$  usada para justapor valores de propriedades distintas.

8.2 DEFINIÇÃO: Adição (+) é uma operação binária tal que f<sub>1</sub>+f<sub>2</sub> é uma OFOL Real cujos valores são determinados pela tabela a seguir:

f <sub>1</sub>	Ω	θ	R <sub>2</sub>	A 2
Ω	Ω	Ω	Ω	Ω
Θ	Ω	ė	θ	Ω
R <sub>1</sub>	Ω	θ	R <sub>1</sub> +R <sub>2</sub>	Ω
A <sub>1</sub>	Ω	Ω	Ω	Ω

em que  $R_1$  e  $R_2$  são números Reais quaisquer,  $A_1$  e  $A_2$  são quaisquer outros valores.

8.3 DEFINIÇÃO: Multiplicação (\*) é uma operação binária tal que  $f_1*f_2$  é uma OFOL Real cujos valores são os da tabela:

f <sub>1</sub>	Ω	θ	R <sub>2</sub>	A 2
Ω	Ω	Ω	Ω	Ω
θ	Ω	θ	· <del>0</del>	Ω
R	Ω	θ	R <sub>1</sub> *R <sub>2</sub>	Ω
A <sub>1</sub>	Ω	Ω	Ω	Ω

em que  $R_1$  e  $R_2$  são números Reais quaisquer,  $A_1$  e  $A_2$  são quaisquer outros valores.

8.4 DEFINIÇÃO: Divisão (/) é uma operação binária tal que f<sub>1</sub>/f<sub>2</sub> é uma OFOL Real cujos valores são determinados pela tabela abaixo:

f <sub>1</sub>	Ω "	θ	0	R <sub>2</sub>	A 2
Ω	Ω	Ω	Ω	Ω	Ω
θ .	Ω	θ	΄ Ω	θ	Ω
0	Ω	θ	Ω	. 0	Ω
R <sub>1</sub>	Ω	. 6	Ω	$(R_1/R_2)$	Ω
A	Ω	Ω	Ω	Ω	Ω

em que R<sub>1</sub> e R<sub>2</sub> são números Reais quaisquer ≠s de 0, A<sub>1</sub> e A<sub>2</sub> são quaisquer outros valores.

8.5 DEFINIÇÃO: Simetrização (-) é um operador unitário tal que -f é uma OFOL Real cujos valores são determinados pela tabela a seguir:

f	Ω	θ	0	R	A
-f	Ω	8	0	-R	Ω

em que R é um número Real qualquer # 0 A é qualquer outro valor.

8.6 DEFINIÇÃO: OU Lógico (\*) à uma operação binária tal que f<sub>1</sub> \* f<sub>2</sub> à uma OFOL Boolsana cujos valores são determinados pela seguinte tabela:

f <sub>1</sub>	Ω	P	Ө	V	A <sub>2</sub>
ß	Ω	Ω	Ω	Ω	Ω
F.	ជ	F	· 0	<b>V</b>	Ω
8	Ω	0	0	V	Ω
V.	Ω	٧	V	V	ប
A <sub>1</sub>	ប	Ω	Ω	ß	ß

em que F e V são os valores "falso" e "verdadeiro", respe<u>c</u> tivamente e

A<sub>1</sub> e A<sub>2</sub> são quaisquer outros valores.

8.7 DEFINIÇÃO: E Lógico (@) é uma operação binária tal que  $f_1$ @  $f_2$  é uma OFOL Booleana cujos valores são determinados, pela seguinte tabela:

1, 12	ល	7	0	V	A <sub>2</sub>
Ω	Ç	Ω	Ω	Ω	Ω
7	Ω	F	F	F	Ω
. •	ត .	F	0	8	ស
V	Ω	F	0	V	Ω
A <sub>1</sub>	ន	Ω	Ω	Ω	Ω

em que F e V são os valores "falso" e "verdadeiro", respectivamente e

A<sub>1</sub> e A<sub>2</sub> são quaisquer outros valores.

8.8 DEFINIÇÃO: Não Lógico (¬) e um operador unitário tal que 
¬f e uma OFOL Booleana cujos valores são determinados, pela tabela a seguir:

£	Ω	F	θ	V	A
٦f	Ω	V	θ	F	Ω

em que F e V são os valores "falso" e "verdadeiro", respectivamente e

A e qualquer outro valor.

8.9 DEFINIÇÃO: Igual Lógico (=) é uma operação binária tal que f<sub>1</sub> = f<sub>2</sub> é uma OFOL de Seleção cujos valores são determinados, pela seguinte tabela:

	f <sub>1</sub> = f <sub>2</sub>
v <sub>1</sub> = v <sub>2</sub>	V
$v_1 \neq v_2$	F

em que v<sub>1</sub> e v<sub>2</sub> são os valores de f<sub>1</sub> e f<sub>2</sub> respectivamente.

8.10 DEFINIÇÃO: Menor que Lógico (<) é uma operação binária tal que  $f_1$  <  $f_2$  é uma OFOL de Seleção cujos valores são deternados, pela tabela a seguir:

	f <sub>1</sub> < f <sub>2</sub>
v <sub>1</sub> < v <sub>2</sub>	V
Qualquer outra condição	F

em que v<sub>1</sub> e v<sub>2</sub> são os valores de f<sub>1</sub> e f<sub>2</sub> respectivamente.

8.11 DEFINIÇÃO: Se-Senão Lógico (+,+) é uma operação ternária tal que f<sub>1</sub> + f<sub>2</sub> + f<sub>3</sub> é uma FOL cujos valores são determina dos.pela tabela a seguir:

f <sub>2</sub>	f <sub>1</sub> + f <sub>2</sub> + f <sub>3</sub>
v	v <sub>1</sub>
F	v <sub>3</sub>
Θ	θ
Qualquer outra	Ω

em que v<sub>1</sub> e o valor de f<sub>1</sub> e v<sub>3</sub> e o valor de f<sub>3</sub>

## 9- CONSTRUÇÕES E EXTENSÕES DA ALGEBRA

9.1 DEFINIÇÃO: Um Argumento de um Feixe (A.) é uma das áreas do Conjunto de Áreas o sobre o qual o feixe B

 $B = B(b, \phi)$ 

é definido.

9.2 DEFINIÇÃO: Se F é uma função de um feixe e L é uma das linhas do feixe

p<sub>F</sub>(L)

e o ponto associado a L por F.

- 9.3 DEFINIÇÃO: A Interseção  $I_i(B,A_i) = B_n A_i$ , de um feixe B com seu i-ésimo argumento  $A_i$  é o conjunto de todos os pontos de  $A_i$  que pertencem a alguma das linhas L do feixe B.
- 9.4 TEOREMA: Para um determinado feixe B sobre um conjunto de áreas  $\phi$  e um determinado argumento  $A_i$  desse feixe, existe uma FOB  $F_i$  tal que  $I_i(B,A_i) = F_i(B)$ .

PROVA: Consideremos  $F_i = \{q_j^i = q_{ij}, j=1,2,...,m\}$ . Então  $F_i$  associa à linha L do feixe B, o ponto  $p_F(L)$  que  $\tilde{e}$  a interseção de L com  $A_i$ . CQD.

9.5 TEOREMA: Para todo o conjunto de Áreas  $\phi$  existe, para cada área  $A_i$  do conjunto, um feixe B sobre  $\phi$  e uma FOB  $F_i$  tal que  $A_i$  =  $F_i(B)$ .

PROVA: Tomemos como função b que determina o feixe aquela que tem valor verdadeiro para todas as linhas possíveis. Cada ponto de A, está, então, contido em alguma linha desse feixe. Tomemos a função F, definida por:

$$F_{i} \equiv \{q_{i}^{t} = q_{ij}^{t}, j = 1, 2, ..., m\}$$

Então,

$$I_i(B,A_i) = A_i = F_i(B)$$
 CQD

- 9.6 DEFINIÇÃO: A Área de um Feixe A(B) é o conjunto de todos os pontos, cada um dos quais está contido, em alguma linha do Feixe.
- A(B) é a união de todas as interseções I, de B com cada área A. Cada I. é definida por uma FOB F. (B). Assim, a Área de um feixe é uma construção baseada no feixe (bundle construct) que pode ser assim expressa:

$$A(B) = F_1(B) U F_2(B) U ... U F_n(B)$$

9.7 TEOREMA: Para todo Conjunto de Áreas  $\phi = (A_1, A_2, ..., A_n)$  existe um feixe B tal que  $A(B) = A_1 \cup A_2 \cup A_3 \cup ... \cup A_n$ .

PROVA: Tomemos como feixe sobre  $\phi$  aquele que consiste de todas as linhas possíveis e como FOBs as  $F_i$  definidas na prova do Teorema 4. Então  $F_i$  (B) =  $A_i$  para i=1,2,...,n CQD

9.8 NOTAÇÃO: O conjunto de pontos de A, que não pertence a I, é o complemento de I, com relação a A, e será notado I¦(A,) enquanto que o complemento de I, com relação a todo o espaço P será notado I! . Segue-se daí que

$$I_{i}^{!}(A_{i}) = A_{i} \cap (I_{i}^{!})$$

9.9 DEFINIÇÃO: Dois conjuntos de Áreas  $\phi_1 = (A_{11}, A_{12}, \dots, A_{1n})$  e  $\phi_2 = (A_{21}, A_{22}, \dots, A_{2n})$  em que cada área de um conjunto pode ser expressa por uma FOB do outro Conjunto, são chamados Feixe-equivalentes.

#### 10 - CLOSSÁRIO

#### Termos e Símbolos em Ordem Alfabática

A Irea (Area)	4.1	G Malha (Glump)	6.1
a Valor particular (particular value)	2.8	g Função que determina a malha	
A(B) Area de um Feixe (Area of a Bundle)	9.6	(Glumping function)	6.1
A Argumento de um Feixe (Bundle argument)	9.1	G Elemento de uma malha	•
ADIÇÃO (ADDITION) (+)	8.2	(Element of a glump)	6.1
Area (Area)(A)	4.1	H	6.4
Area de um feixe (Area of a Bundle) A(B)	9.6	I <sub>l</sub> Intersação do Feixe com seu i-ésimo	
Argumento de um Paixe		Argumento (Intersection of Bundle	
(Bundle argument) (A <sub>ξ</sub> )	9.1	with Bundle Argument)	9.3
B Peixe (Bundle)	5.2	$I_i$ , Complemento de $I_i$ (Complement of $I_i$ )	9.8
b Função que determina o faixe (bundling		IGUAL LÓGICO (EQUALS) (*)	8.9
function)	5.2	Imagem de una FOA (Value set of an FOA)	4.2
Complemento de $I_i$ (Complement of $I_i$ ) $(I_i^*)$	9.8	Imagem de una POL (Value set of an POL)	3.3
CONCATENAÇÃO (GONCATENATION) (4)	8.1	Interseção do Feixe com seu i-ésimo Argu	
Conjunto de áreas (Area Set) (†)	5.1	mento (Intersection of Bundle with	
Conjunto de Coordenadas		Bundle Argument) (I4)	9.3
(Coordinate Set) (Q)	2.5	L Linha (Line)	3.1
Conjunto de valores de Propriedades	•	Linha (Line) (L)	3.1
(Property Value Set) (Y)	2.2	Halha (Glump) (G)	4.1
d Ponto Representativo (Datum Point)	2.9	MENOR-QUE LÓGICO (LESS THAN) (<)	8;-10
Desconhecido (missing) (6)	2.3	MULTIPLICAÇÃO (MULTIPLICATION) (4)	. 8.3
Discriminatório (discriminatory)	2.11	Não Definido (Undefined) (Q)	2.3
DIVISÃO (DIVISION) (/)	8:4	NÃO LÓGICO (NOT) (¬)	8.8
e Entidade (Entity)	2.1	O Ordenação (Ordering)	7.1
E LÓGICO (AND) (@ )	8.7	OFOL FOL de Orden (Ordinal FOL)	3.9
Elemento de uma malha (element of a		OFOL de Seleção (Selection OFOL)	3.9
glump) (G <sub>s</sub> )	6.1	Ordem de grandeza de uma FOL	
Entidade (Entity) (e)	2,1	(Span of an FOL)	3.8
Espaço de propriedades		Orden de grandesa de uma linha	
(Property Space) (P)	2.8	(Span of a line)	3.1
P e V Valores Booleanos Falso e Verdadeire		Ordenação (Ordering) (O)	7.1
(The Boolean values False and True)	•	OU LÉGICO (OR) (0)	8.6
Peixe (Bundle) (B)	5.2	P Espaço de propriedades (Property Space)	2.8
Peixe-equivalente (Bundle-equivalent)	9.9	p Ponto (Point)	2.8
FOA Função de Áreas (Function of an Area)	4.2	pp Ponto associado a L por F	,
POB Função de um feixe		(Point assigned to L by F)	9.2
(Function of a Bundle)	5.3	Ponto (point) (p)	2.8
FOG Função de uma malha		Ponto associado a L por Y	
(Punction of a Glump)	6.3	(Point assigned to L by P) (pp)	9.2
FOL Função de linhes (Function of a Line)	3.3	Ponto Representativo (Datum Point) (d)	2.9
FOL de Ordem (Ordinal FOL) OFOL	3.9	Propriedade (Property) (q)	2.1
Punção de Áreas (Function of an Area) FOA	4.2	Q Conjunto de coordenadas (Coordinate Set)	
Função de Linhas (Function of a Line) FOL	3.3	q Propriedade (Property)	2.1
Punção da Salação (Salaction Function) s	5.7	R Pasigna um número real	• • •
Função de um Feixe	311	(designates a real number)	
(Function of a Bundle) FOB		Relação (Relational operator) (8)	A
Função de uma malha	5.3	s Função de Saleção (Selection function)	3.9
(Function of a Glump) FOG		SE-SENKO LÖGICO (IP-OTHERWISE) (+,+)	5.7
Função que determina a malha	6.3	a.a. * aa.a	8.11
(Glumping function) (g)			9, 5.7
	6.1	SINETRIZAÇÃO (NEGATION) (-)	8.5
Função que determina o Feixe		V Conjunto de valores de propriedade	
(Bundling function) (b)	5.2	(Property Value Set)	2.2

v Va	lor (Value)	2.1	1 Valores Booleanos Falso e Verdadeiro	
Y e F Valores Booleanos Verdadeiro e Fal-			(The Boolean Values False and True) (F	e V)
8.0	(The Boolean values True and False)		Valor Particular	
Valor	(Yalue) p	2.1	(Particular value) a	2.8

#### Simbolos Não Alfahéticos

(Com exceção dos quatro últimos, os demais são de operações sobre FOL'S)

+ ADICÃO (ADDITION)	8.2	* MULTIPLICAÇÃO (MULTIPLICATION)	8.3
@ E LOGICO (AND)	8.7	¬ NÃO LÓCICO (NOT)	8.8
d CONCATENAÇÃO (CONCATENATION)	· 8.1	● OU LÓGICO (OR)	. 8.6
/ DIVISÃO (DIVISION)	8.4	♦ Conjunto de Áreas (Area Set)	5.1
- IGUAL LÓGICO (EQUALS)	8.9	0 Desconhecido (missing)	2.3
+,+ SE-SENÃO LÓGICO (IT-OTHERWISE)	8.11	Ω NÃO DEFINIDO (UNDEFINED)	2.3
< MENOR-OUR LÓGICO (LESS THAN)	8.10		

### APÊNDICE II

DOCUMENTAÇÃO DO PROGRAMA ELABORADO PARA A COMPARAÇÃO DAS SOLUÇÕES DADAS AO PROBLEMA DE DETERMINAÇÃO DOS PONTOS DE SELA DE MATRIZES

#### II.1 - PROGRAMA

O programa principal limita-se a ler a matriz de cartões de dados, armazena-la em disco e chamar cada uma das subrotinas elaboradas com base nas 3 soluções a serem comparadas. Tanto o programa como as subrotinas foram programados em linguagem FORTRAN do computador IBM1130.

O sistema adotado para o armazenamento da matriz, no disco, foi de driação de um arquivo na Working Storage (WS) com 80 registros, cada um de 80 palavras, de modo a armazenar cada linha da matriz em um registro desse arquivo.

A listagem do deck desse programa é apresentada abaixo.

. LISTAGEN DO DECK ..

```
FOR ATTECHN TO THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF THE PROPERTY OF
```

### II.2 - DADOS

Nas paginas seguintes estão listados os cartões de dados, nos quais estão perfurados os elementos das quatro matrizes utilizadas no esquema de comparação das soluções.

### II.3 - SUBROTINAS

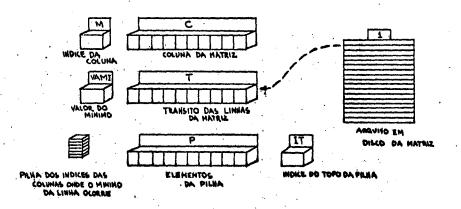
Nas demais paginas deste apendice, são apresentados os seguintes elementos, em relação a cada uma das 3 subrotinas:

Esquema de armazenamento da informação. Fluxograma geral. Fluxograma detalhado. Programação em Linguagem FORTRAN.

1 11 111	
	011111111111111111111111111111111111111
1	and and produced and and and produced and and produced as
1 1 1 1 1 1 1 1	
	ALLEGE THE STATE OF THE STATE O
11 TIOILIII LILLI	إنجر أحد أجو أجو أجو أحد أحد أحد أحد أحد أحد أحد أحد أحد أحد
	I THE TAXABLE PROPERTY OF THE
	HILL HILL HILL
1111	al and and and and and and and and and and
	MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA MANAGAMANA M
LIOIL LILLILL IL	
id limite the same	HALLINIALIANIANIALIANIA
	delled all and the second of t
111111111111111111111111111111111111111	HALLANDER STANDER STANDER STANDER STANDER STANDER STANDER STANDER STANDERS
	reduced and and and and and and and and and an
1 11 111 1111 11	lilliandididididocoonadadadadadadadadadadadadadadadadadada
1 1111111111111111111111111111111111111	
11 11 11 11 11 11 11 11 11 11 11 11 11	
	te de la la la la la la la la la la la la la
1 1 1 1 1 1 1 1 1	1
11 1111111	
11 11111111111	
	indicated and the state of the
	والمراجع المراجع المرا
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	desired the desired the second
111 1 1111 1111	
	rendered and and and and and
	leading little still telling
	LILLIAN TERRETARIA TER
111111111111111111111111111111111111111	
111111111111111111111111111111111111111	
12111	lilli illi illi il il
1 1 1 1 1 1 1	
	. 1
1	1 1111
111111	enviolatelentation as er
	a inspirational and and and and and and and and and and
	11111111111111111111111111111111111111
1111	ii i iiiiii iiiii iiii iiii iiiiiiiiii
	HIMMENDALIA MANAGEMENT AND AND AND AND AND AND AND AND AND AND
1 1 1 1	
1 11 1 111111	THE PROPERTY IN THE PROPERTY IN THE PROPERTY OF THE PROPERTY O
	THE PERMIT THE PERMIT PRINCES OF THE PERMIT

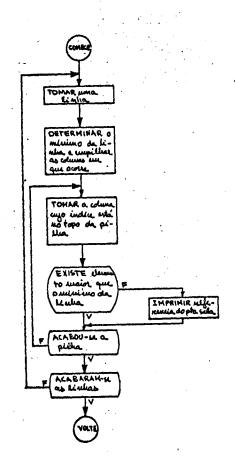
101211031102222101011021243310223123422002412344123212414141421231433131334224

<u>ADAEFGYJHBCOEGHFĞHFŸĞFĞHÖTÖAECDEAGHİJKBCÖEHDEJIHJIKABCDEAHSHJGHJĞHJJGGHGHGDAEÇAA</u>

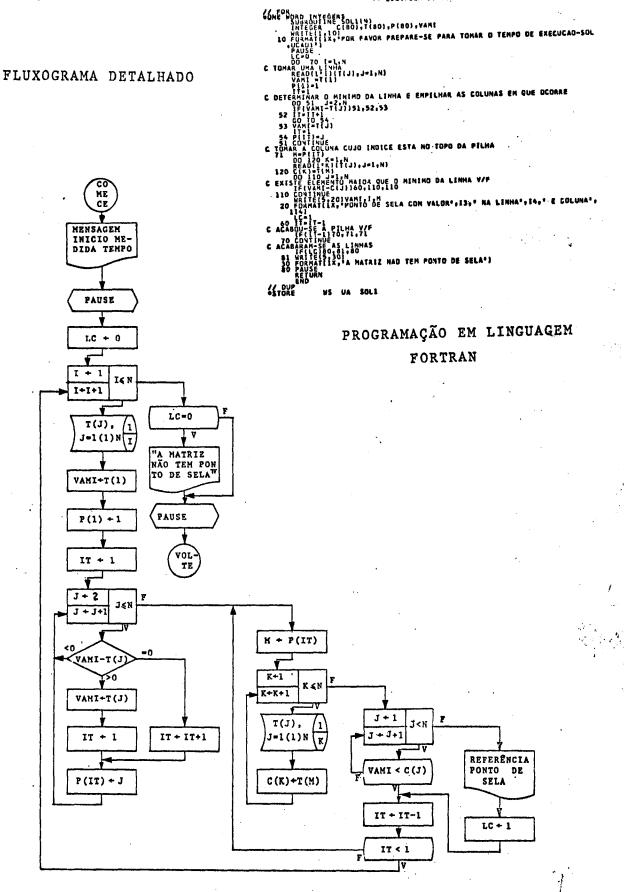


# ESQUEMA DE ARMAZENAMENTO DA INFORMAÇÃO

## FLUXOGRAMA GERAL

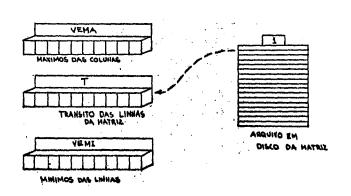


PAGINA GOL

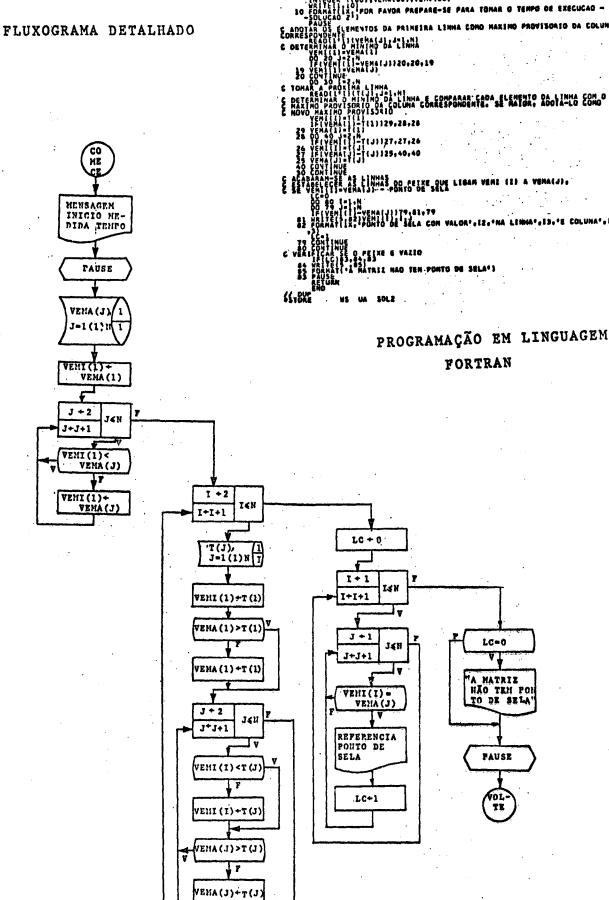


#### FLUXOGRAMA GERAL





ESQUEMA DE ARMAZENAMENTO
DA INFORMAÇÃO



### FLUXOGRAMA GERAL

