UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E CIÊNCIA
DA COMPUTAÇÃO

LAD - LINGUAGEM PARA ANÁLISE DIMENSIONAL

JOSÉ OSVALDO FERRARI

Orientador:
Prof. Dr. NELSON CASTRO MACHADO

Dissertação apresentada ao Instituto de Matemática, Estatística e Ciência da Computação, para obtenção do título de "Mestre"

CAMPINAS
DEZEMBRO DE 1973

UNICAMP
BIBLIOTECA CENTRAL

À VANDA

Minha esposa

AGRADECIMENTOS

A todos que direta ou indiretamente contribuiram para a conclusão deste trabalho expresso minha gratidão.

Agradecimentos especiais se devem:

Ao Prof. Dr. Jacques Cohen, da Universidade de Brandeis, Estados Unidos, pela proposição, apoio e incentivo, fatores marcantes para a realização deste trabalho.

Ao Prof. Dr. Nelson Castro Machado, orientador deste trabalho, pela orientação e paciente leitura do manuscrito dando-lhe a necessária coerência e forma apresentável.

Aos amigos e colegas do Instituto e em especial aos colegas de Departamento Tadao, Valente e Vanini, pelas críticas, oportunas sugestões e valiosa colaboração que dispensaram a este trabalho.

SINÓPSE

O presente trabalho descreve o Sistema LAD (Linguagem para Análise Dimensional) destinado à solução de problemas ligados à Análise Dimensional através da utilização de um computador digital de grande porte em "time-sharing".

Problemas envolvendo conversão de unidades ou cálculo de relações entre grandezas, podem ser rapidamente resolvidos em modo conversacional pelo sistema que fornece ainda opções para: saída de resultados em vários formatos, listagem de programas fonte ou armazenamento de programas para uso futuro. Os comandos, cuja forma é bastante próxima à notação usada em Análise Dimensional, são poucos e simples, tornando extremamente fácil seu aprendizado e utilização por pessoas não especializadas em computação.

ÍNDICE

CAPÍTULO 1

INTRODUÇÃO 1

CAPÍTULO 2

FUNDAMENTOS DE ANÁLISE DIMENSIONAL

2.1 - Considerações Iniciais 3
2.2 - Introdução Histórica 3
2.3 - Sistemas de Unidades e Medidas 4
2.3.1 - Grandezas, Unidades e Medidas ... 4
2.3.2 - Sistema de Unidades 5
2.4 - Equações Dimensionais 6
2.5 - Conversão de Unidades 7
2.6 - Análise Dimensional 8
2.6.1 - Conjunto Completo de Produtos Adimensionais 8
2.6.2 - Número de Produtos Adimensionais. 8
2.6.3 - Cálculo de Produtos Adimensionais 11
2.6.4 - Disposição das Variáveis 12

CAPÍTULO 3

O SISTEMA LAD

3.1 - Descrição Geral 13
3.2 - Comandos do Interpretador LAD 14
3.2.1 - Regras Gerais 14
3.2.2 - Comandos do Interpretador 16
3.2.2.1 - Declaração de Grandezas e Unidades Básicas 16
3.2.2.2 - Comando Aritmético 16
3.2.2.3 - Comando de Cálculo de

Produtos Adimensionais. 17
3.2.2.4 - Comando Comentário 17
3.2.2.5 - Comando de Impressão ou Saída 17
3.3 - Erros 19

CAPÍTULO 4

UTILIZAÇÃO DO SISTEMA LAD
4.1 - Entrada e Saída do Sistema 21
4.1.1 - Acesso ao Sistema 21
4.1.2 - Supervisor 21
4.1.3 - Interpretador 23
4.1.4 - Fluxograma do Sistema 23
4.2 - Exemplos 25

CAPÍTULO 5

IMPLEMENTAÇÃO DO SISTEMA LAD
5.1 - Observações Gerais 28
5.2 - Esquema Básico 28
5.3 - Rotinas Geradas Automaticamente 30
5.3.1 - Programa Principal 30
5.3.2 - Rotina de Análise Léxica 30
5.3.3 - Rotina de Análise Sintática 31
5.4 - Rotina de Ações Semânticas 31
5.4.1 - Descrição Geral 31
5.4.2 - Estrutura de Dados de LADAC 32
5.4.3 - Descrição das Ações Semânticas .. 34
5.5 - Rotina de Erros 36
5.6 - Performance 36

CAPÍTULO 6

CONCLUSÕES 37

BIBLIOGRAFIA 39

APÊNDICES

1 - Grandezas e Suas Dimensões 40
2 - Fatores de Conversão de Unidades 41
3 - BNF do Interpretador LAD 42
4 - Exemplos 43

CAPÍTULO 1

INTRODUÇÃO

A Análise Dimensional é uma técnica largamente empregada por ciências experimentais, principalmente em Física e em Engenharia dentro da Teoria dos Modelos, para verificação de leis, resultados de observações e mesmo suprindo a necessidade de experiências práticas. |5|

De uma forma geral, a Análise Dimensional envolve dois tipos de problemas.

O primeiro tipo trata da questão de mudança de unidades. Este problema pode ser enunciado da seguinte forma: Dada a medida de uma grandeza em um sistema de unidades S_1, qual seria a medida da mesma grandeza no sistema de unidades S_2? Este problema ocorre frequentemente em Física, Engenharia, etc.. |6| |8|

O segundo tipo de problema a considerar é a obtenção de relações entre as várias grandezas que influem em um determinado fenômeno físico. O problema se resume no cálculo dos produtos chamados adimensionais, os quais, aplicados à Teoria dos Modelos, constituem um método teórico que possibilita ensaios de laboratório com modelos.|5||7|

Nos dois problemas acima enunciados, os cálculos envolvidos, embora não particularmente complicados, são muitas vezes bastante trabalhosos. Considerando a frequência com que pesquisadores se veem às voltas com esses cálculos, é bastante desejável a criação de um método automático para executá-los.

O trabalho aqui apresentado introduz um sistema de programação para atender a estas necessidades: o

Sistema LAD (Linguagem para Análise Dimensional).

Este sistema consiste basicamente em um programa que aceita como entrada um conjunto de comandos adequados e, através da análise individual desses comandos, executa uma série de cálculos específicos em análise dimensional, comunicando ao usuário os resultados e eventuais situações de erro. O sistema oferece além destas funções básicas, comandos adicionais para a obtenção de informações úteis e escolha de opções. Assim, é possivel "pedir" listagem de programas e informações sobre tipo de execução e tempo de processamento.

Uma característica fundamental do Sistema LAD é a possibilidade de operação de modo conversacional, com o que o usuário mantém um verdadeiro diálogo com o sistema, fazendo perguntas e recebendo respostas.

No capítulo 2 são apresentados fundamentos de Análise Dimensional. O capítulo 3 consiste de uma descrição geral do Sistema LAD e seus comandos, seguido no capítulo 4 pela apresentação e descrição de exemplos, dando uma visão das facilidades oferecidas. A implementação do Sistema LAD é descrita no capítulo 5.

Ao leitor que se interessa somente pelo uso do Sistema LAD, tendo conhecimentos de Análise Dimensional é suficiente a leitura dos capítulos 3 e 4.

CAPÍTULO 2

FUNDAMENTOS DE ANÁLISE DIMENSIONAL

2.1 - Considerações Iniciais.

Usuários não familiarizados com as notações e princípios básicos da Análise Dimensional podem, através deste capítulo, de uma maneira rápida, adquirir informações suficientes para acompanhar o desenvolvimento dos demais capítulos.

Serão vistos teoremas, definições e métodos de cálculos, cujas deduções não serão apresentadas uma vez que somente os resultados interessam ao trabalho.

2.2 - Introdução Histórica.

A Análise Dimensional é uma teoria que começou a ser formalmente definida no fim do século XIX.

O princípio básico no qual se apoia toda a teoria é o princípio da homogeneidade dimensional, que pode ser assim formulado:

> "Uma condição necessária para que uma equação física seja verdadeira é que ela seja dimensionalmente homogênea."

Apoiado neste princípio E. Buckingham enunciou um teorema que sumariza toda a Análise Dimensional:

> "Se uma equação é dimensionalmente homogênea ela pode ser reduzida a uma relação entre produtos adimensionais que formam um conjunto completo."

Buckingham e outros estudiosos estabeleceram uma regra sôbre o número de produtos adimensionais que em 1922 foi apontada por P. W. Bridgman como incompleta.

Em 1946 E. R. Van Driest formulou a regra completa:

> "O número de produtos adimensionais em um conjunto completo é igual ao número total de variáveis menos o máximo número das variáveis que não formam um produto adimensional."

Com o princípio da homogeneidade dimensional, o teorema de Buckingham e a regra de Van Driest estava formulada a Teoria da Análise Dimensional. |5|

2.3 - Sistemas de Unidades e Medidas.

2.3.1 - Grandezas, Unidades e Medidas.

As mais importantes leis que regem os fenômenos naturais são leis quantitativas, isto é, são leis que podem ser descritas por fórmulas matemáticas que contem símbolos representativos das grandezas que influem naquele fenômeno.

Definição 1 - Grandeza é um parâmetro ou propriedade que caracteriza um ente físico.

Uma grandeza é dada em função de dois parâmetros: uma medida e uma unidade.

Definição 2 - Unidade de uma grandeza é uma quantidade arbitrária que se toma para termo de comparação entre grandezas de mesma espécie.

Definição 3 - Medida de uma grandeza é um número puro que expressa a comparação entre uma certa

extensão da grandeza com sua unidade.

Portanto uma grandeza G com medida m e unidade U pode ser expressa pela equação:

$$G = m \cdot U \quad (1) \quad |5|$$

2.3.2 - Sistema de Unidades.

Definição 4 - Sistema de Unidades é um conjunto de unidades tomadas em número suficiente para medir todas as grandezas que figuram no campo de ciência que se deve trabalhar.

Um sistema de unidades contem duas espécies de unidades: unidades básicas ou fundamentais e unidades derivadas.

Definição 5 - Uma unidade é dita básica ou fundamental se ela independe de qualquer outra.

Definição 6 - A unidade é derivada se depende de outras, sejam elas básicas ou derivadas.

Em um sistema de unidades as grandezas de mesma espécie que as unidades básicas são as grandezas básicas.

Para a construção de um sistema de unidades escolhem-se as grandezas e unidades básicas e a partir delas constroem-se as grandezas e unidades derivadas.

Um sistema construido como acima descrito é chamado Sistema Coerente de Unidades.

Com um mesmo conjunto de grandezas básicas pode-se construir um número infinito de sistemas, pois a escolha da extensão da unidade de uma grandeza é arbitrária. |6|

2.4 - Equações Dimensionais.

Seja a equação (1) da seção 2.3 :

$$G = m \cdot U$$

Teorema 1 - A razão entre as medidas de uma mesma grandeza com unidades diferentes é igual ao inverso da razão entre essas unidades.

Sejam m_o e U_o medida e unidade antigas de G e m_1 e U_1 medida e unidade novas da mesma grandeza G. Tem-se que:

$$G = m_o \cdot U_o$$
$$G = m_1 \cdot U_1$$

ou ainda:

$$m_1 = m_o \cdot (U_o/U_1) \qquad (2)$$

A equação (2) expressa o que foi dito no teorema 1.

Definição 7 - Chama-se símbolo dimensional da grandeza G, e é indicado por $|G|$, à relação dada por (U_o/U_1).

Note-se que o símbolo dimensional não depende das unidades. |6|

Facilmente se vê que o símbolo dimensional de um número é 1.

Definição 8 - Equação dimensional é uma equação entre símbolos dimensionais.

Diz-se que uma grandeza G tem dimensões i em relação a uma grandeza X, j em relação a uma grandeza Y, k em relação a uma grandeza Z,..., se a equação dimensional de

G em relação a X,Y,Z,..., for a seguinte:

$$|G| = |X|^i . |Y|^j . |Z|^k ...$$

Definição 9 - Uma equação é dita <u>dimensionalmente homogênea</u> se sua forma não depende das unidades fundamentais de medida. |5|

Uma equação da forma:

$$x = a + b + c + ...$$

é dimensionalmente homogênea se x,a,b,c,.. tem as mesmas dimensões em relação aos símbolos dimensionais das grandezas básicas.

O princípio da homogeneidade dimensional baseia-se na definição 9.

2.5 - <u>Conversão de Unidades</u>.

O teorema 1, da seção 2.4, dá a formula geral da conversão de unidades, através da equação (2):

$$m_1 = m_o . (U_o/U_1)$$

Se uma grandeza G tem medida m_o em relação à unidade U_o, a mesma grandeza pode ser expressa em termos da medida m_1 em relação à unidade U_1 da seguinte forma:

$$G = m_o . U_o = m_o . (U_o/U_1) . U_1 = m_1 . U_1$$

A partir de m_o e da relação U_o/U_1, a mudança de unidades como acima calculado é simples. Se as unidades U_o e U_1 não são básicas, pode-se escrevê-las em função das básicas e o resultado será o mesmo. |6|

2.6 - Análise Dimensional.

2.6.1 - Conjunto Completo de Produtos Adimensionais.

Definição 10 - Produto adimensional é uma expressão que tem dimensões nulas em relação aos símbolos dimensionais das grandezas básicas.

Assim sendo, qualquer equação que seja dimensionalmente homogênea pode formar um produto adimensional. O nome produto adimensional deriva do fato de que na expressão só comparecem produtos de potências cuja forma geral é:

$$x_1^i \cdot x_2^j \cdot \ldots x_n^k$$

Definição 11 - Um conjunto de produtos adimensionais de grandezas dadas é completo, se cada produto do conjunto é independente dos outros, e qualquer outro produto não do conjunto, das mesmas grandezas, é um produto de potências dos produtos do conjunto.

2.6.2 - Número de Produtos em um Conjunto.

Sejam $G_1, G_2, \ldots, G_m$ as grandezas básicas de um sistema de unidades, e $x_1, x_2, \ldots, x_n$ grandezas deste sistema.

Um produto adimensional entre as grandezas x_i tem a forma:

$$\pi = x_1^{k_1} \cdot x_2^{k_2} \ldots x_n^{k_n} \qquad (3)$$

Pela definição 10, a equação dimensional da expressão acima é:

$$|\pi| = |G_1|^0 \cdot |G_2|^0 \ldots |G_m|^0 = |x_1|^{k_1} \cdot |x_2|^{k_2} \ldots |x_n|^{k_n} \quad (4)$$

As equações dimensionais das grandezas x_i, são:

$$|x_1| = |G_1|^{a_1} \cdot |G_2|^{a_2} \cdots |G_m|^{a_m}$$

$$|x_2| = |G_1|^{b_1} \cdot |G_2|^{b_2} \cdots |G_m|^{b_m}$$

$$\vdots$$

$$|x_n| = |G_1|^{p_1} \cdot |G_2|^{p_2} \cdots |G_m|^{p_m}$$

os quais afetados dos expoentes k_i, de (3), se tornam:

$$|x_1|^{k_1} = |G_1|^{a_1 k_1} \cdot |G_2|^{a_2 k_1} \cdots |G_m|^{a_m k_1}$$

$$|x_2|^{k_2} = |G_1|^{b_1 k_2} \cdot |G_2|^{b_2 k_2} \cdots |G_m|^{b_m k_2} \qquad (5)$$

$$\vdots$$

$$|x_n|^{k_n} = |G_1|^{p_1 k_n} \cdot |G_2|^{p_2 k_n} \cdots |G_m|^{p_m k_n}$$

Substituindo este resultado em (4), obtém-se:

$$|G_1|^0 \cdot |G_2|^0 \cdots |G_m|^0 = |G_1|^{a_1 k_1} \cdot |G_2|^{a_2 k_1} \cdots |G_m|^{a_m k_1} \cdot$$
$$|G_1|^{b_1 k_2} \cdot |G_2|^{b_2 k_2} \cdots |G_m|^{b_m k_2} \cdot$$
$$\vdots$$
$$|G_1|^{p_1 k_n} \cdot |G_2|^{p_2 k_n} \cdots |G_m|^{p_m k_n}$$

Agrupando as potências de mesma base, fica finalmente:

$$|G_1|^0 \cdot |G_2|^0 \cdots |G_m|^0 = |G_1|^{(a_1 k_1 + b_1 k_2 + \cdots + p_1 k_n)} \cdot$$
$$|G_2|^{(a_2 k_1 + b_2 k_2 + \cdots + p_2 k_n)} \cdot$$
$$\vdots$$
$$|G_m|^{(a_m k_1 + b_m k_2 + \cdots + p_m k_n)}$$

Da igualdade das potências tem-se um sistema de equações lineares homogêneas, cujas incógnitas são os k_i:

$$\begin{cases} a_1k_1 + b_1k_2 + \dots + p_1k_n = 0 \\ a_2k_1 + b_2k_2 + \dots + p_2k_n = 0 \\ \qquad\vdots \\ a_mk_1 + b_mk_2 + \dots + p_mk_n = 0 \end{cases} \qquad (I)$$

cuja matriz dos coeficientes é:

$$\begin{pmatrix} a_1 & b_1 & \dots & p_1 \\ a_2 & b_2 & \dots & p_2 \\ \vdots & \vdots & & \vdots \\ a_m & b_m & \dots & p_m \end{pmatrix}$$

Definição 12 - A matriz dos coeficientes do sistema (I) é chamada <u>matriz dimensional</u>.

Definição 13 - Se a matriz dimensional contem um determinante de ordem <u>c</u>, e se todos os determinantes de ordem maior que <u>c</u> contidos pela matriz tiverem o valor zero, então a <u>característica</u> da matriz dimensional é dita ser <u>c</u>.

Teorema 2 - O número de produtos adimensionais em um conjunto completo é igual ao número total de grandezas menos a característica da matriz dimensional formada a partir das equações dimensionais destas grandezas.

Teorema 3 - As linhas da matriz dimensional são linearmente dependentes se e somente se a característica desta matriz é menor que o número de linhas.

O teorema 3 afirma que se todos os determinantes cuja ordem é igual ao número de linhas é zero, então existem linhas que são proporcionais a combinações de outras linhas. Neste caso a matriz dimensional é dita ser <u>singular</u>.

2.6.3 - Cálculo dos Produtos Adimensionais.

Quanto às soluções do sistema (I), pela teoria da álgebra linear, tem-se que:

Teorema 4 - Desprezando a solução trivial ($k_i=0$), o sistema (I) possui exatamente (n-c) soluções linearmente independentes, onde c é a característica da matriz dos coeficientes.

Definição 14 - Um conjunto de (n-c) soluções linearmente independentes do sistema (I) é chamado de sistema fundamental de soluções, e qualquer solução é uma combinação linear de soluções em qualquer sistema fundamental.

Estes princípios algébricos têm um significado direto sobre a análise dimensional, desde que soluções linearmente independentes do sistema (I) fornecem os expoentes das variáveis que comparecem nos produtos adimensionais. Consequentemente qualquer sistema fundamental de soluções fornece um conjunto completo de produtos adimensionais.

Assim para as variáveis k_i do sistema (I) com $1 \leqslant i \leqslant (n-c)$ atribui-se:

$k_1=1$ e para $i \neq 1$ $k_i=0$, obtendo-se um sistema de equações lineares determinado. Pela regra de Cramer calcula-se $k_{j,1}$ onde $(n-c+1) \leqslant j \leqslant n$;

$k_2=1$ e para $i \neq 2$ $k_i=0$, obtendo-se outro sistema de equações lineares determinado que resolvido pela regra de Cramer, dá $k_{j,2}$ com $(n-c+1) \leqslant j \leqslant n$.

Repete-se este processo até resolver os (n-c) sistemas de equações determinados. A matriz solução dará os expoentes das grandezas que comparecem nos produtos adimensionais, os quais são da forma:

$$\Pi_i = x_1^{k_{i,1}} \cdot x_2^{k_{i,2}} \cdots x_n^{k_{i,n}} \quad \text{com} \quad 1 \leqslant i \leqslant (n-c)$$

2.6.4 - Disposição das variáveis.

Desde que cada uma das primeiras (n-c) variáveis da matriz dimensional pode ocorrer em somente um produto adimensional, a seguinte regra pode ser observada:

"Na matriz dimensional, deixe sempre as variáveis independentes nas colunas mais à direita."

Em casos excepcionais esta disposição pode levar a um impasse, porque a matriz dimensional pode não conter um determinante de ordem c diferente de zero nas c colunas mais à direita. As variáveis deverão ser então rearranjadas, sem que se altere a disposição recomendada além do estritamente necessário. [5]

CAPÍTULO 3

O SISTEMA LAD

3.1 - Descrição Geral.

O Sistema LAD é basicamente um programa que aceita como entrada comandos na linguagem LAD, executando cada comando imediatamente após recebido.

Destina-se este sistema basicamente a usuários não versados em Ciência da Computação. Para tanto foram tomados cuidados especiais, de modo a simplificar seu uso e aprendizado. A forma, tanto dos comandos de entrada como dos resultados de saída, é bastante próxima à linguagem natural do usuário, de modo que este não necessite grandes esforços para assimilá-los e entendê-los.

O sistema pode ser utilizado em duas modalidades distintas: com programas armazenados e com programas conversacionais.

Um programa armazenado é um programa previamente montado e armazenado em um equipamento periférico como o disco magnético.

Um programa do tipo conversacional é resolvido de forma iterativa entre o usuário e o computador. Quando informado que se trata de um programa do tipo conversacional, o sistema solicita do usuário comando após comando e os executa à medida que forem sendo recebidos.

Para o desenvolvimento inicial de um programa, a modalidade conversacional é a mais indicada. Se entretanto o usuário desejar armazenar o seu programa para não ter que repetir o diálogo a cada utilização, o sistema ofere

ce a possibilidade de transformação de um modo para outro.

Os comandos são de duas espécies: comandos do sistema operacional ou Supervisor LAD e comandos do Interpretador LAD. O Sistema LAD apresenta, portanto, duas fases de operação: fase de supervisão e fase de interpretação, correspondendo às duas espécies de comando.

Quando em fase de supervisão o sistema recebe informações a respeito de:

- tipo de programa: armazenado ou conversacional;
- listagem de programa: requerida ou não;
- se há mais programas a serem processados.

Na fase de interpretação são dados os comandos do programa, escritos em LAD.

3.2 - Comandos do Interpretador LAD.

3.2.1 - Regras Gerais.

Regra 1 - Início e Término de Programa.
Um programa deve sempre começar com um e um só sinal cifrão ($), e terminar com pelo menos dois destes sinais.

Regra 2 - Separação de Comandos.
Todo comando deve ser separado do comando seguinte por um ponto e vírgula (;), com exceção do último comando, que deve ser separado do duplo cifrão por pelo menos um branco.

Regra 3 - Constantes.
Uma constante pode ser inteira ou real, não sendo admitidos brancos internos.
Constantes inteiras são aquelas que só possuem dígitos do sistema decimal (0 a 9).
Constantes reais apresentam um ponto decimal, ten-

do na parte fracionária pelo menos um dígito.

Regra 4 - Variáveis.
O nome de uma variável pode ser formado de um a cinco caracteres, letras ou dígitos, sendo o primeiro necessariamente uma letra.
Qualquer variável para ser usada deve ser previamente definida.

Regra 5 - Nome de Comando.
Cada nome de comando deve ser escrito entre apóstrofes ('), não sendo permitidos brancos entre os dois apóstrofes.

Regra 6 - Expressão Aritmética.
É uma sequência de constantes e/ou variáveis separadas por operadores aritméticos.
Os operadores aritméticos são os seguintes:

Operadores	Significados
-	menos unário
^	exponenciação
*	multiplicação
/	divisão
+	adição
-	subtração

A precedência dos operadores na tabela acima, cresce no sentido de baixo para cima. Os operadores de multiplicação e divisão, têm a mesma precedência, sendo a indeterminação resolvida da esquerda para a direita. O mesmo sucede entre os operadores de adição e subtração.
Para modificar a ordem de execução de uma expressão aritmética, o uso de parêntesis é permitido, sendo as primeiras operações a serem executadas aquelas que estiverem entre os parêntesis mais internos. |2|

3.2.2 - Comandos do Interpretador.

3.2.2.1 - Declaração de Grandezas e Unidades Básicas.

Este comando declara quais as grandezas e unidades básicas que serão usadas no programa. Sua forma geral é a seguinte:

'DECLARO' : $G_1(U_1), G_2(U_2), \ldots, G_n(U_n)$

onde: G_i são as grandezas básicas;
U_i são as unidades associadas a G_i.
G_i e U_i são nomes de variáveis.

O nome de uma grandeza pode ser o mesmo que o de sua unidade.

3.2.2.2 - Comando Aritmético.

Este comando declara grandezas derivadas ou secundárias. Sua forma geral é a seguinte:

A=E

onde: A é uma variável e expressa o nome de uma grandeza derivada das básicas ou de outra grandeza derivada.
E é uma expressão aritmética.

O sinal de igual (=) significa substituição, não igualdade.

As variáveis (grandezas) pertencentes a E já devem ter sido definidas, ou por um comando de declaração ou por um outro comando aritmético executado anteriormente. |2|

3.2.2.3 - Comando de Cálculo de Produtos Adimensionais.

Para o cálculo dos produtos adimensionais entre várias grandezas utiliza-se o comando FUNCAOPI, cuja forma geral é a seguinte:

'FUNCAOPI' | $V_1, V_2, \ldots, V_n$ |

onde: V_i é uma variável ou expressão aritmética.
Se V_i é uma variável, então é o nome de uma grandeza já definida. Se for uma expressão aritmética, V_i deve estar escrita em função de grandezas já definidas.

As variáveis ou expressões de índice i mais baixos serão consideradas como dependentes das variáveis de índice i mais altos.

3.2.2.4 - Comando Comentário.

Este comando pode ser usado em qualquer ponto do programa e tem a forma geral seguinte:

'COMENTAR' : frase

onde: frase é um conjunto de palavras separadas por brancos. Uma palavra é um conjunto de caracteres alfanuméricos começado por letra e com no máximo 45 caracteres.

O comando comentário não causa ação alguma do sistema a não ser a impressão da frase. É usado para possibilitar a documentação de programas.

3.2.2.5 - Comando de Impressão ou de Saída.

É o comando que possibilita ao usuário a saída de algum resultado ou de alguma matriz montada a partir dos comandos anteriormente descritos. Este comando tem op

ções e sub-opções, e sua forma geral é:

'IMPRIMIR' (OP_1;OP_2;...;OP_n)

onde: OP_i pode ser:

a) Uma expressão aritmética: será impressa a expressão, suas dimensões em função das grandezas e unidades básicas definidas e o seu valor.

b) 'FUNCAO'(V) - onde V é uma grandeza; será impressa uma equação do tipo:

$$V = (exp_i) * FUNCAO$$
$$(PI_1 , PI_2 , \vdots , PI_{i-1} , PI_{i+1} , \vdots , PI_n)$$

sendo $V/exp_i = PI_i$, e sendo os PI's os produtos adimensionais. Note-se que V deve pertencer a PI_i.

c) 'MATDEF' - será impressa uma matriz que contém as grandezas e unidades básicas e as grandezas derivadas, juntamente com os seus valores (medidas). Esta matriz tem o nome de matriz de definição.

d) 'MATDIM' - será impressa uma matriz que contém as dimensões das variáveis ou expressões aritméticas presentes no comando de cálculo de produtos adimensionais. O nome desta matriz é matriz dimensional total.

e) 'MATDIMP' - será impressa a matriz dimensional total se esta for não singular, ou será impressa a matriz dimensional total sem a(s) linha(s) eliminada(s), se esta for singular. Esta matriz é chamada

matriz dimensional parcial.

f) 'MATADIM' - será impressa a matriz dos expoentes das variáveis ou expressões aritméticas que figuram nos produtos adimensionais. É a matriz adimensional.

g) Sub-opções - se, no comando de cálculo de produtos adimensionais, V_i for uma expressão aritmética, nos itens d e e acima, pode-se utilizar a sub-opção 'OP' para que sejam impressas as expressões aritméticas correspondentes a cada coluna da matriz dimensional total e matriz dimensional parcial. A sub-opção 'OP' pode também ser usada com a opção f, para que sejam impressos os produtos adimensionais como fórmulas.

3.3 - Erros.

Qualquer ocorrência de erro é fatal ao programa que está sendo processado.

Duas classes de erros são possíveis:

a) erros de sintaxe: são aqueles ocorridos devido à inobservância das regras gerais ou devido à má formação de comandos.

b) erros de semântica, cujas causas podem ser:
   b-1) variável não definida antes de usada;
   b-2) limites de matrizes ultrapassados (overflow);
   b-3) tentativa de adicionar ou subtrair grandezas de dimensões diferentes;
   b-4) tentativa de exponenciação com expoente não constante.

Em ambos os casos são impressas as seguintes mensagens:

- o número da linha em que o erro ocorreu;

- a classe do erro (sintaxe ou semântica);
- a linha em que o erro ocorreu;
- um apontador para o último caráter analisado.

Com estas mensagens o usuário poderá rápida e facilmente localizar o erro e corrigí-lo.

CAPÍTULO 4

UTILIZAÇÃO DO SISTEMA LAD

4.1 - Entrada e Saída do Sistema.

4.1.1 - Acesso ao Sistema.

O usuário que deseja processar um programa em LAD, deve inicialmente carregar o sistema na memória principal do computador. Para isto basta o comando:

.RUN LAD

O sistema será carregado e a seguinte mensagem será impressa:

```
&&&
```

A listagem do programa fonte é uma saída opcional para o usuário. Para solicitá-la, basta datilografar o caráter barra (/) após o nome do programa ou após a palavra CONVE. O supervisor se encarregará de imprimir a listagem após a execução do programa.

Uma vez recebido o nome do programa armazenado ou a palavra reservada CONVE, o supervisor imprime uma mensagem de início, com data e hora, na forma:

INICIO : dd-mmm-aa hh:mi ss.ds
onde: dd=dia mmm=mes aa=ano hh=hora mi=minutos
ss=segundos ds=décimos de segundo.

A partir deste ponto o Sistema LAD entra em fase de interpretação, analisando e executando um comando por vez.

Após o término da fase de interpretação de todo um programa, o sistema volta à fase de supervisão, imprimindo a mensagem:

FINAL : dd-mmm-aa hh:mi ss.ds

e provendo a listagem do programa executado se previamente solicitada.

Imprimindo ou não a listagem, o supervisor novamente "perguntará" ao usuário se o próximo programa é conversacional ou não.

Quando não houver mais programas em LAD a executar dever-se-á imprimir após o asterisco a palavra reservada EXIT, que causará o término da execução do Sistema LAD.

4.1.3 - Interpretador.

Nesta fase, o Sistema LAD procederá à análise e execução de comandos de programa fonte, seja este conversacional ou armazenado.

A transição entre as fases de interpretação e supervisão pode se dar de duas maneiras:

1) saída normal através da regra 1 (capítulo 3);
2) saída anormal devido a erro ocorrido.

Na saída por ocorrência de erro, serão impressas as mensagens como descrito na seção 3.3 .

Em qualquer dos casos acima, o controle volta ao supervisor.

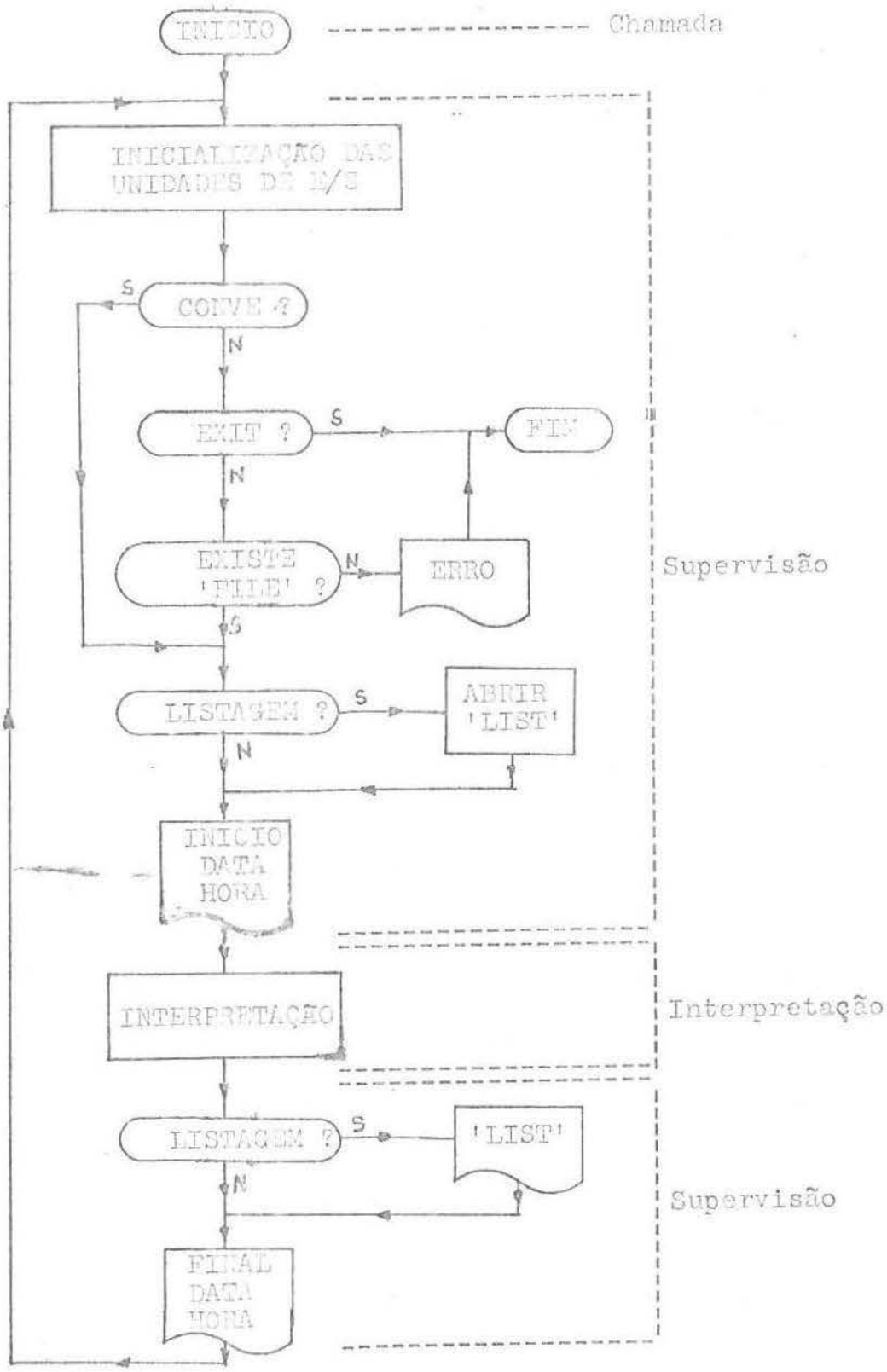
Se o programa que está sendo analisado e executado pelo interpretador for conversacional, a solicitação de um comando far-se-á por meio da impressão de um sinal de interrogação (?) no início da linha. Após este sinal, o usuário deverá datilografar o comando a ser analisado e executado.

4.1.4 - Fluxograma do Sistema.

O fluxograma da página seguinte mostra a estrutura do Sistema LAD. 'FILE' é um nome genérico de programa armazenado, e 'LIST' é o nome do arquivo de listagens.

No bloco denominado INTERPRETAÇÃO, suponha-se a chamada da rotina de análise sintática.

No Exemplo 1 do Apêndice 4, tem-se uma listagem com as mensagens específicas.


INICIO
Chamada
INICIALIZAÇÃO DAS UNIDADES DE E/S
CONVE ?
S
N
EXIT ?
S
FIM
N
EXISTE 'FILE' ?
N
ERRO
S
Supervisão
LISTAGEM ?
S
ABRIR 'LIST'
N
INICIO DATA HORA
INTERPRETAÇÃO
Interpretação
LISTAGEM ?
S
'LIST'
N
Supervisão
FINAL DATA HORA

4.2 - Exemplos.

Sejam os seguintes problemas:

1) Tem-se um corpo de massa 100 gramas e velocidade 10 cm/seg. Deseja-se saber qual é a massa, velocidade e energia cinética deste corpo, medidas no sistema MKS Giorgi.

2) Um certo fenômeno depende das grandezas: massa, força, velocidade, aceleração, massa específica e energia cinética. Verificar:
   a) qual a relação entre força, massa e aceleração;
   b) qual a relação entre todas as grandezas presentes;
   c) dar a forma mais geral para a força em função das demais grandezas.

Os problemas acima ilustram os dois tipos básicos de aplicação. O primeiro envolve mudança de unidades e o segundo cálculo de produtos adimensionais.

Para a solução do problema 1, declara-se o sistema de unidades novo (MKS Giorgi) com as unidades fundamentais: quilograma (KG), metro (METRO) e segundo (SEG). O sistema de unidades antigo deve ser definido em função do novo. No problema 1, então, grama (GR), centímetro (CM) e segundo (SEG) devem ser definidos em função de KG, METRO e SEG, respectivamente.

Como a unidade de tempo, em ambos os sistemas é a mesma, não há necessidade de redefiní-la como unidade antiga.

Qualquer grandeza calculada pelo sistema antigo será agora referida ao sistema novo.

A codificação em linguagem LAD, para resol-

ver este problema poderia ser a seguinte:

```
$
'DECLARO' : MASSA(KG),COMPR(METRO),TEMPO(SEG);
GR=KG/1000;
CM=METRO/100;
MASSA=100*GR;
'IMPRIMIR' (MASSA);
VELOC=10*CM/SEG;
'IMPRIMIR'(VELOC);
ENERG=(1/2)*MASSA*VELOC^2;
'IMPRIMIR' (ENERG);
'IMPRIMIR' ('MATDEF');
'COMENTAR' : FIM DE EXEM1
$$$
```

No Apêndice 4, Exemplo 2, tem-se este programa executado de modo conversacional com listagem, e o Exemplo 3 ilustra a execução do mesmo programa como programa armazenado, também com listagem, sob o nome de EXEM1.

Para a solução do problema 2, pode-se declarar um sistema de unidades arbitrário, tal como M,L,T, pois só interessam as equações dimensionais das grandezas. A codificação em linguagem LAD do problema, poderia ser:

```
$'DECLARO' : M(M),L(L),T(T) ;
MASSA=M ;
FORSA=M*L/T^2 ;
VELOC=L/T ;
ACELE=L/T^2 ;
RO=M/L^3 ;
ENERG=M*L^2/T^2 ;
'FUNCAOPI' |FORSA,MASSA,ACELE| ;
'IMPRIMIR'('OP';'MATDIM';'MATDIMP';'OP';'MATADIM') ;
'FUNCAOPI' |FORSA,ENERG,VELOC,MASSA,RO,ACELE| ;
'IMPRIMIR'('MATDIM';'OP';'MATADIM';'FUNCAO'(FORSA)) ;
'IMPRIMIR' ('MATDEF') ;
```

'COMENTAR' : FIM DE EXEM2
$$$

No Apêndice 4, Exemplo 4, tem-se a solução deste programa, como conversacional e sem listagem.

Os Exemplos 5 e 6 do Apêndice 4 são codificações diferentes para os problemas 1 e 2 respectivamente.

4.3 - Transformação de Programa Conversacional para Programa Armazenado.

Por vezes o usuário necessita que um programa resolvido de modo conversacional se transforme em armazenado.

O Sistema LAD permite esta transformação da seguinte maneira: após o término do processamento do programa conversacional em questão, o usuário sai do Sistema LAD através do comando EXIT. A transformação só poderá ser feita se a listagem foi requerida. Se assim foi feito, o programa está em um arquivo de nome FORØ3.DAT, cujas linhas estão enumeradas. Após sair do sistema o usuário, através do comando RENAME do PDP-10 dá um novo nome ao arquivo. Os números das linhas devem ser eliminados, o que facilmente se consegue através do programa TECO (Text and Editor Corrector). |3|

No Apêndice 4 o Exemplo 7 ilustra todos os passos do procedimento acima descrito.

CAPÍTULO 5

IMPLEMENTAÇÃO DO SISTEMA LAD

5.1 - Observações Gerais.

O Sistema LAD foi implementado no Sistema DEC-10 da UNICAMP, em linguagem FORTRAN IV. |2|

O PDP-10, computador de grande porte, apresenta entre outras duas características bastante importantes para o desenvolvimento do Sistema LAD: operação em "time-sharing", permitindo ao usuário itereção contínua com o sistema e um conjunto de programas para criação e manutenção de arquivos bastante sofisticado. |3|

5.2 - Esquema Básico.

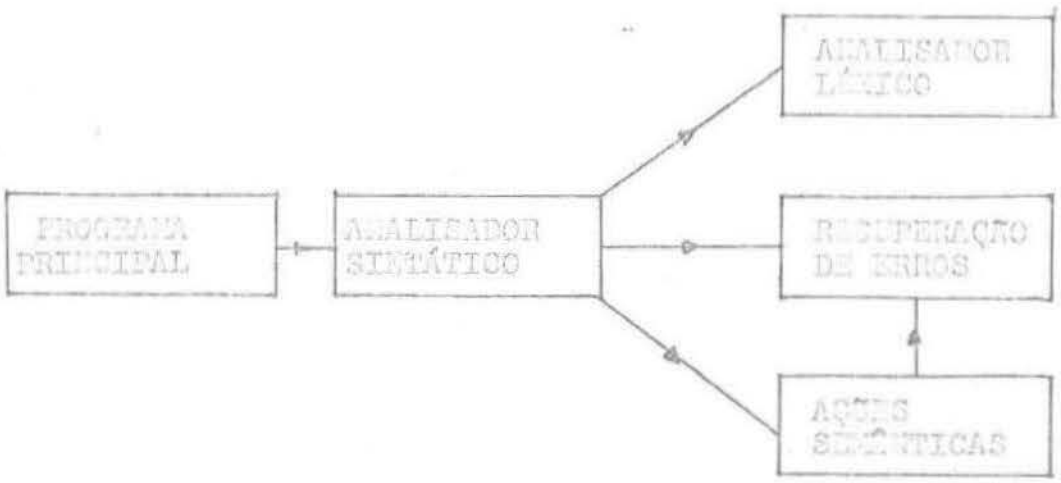
O sistema foi implementado utilizando-se um gerador automático de compiladores criado em Brandeis University |1|, o que em muito simplificou o trabalho.

Os passos envolvidos no processamento de um comando em LAD, são os seguintes:

(a) leitura do comando e verificação da não ocorrência de símbolos ilegais;
(b) análise do comando e determinação da ação desejada;
(c) execução da ação determinada no item b;
(d) em caso de ocorrência de erro durante qualquer das passagens anteriores, emissão da mensagem e procedimento de recuperação da condição normal, se possível.

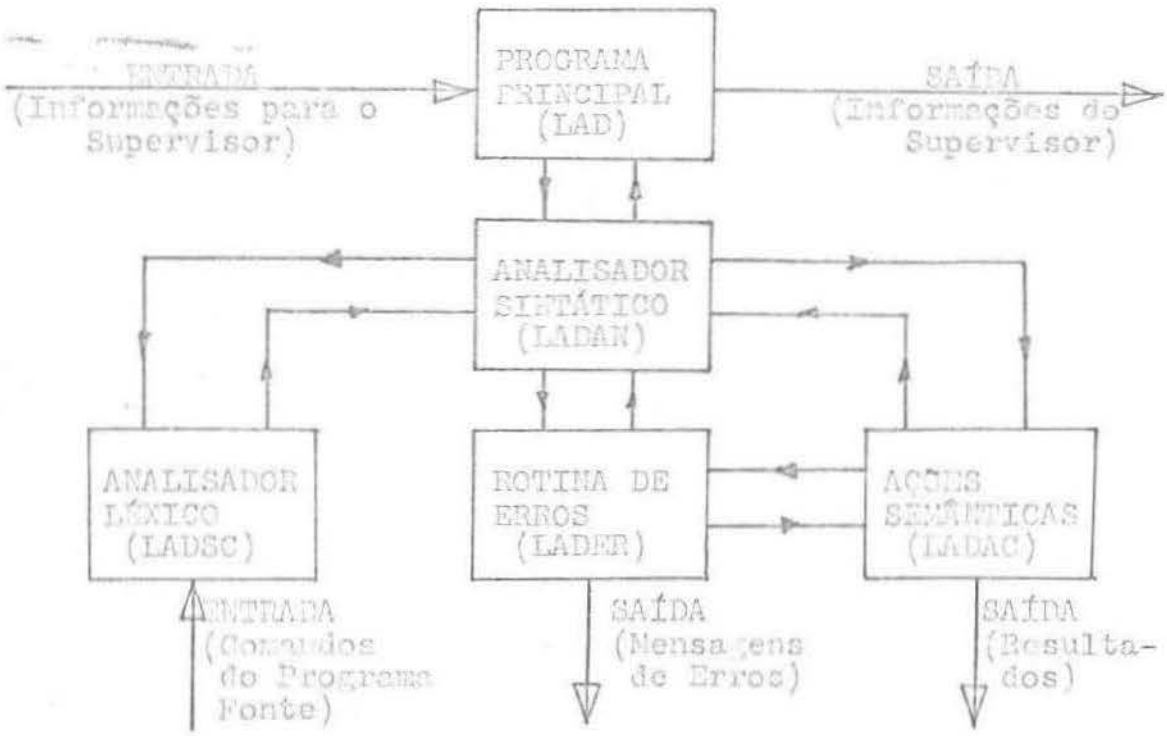
O gerador de compiladores constrói automaticamente, mediante entradas apropriadas, as rotinas de scanning (item a), análise (item b), recuperação de erro (item d) e

um programa principal, completando a estrutura esquematizada abaixo:



As ações semânticas (item c), que são particulares de cada tipo de aplicação, ficam a cargo do usuário.

Utilizando o gerador de compiladores foi obtido o seguinte esquema:



A geração automática da rotina de erros com o

UNICAMP
BIBLIOTECA CENTRAL

sistema gerador de compiladores utilizado é razoavelmente complexa. Assim sendo, para simplificar uma primeira implementação decidiu-se programar manualmente a rotina de erros ao invés de gerá-la com o sistema.

5.3 - Rotinas Geradas Automaticamente.

5.3.1 - Programa Principal.

Para a geração do programa principal, o gerador de compiladores exige apenas o nome que será dado ao programa; foi utilizado o nome LAD.

Após a geração de LAD, foi necessário modificá-lo parcialmente para que o Sistema aceitasse o modo conversacional de processamento, oferecesse a opção de listagem do programa fonte e fornecesse informações tais como data e hora de início e término de processamento de um programa.

O esquema da seção 4.1.4 é o fluxograma do programa principal, sendo o bloco INTERPRETAÇÃO a chamada da rotina de análise sintática e semântica.

5.3.2 - Rotina de Análise Léxica.

Esta rotina recebeu o nome de LADSC, e é a responsável pela leitura do programa fonte seja ele conversacional ou armazenado e pela análise dos elementos que constituem um comando.

Para a geração de LADSC o gerador de compiladores necessita de informações sôbre os elementos léxicos da linguagem. Estes elementos são:

(a) símbolos básicos, bem como seus delimitadores e comprimento máximo;
(b) modo de tratamento de brancos;
(c) código de identificadores e seu tamanho máximo;

(d) tipos de constantes e seus códigos.

Os códigos dos elementos são dados pela gramática LADGR (ver Apêndice 3).

5.3.3 - Rotina de Análise Sintática.

Esta rotina, chamada LADAN, é encarregada de verificar se os comandos estão escritos corretamente (isto é se estão sintaticamente corretos) e de chamar, em certos pontos do reconhecimento, ações semânticas apropriadas.

Para sua geração, é dada ao gerador de compiladores a gramática da linguagem LAD, definida em LADGR (ver Apêndice 3) no formato adequado.

Erros detetados durante a execução de LADSC ou LADAN aparecem na listagem como erros de sintaxe.

5.4 - Rotina de Ações Semânticas.

5.4.1 - Descrição Geral.

Enquanto as rotinas LADAN e LADSC se desincumbem da aceitação de um comando segundo a gramática LADGR, a rotina de ações semânticas é a responsável pela execução propriamente dita do comando quando este for considerado sintaticamente correto.

A rotina de ações semânticas recebeu o nome de LADAC, constando de várias subrotinas. A ação semântica número n é "chamada" na gramática LADGR (ver Apêndice 3) por um símbolo da forma #n . Cada subrotina de LADAC constitui uma ação semântica.

Há um total de 29 ações semânticas cada uma com uma finalidade específica.

5.4.2 - Estrutura de Dados do LADAC.

(a) Matriz Definição de Grandezas e Unidades - MATDMF(I,J)
É a matriz que armazena as grandezas e suas unidades. Esta matriz funciona como pilha de cálculos de expressões aritméticas, com o topo caminhando no sentido da última para a primeira linha.

(b) Vetor de Medidas das Grandezas - VALOR(I)
É um vetor no qual cada elemento corresponde à medida da grandeza cujas dimensões estão na linha de mesmo índice de MATDEF.

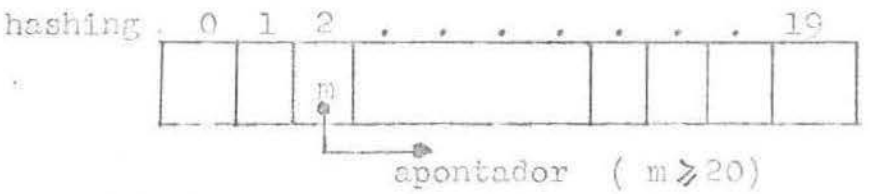
(c) Matriz Dimensional Total - MATDIM(I,J)
É a matriz que armazena em cada coluna as dimensões das grandezas ou expressões aritméticas com as quais serão calculados os produtos adimensionais.

(d) Matriz Dimensional Parcial - MATDIMP(I,J)
Esta matriz tem as mesmas características de MATDIM, dela diferindo apenas quando MATDIM é singular. Neste caso MATDIMP é obtida a partir de MATDIM eliminando-se uma ou mais linhas.

(e) Matriz de Produtos Adimensionais - PI(I,J)
É a matriz que armazena os expoentes das grandezas ou expressões aritméticas que tomam parte na formação dos produtos adimensionais. Cada linha corresponde a um produto.

(f) Dicionário de Variáveis e Expressões Aritméticas - DICION(I)
É um vetor que armazena variáveis e expressões aritméticas segundo a técnica de "hashing". [4] Para índices i, tal que $0 \leqslant i \leqslant 19$, cada elemento de DICION é um apontador para a posição onde está armazenada a variável ou expressão. As variáveis que representam as grandezas e expressões aritméticas estarão armazenadas a partir de i=20.

O hashing da variável ou expressão é calculado segundo código de módulo 20 e este será o índice da parte de hashing do dicionário. A figura seguinte ilustra a parte de hashing do dicionário:



Se DICION(hashing) < 0 então não há nenhuma variável ou expressão com aquele código hashing.

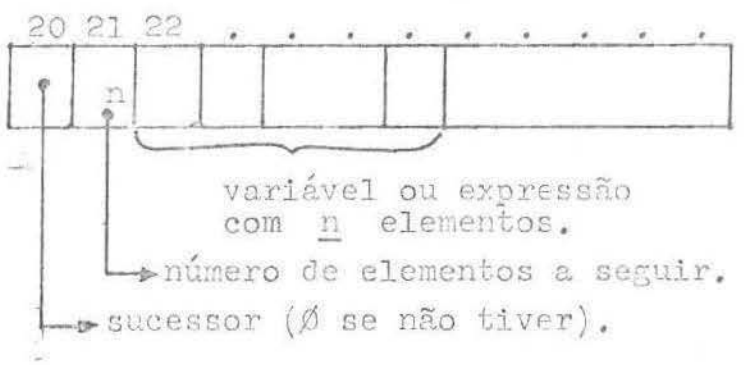
Para i ≥ 20 tem-se a parte de armazenamento, que, para cada variável ou expressão aritmética, consta de 3 campos:

Campo 1 - aponta para o sucessor que tem o mesmo hashing

Campo 2 - dá o número de elementos que a variável ou expressão contem;

Campo 3 - contem a variável ou expressão armazenada.

A figura seguinte mostra os campos citados:



(g) Linhas de Dados - CARD(I)

onde 1 ≤ i ≤ 144 . Este vetor, para 73 ≤ i ≤ 144, armazena a linha que está sendo analisada e, para 1 ≤ i ≤ 72, a parte da linha já analisada.

As outras estruturas são auxiliares e têm usos locais.

5.4.3 - Descrição das Ações Semânticas.

Ação #1 - Inicializa matrizes, vetores e variáveis.

Ações #2 e #23 - Definem uma grandeza em MATDEF, quando a grandeza é definida via um comando aritmético.

Ações #3 e #6 - São as ações que montam a matriz MATDIM e são executadas quando do reconhecimento do comando cujo nome é 'FUNCAOPI'.

Ações #4 e #7 - Respondem pela impressão das fórmulas gerais, para a opção de impressão 'FUNCAO'.

Ação #5 - Ação que calcula os expoentes das grandezas ou expressões aritméticas que formam os produtos adimensionais, a partir de MATDIMP, e os coloca na matriz PI.

Os expoentes dos produtos adimensionais são calculados a partir das soluções de um sistema de equações lineares homogêneas cuja matriz dos coeficientes é dada pela matriz MATDIMP. Esta matriz tem um determinante d_c, formado pela matriz quadrada D das c últimas colunas. O número de linhas de MATDIMP é c e o número de colunas é n.

Sejam $d_{i,j}$ os determinantes das matrizes formadas a partir da matriz D, eliminando-se desta a coluna j e substituindo-a pela coluna i de MATDIMP.

Pelo capítulo 2, o número de produtos adimensionais será (n-c). Ainda pelo capítulo 2, seção 2.6.3 as variáveis dependentes em uma linha seriam zero ou um. Como os expoentes devem ser inteiros, a solução do sistema não se modifica se forem multiplicadas por d_c. Com as (n-c) soluções do sistema de equações lineares homogêneas multiplicadas por d_c, cada solução será o conjunto de expoentes das variáveis ou expressões aritméticas que formarão um produto adimensional.

Após o cálculo do sistema fundamental de soluções é conveniente dividir cada linha solução pelo seu máximo divisor comum.

Os determinantes são calculados através do método de Gauss modificado, pelo fato de serem os seus elementos valores inteiros.

Ações #9 e #10 - Estas ações imprimem as frases especificadas no comando 'COMENTAR'.

Ação #9 - Ação responsável pela impressão das dimensões, unidades e medida de grandezas ou expressões aritméticas, quando presentes como opção no comando 'IMPRIMIR'.

Ações #11 e #12 - são responsáveis, respectivamente pela execução na pilha de cálculo, de somas e subtrações presentes em expressões aritméticas.

Ação #13 - Inverte o sinal de constantes ou variáveis em expressões aritméticas, na pilha de cálculo. É a ação ativada pela presença do menos unário.

Ações #14 e #15 - são responsáveis, respectivamente pela execução de multiplicações e divisões em expressões aritméticas na pilha de cálculo.

Ação #16 - executa a exponénciação de constantes, variáveis ou expressões aritméticas, na pilha de cálculo.

Ações #17 e #18 - coloca no topo da pilha de cálculo dimensão nula e no topo de VALOR o valor real de constantes reais e inteiras, respectivamente.

Ação #19 - coloca no topo da pilha de cálculo as dimensões de uma grandeza e no topo de VALOR a medida desta grandeza.

Ações #20, #26, #27, #28 e#29 - são responsáveis pela impres

UNICAMP

são de: MATDEF, MATDIM, MATDIMP, PI ('MATADIM') e 'OP', respectivamente.

Ações #21 e #22 - armazenam em MATDEF grandezas e unidades básicas. É ativada pelo comando 'DECLARO'.

Ações #24 e #25 - são as responsáveis pelo tratamento de parêntesis à esquerda e à direita, respectivamente, em expressões aritméticas.

5.5 - Rotina de Erros.

Com o nome LADER, esta rotina é a responsável pela manipulação de condições de erro.

Quando chamada, LADER imprime uma mensagem conforme descrito em 3.3 e faz com que o processamento do programa seja interrompido, voltando o controle ao Supervisor LAD.

5.6 - Performance.

O Sistema LAD ocupa 17K palavras de memória no PDP-10.

O código obtido pela utilização do gerador de compiladores é bastante otimizado e comprovadamente eficiente. Em adição, procurou-se, através da adoção de técnicas sofisticadas de estrutura de dados |4|, na rotina de ações semânticas, melhorar ainda mais para o desempenho total do sistema.

Tempos típicos de execução de programas de aplicação podem ser encontrados no Apêndice 4.

CAPÍTULO 6

CONCLUSÕES

O Sistema LAD, como descrito neste trabalho, já é totalmente operacional no computador PDP-10 do Centro de Computação da Universidade Estadual de Campinas (CCUEC). O sistema foi exaustivamente testado e demonstrou ser bastante confiável, resolvendo com grande rapidez problemas de Análise Dimensional apresentados em formato bastante próximo à notação comumente empregada pelos pesquisadores. Utilizando o modo conversacional de processamento, o usuário encontra grande facilidade de programação o que em muito simplifica a pesquisa da forma mais conveniente dos produtos adimensionais. A característica de transformação de programas da forma conversacional para a forma de programa armazenado evita ao usuário a monótona tarefa de repetir todo o diálogo de entrada, a cada vez que deseja executar o programa.

Em resumo, acredita-se ter desenvolvido um sistema prático e poderoso que, embora ainda em uma primeira implementação, já pode ser bastante útil como ferramenta de pesquisa e ensino em Análise Dimensional e suas aplicações.

Dentre os melhoramentos que poderão ser introduzidos em futuras versões do sistema, citam-se:

(a) Recuperação de Erros: certos tipos de erros podem ser recuperados. A rotina de reconhecimento de erros atualmente utilizada não o permite. Se esta rotina for suprida pelo gerador de compiladores |1| a recuperação de erros poderá ser feita possível;

(b) Transformação de Programas: para a transformação de programas do modo conversacional para armazenado o usuário deve recorrer a recursos de software do

PDP-10. Isto pode ser evitado se for projetado um comando em LAD, de tal modo que não se necessite interromper o processamento do Sistema LAD para fazer a transformação.

(c) Gráficos Adimensionais: gráficos entre produtos adimensionais são largamente empregados em ciências experimentais. Assim sendo, uma adição altamente desejável seria a implementação no Sistema LAD de comandos que permitissem o levantamento de gráficos entre produtos adimensionais via um equipamento traçador de gráficos.

(d) Extensão para Teoria dos Modêlos: também é uma característica extremamente desejável para aplicações experimentais. Dadas as propriedades de um modêlo e fatôres de escala entre modêlo e protótipo um ou mais comandos em LAD poderiam ser criados de maneira a permitir o cálculo direto das características do protótipo.

(e) Tabelas Internas: introdução de tabelas internas de conversão entre as unidades mais usuais, juntamente com nomes especiais reservados para tais unidades. Desta maneira será desnecessário entrar com as relações entre tais unidades para cada problema, tornando mais rápida e cômoda a solução de problemas de conversão de unidades.

As melhorias acima sugeridas contribuirão para que o Sistema LAD se torne cada vez mais geral e adequado para a classe de usuários a que se destina.

BIBLIOGRAFIA

|1| COHEN, J. A compiler generator. Walthan, Brandeis Univ, 1972. 2v. Cópia Xerox.

|2| DECSYSTEM; mathematical language handbook. 2ª ed. Maynard, Dec /1972/ 388p. (Decsystem10 Handbook Series)

|3| DECSYSTEM; users handbook. Maynard, Dec /1972/ 836p. (Decsystem10 Handbook Series)

|4| KNUTH, D. E. The art of computer programming; sorting and searching. Reading, Addison-Wesley /c1973/ 722p. (Addison-Wesley Series in Computer Science and Information Processing, v.3)

|5| LANGHAAR, H. L. Dimensional analysis and theory of models. New York, John Wiley /1951/ 166p.

|6| MAIA, L. P. M. Análise dimensional. Rio de Janeiro, Nacionalista, 1960. 110p.

|7| MIKHEYEV, M. Fundamentals of heat transfer. Trad. S. Semyonov. Moscow, Peace Publ, s.d. 376p.

|8| PARKINSON, M. An axiomatic approach to dimensions in physics. American Journal of Physics, Lancaster, 32(3):200-205, 1964.

APÊNDICE 1 - Grandezas e suas Dimensões.

GRANDEZA	SITEMA MASSA	SITEMA FORÇA
Comprimento	$\|L\|$	$\|L\|$
Tempo	$\|T\|$	$\|T\|$
Temperatura	$\|C\|$	$\|C\|$
Massa	$\|M\|$	$\|FL^{-1}T^{-2}\|$
Força	$\|MLT^{-2}\|$	$\|F\|$
Peso Específico	$\|ML^{-2}T^{-2}\|$	$\|FL^{-3}\|$
Densidade	$\|ML^{-3}\|$	$\|FL^{-4}T^{-2}\|$
Pressão	$\|ML^{-1}T^{-2}\|$	$\|FL^{-2}\|$
Velocidade	$\|LT^{-1}\|$	$\|LT^{-1}\|$
Aceleração	$\|LT^{-2}\|$	$\|LT^{-2}\|$
Velocidade Angular	$\|T^{-1}\|$	$\|T^{-1}\|$
Aceleração Angular	$\|T^{-2}\|$	$\|T^{-2}\|$
Energia, Trabalho	$\|ML^{2}T^{-2}\|$	$\|FL\|$
Momentum	$\|MLT^{-1}\|$	$\|FT\|$
Potência	$\|ML^{2}T^{-3}\|$	$\|FLT^{-1}\|$
Momento de uma força	$\|ML^{2}T^{-2}\|$	$\|FL\|$
~~Coeficiente~~ Dinâmico de Viscosidade	$\|ML^{-1}T^{-1}\|$	$\|FL^{-2}T\|$
Coeficiente Cinemático de Viscosidade	$\|L^{2}T^{-1}\|$	$\|L^{2}T^{-1}\|$
Momento de Inércia de uma Área	$\|L^{4}\|$	$\|L^{4}\|$
Momento de Inércia de uma Massa	$\|ML^{2}\|$	$\|FLT^{2}\|$
Tensão Superficial	$\|MT^{-2}\|$	$\|FL^{-1}\|$
Módulo de Elasticidade	$\|ML^{-1}T^{-2}\|$	$\|FL^{-2}\|$

APÊNDICE 2 - Fatores de Conversão de Unidades.

UNIDADES DE COMPRIMENTO:

1 m = 39,37 in = 3,281 ft
1 ft = 30,480 cm = 0,30480 m
1 in = 2,54 cm
1 mi = 5280 ft = 1609,4 m = 160940 cm
1 cm = $6,2137 \times 10^{-6}$ mi

UNIDADES DE FORÇA E MASSA:

1 kg = 2,2046 lb
1 lb = 0,4536 kg
1 ton métrica = 1000 kg
1 kg = 0,06852 slug
1 lb = 4,4482 newton = 444820 dyna
1 slug = 14,594 kg = 14594 g
1 kg (força) = 980665 dyna
1 newton = 0,2248 lb = 10^5 dyna

UNIDADE DE TEMPERATURA:

1 °C = 1,8 °F

UNIDADES DE VISCOSIDADE:

1 poise = 1 g/cm*seg = 1 dyna*seg/cm^2 = 0,002088 slug/ft*seg
= 0,002088 lb*seg/ft^2
1 stoke = 1 cm^2/seg = 0,001076 ft^2/seg

APENDICE 3 : PDE DO INTERPRETADOR LAD
==========

```
S
'S' ';' 'DECLARAR' ':' 'INTEIRO' '(' ')' 'FUNCAOPI' '[' ']'
',' 'COMENTAR' 'MATDEF' 'MATDIM' 'MATDIMP'
'MATALT' 'FUNCAO' 'IDENT' '+' '-' '*' '/' '^' 'REAL'
'INTEIRO' '=' '(' '[' '{'
PROGRAMA BEGIN F1 C S F LSTUBAS F3  LE UB DIMENSAO IDENTIFIC
E2 E1 E4 SEQ I F P ABREPAR FECHAPAR
!!!
PROGRAMA => BEGIN F1 'S' .
BEGIN => 'S' #1 .
F1 => F1 ';' C .
F1 => C .
C => P E #2 .
C => 'DECLARAR' ':' LSTUBAS .
C => 'INTEIRO' '(' E3 ')' .
C => 'COMENTAR' ':' SEQ #3 .
C => 'FUNCAOPI' '[' LE ']' #5 .
LSTUBAS => LSTUBAS ',' UB .
LSTUBAS => UB .
UB => DIMENSAO '(' IDENTIFIC ')' .
LE => LE ',' E2 #6 .
LE => E2 #3 .
F1 => '(' 'IDENT' ')' #7 .
E3 => E3 ';' E4 .
E3 => E4 .
E4 => E2 #9 .
E4 => 'MATDEF' #21 .
E4 => 'MATDIM' #26 .
E4 => 'MATDIMP' #27 .
E4 => 'MATALT' #28 .
E4 => 'FUNCAO' F1 #4 .
E4 => 'GOTO' #29 .
SEQ => SEQ 'IDENT' #10 .
SEQ => 'IDENT' #10 .
E2 => E .
E => E '+' T #11 .
E => E '-' T #12 .
E => '-' T #13 .
E => T .
T => T '*' F #14 .
T => T '/' F #15 .
T => F .
F => F '^' P #16 .
F => P .
P => ABREPAR E FECHAPAR .
P => 'REAL' #17 .
P => 'INTEIRO' #18 .
P => 'IDENT' #19 .
DIMENSAO => 'INTEIRO' #21 .
IDENTIFIC => 'IDENT' #22 .
E => 'IDENT' '=' #23 .
ABREPAR => '(' #24 .
FECHAPAR => ')' #25 .
S
```

ÍNDICE 4 : EXEMPLOS

EXEMPLO 1 : FASES DE SOLUÇÃO DE

```
.RUN LAD                                                    CHAMADA

&&&
```

EXEMPLO 2 : MUDANCA DE UNIDADES (PROGRAMA CONVERSACIONAL).
==========

. JN LAD

&&&

```
A EXPRESSAO ABAIXO :
* ================

ENERG
TEM A DIMENSAO :
MASSA=     1  (KG    )
COMPR=     2  (METRO)
TEMPO=    -2  (SEG   )
TEM O VALOR :        0.0005000
?'IMPRIMIR'('MATRIZ') :

* ***********************************************************
*            MASSA COMPR TEMPO                        VALORES
*      KG        1     0     0                   1.0000000*
*      METRO     0     1     0                   1.0000000*
*      SEG       0     0     1                   1.0000000*
*      GR        1     0     0                   0.0010000*
*      CM        0     1    -2                   0.0100000*
*      MASSA     1     0     0                   0.1000000*
*      VELOC     0     1    -1                   0.1000000*
*      ENERG     1     2    -2                   0.0005000*
* ***********************************************************
?'COMENTAR' : FIM DE EXEM1

?SSS

FIM DE EXEM1

FINAL : 17-DEC-73   14:45  20,2


PROGRAMA FONTE
  1  S
  2  'DECLARAR' : MASSA(KG),COMPR(METRO),TEMPO(SEG) ;
  3  GR=KG/1000 ;
  4  CM=METRO/100 ;
  5  MASSA=1000GR ;
  6  'IMPRIMIR'(MASSA) ;
  7  VELOC=100CM/SEG ;
  8  'IMPRIMIR'(VELOC) ;
  9  ENERG=(1/2)*MASSA*VELOC**2 ;
 10  'IMPRIMIR'(ENERG) ;
 11  'IMPRIMIR'('MATRIZ') ;
 12  'COMENTAR' : FIM DE EXEM1
 13  SSS


&&&
```

E EMPLO 3 : MUDANCA DE UNIDADES (PROGRAMA ARMAZENADO).
=======

,"UN LAD

&&&

F NAL : 17-DEC-73 14:09 54.0

F OGRAMA FONTE

```
  1        S

  2        'DECLARO' : MASSA(KG),COMPR(METRO),TEMPO(SEG) ;

  3        GR=KG/1000 ;

  4        CM=METRO/100 ;

  5        MASSA=100*GR ;

  6        'IMPRIMIR'(MASSA) ;

  7        VELOC=100*CM/SEG ;

  8        'IMPRIMIR'(VELOC) ;

  9        ENERG=(1/2)*MASSA*VELOC**2 ;

 10        'IMPRIMIR'(ENERG) ;

 11        'IMPRIMIR'('CAJDEF') ;

 12        'COMENTAR' : FIM DE EXEM1

 13        SSS
```

```
&&&
```

EXEMPLO 4 : CALCULO DE PRODUTOS ADIMENSIONAIS
(PROGRAMA CONVERSACIONAL).

UM LAD

&&&

```
MATRIZ DIMENSIONAL (TOTAL)
====== ===========
      FORSA MASSA ACELE
M         1     1     0
L         1     0     1
T        -2     0    -2
: ##########################################################

MATRIZ DIMENSIONAL
: ==== ===========
      FORSA MASSA ACELE
M         1     1     0
T        -2     0    -2
: ##########################################################

==========================================================
- -   PI( 1)   ---


        **              **
        *                *
FORSA = * MASSA * ACELE *
        **              **
: ========================================================


MATRIZ ADIMENSIONAL
: ==== ===========
          FORSA MASSA ACELE
PI( 1)        1    -1    -1
? FUNCAOPI'(FORSA,ENERG,VELOC,MASSA,RO,ACELE) ;

?'IMPRIMIR('MATADIM';'OP';'MATADIM';'FUNCAO'(FORSA)) ;

# ##########################################################

MATRIZ DIMENSIONAL (TOTAL)
: ==== ===========
      FORSA ENERG VELOC MASSA    RO ACELE
M         1     1     0     1     1     0
L         1     2     1     0    -3     1
T        -2    -2    -1     0     0    -2
##########################################################
.
: ========================================================
---   PI( 1)   ---


        **              **
        *                *
FORSA = * MASSA * ACELE *
        **              **
==========================================================


==========================================================
---   PI( 2)   ---
```

```
        **                          ** (1/3)
        *       4     -1          3  *
ENERG = * MASSA * RO    * ACELE    *
        **                          **
= ==========================================


= ==========================================
= =   P1( 3)    ---


        **                          ** (1/6)
        *             -1          3 *
VELOC = * MASSA * RO    * ACELE   *
        **                          **
==========================================


MATRIZ ADIMENSIONAL
=====  ============
          FORSA ENERG VELOC MASSA     RO ACELE
P'( 1)        1     0     0    -1      0    -1
P1( 2)        0     3     0    -4      1    -3
P ( 3)        0     0     6    -1      1    -3
FI RELACAO A GUAL P1
VOCE QUER A FUNCIONAL ?
1


        **                **
        *                 *
FORSA = * MASSA * ACELE * * FUNCAO
        **                **
   **
   *      3          -4                -3
   * ENERG  * MASSA    * RO * ACELE    ,
   **
                                        **
          6          -1                -3 *
      VELOC  * MASSA    * RO * ACELE    *
                                        **
?'IMPRIMIR'('MATOFF') :


 ***********************************************
*                 M     L     T              VALOR*
*    M            1     0     0            1.000000*
*    L            0     1     0            1.000000*
*    T            0     0     1            1.000000*
*    MASSA        1     0     0            1.000000*
*    FORSA        1     1    -2            1.000000*
     VELOC        0     1    -1            1.000000*
*    ACELE        0     1    -2            1.000000*
*    RO           1    -3     0            1.000000*
     ENERG        1     2    -2            1.000000*
 ***********************************************
?'COMENTAR' : FIM DE EXEC?
```

```
FINAL : 17-DEC-73   15:03 13.2


& &&&&
```

EXEMPLO 5 : PROBLEMA DO EXEMPLO 3 COM CODIFICACAO DIFERENTE
(PROGRAMA ARMAZENADO).

. UN LAD

```
F &&&&
```

FINAL : 18-DEC-73 16:18 51.9

PROGRAMA FONTE

```
  1  S'DECLARO' : MASSA(KG),COMPR(METRO),TEMPO(SEG) ;

  2  GR=KG/1000; CM=METRO/100;

  3  'IMPRIMIR'(100*GR:

  4                 10*CM/SEG;

  5                 1/2*100*GR*(10*CM/SEG)**2;

  6                 'MATDEF'):

  7  'COMENTAR' : FIM DE EXEMP

  8  SSS
```

```
&&&
```

EXEMPLO 6 : PROBLEMA DO EXEMPLO 4 COM CODIFICACAO DIFERENTE
(PROGRAMA ARMAZENADO).

```
.RUN LAD

&&&
```

```
====== ============
M          1      1      0
T         -2      0     -2
##################################################


==================================================
---   PI( 1)   ---

          **                  **
          *                    *
COL  1  = * COL  2 * COL  3 *
          **                  **
==================================================


MATRIZ ADIMENSIONAL
====== ============
PI( 1)         1     -1     -1
##################################################

COLUNA = 1
EXPRESSAO =
==========
M*L/T^2
COLUNA = 2
EXPRESSAO =
==========
M*L^2/T^2
COLUNA = 3
EXPRESSAO =
==========
L/T
COLUNA = 4
EXPRESSAO =
==========
M
COLUNA = 5
EXPRESSAO =
==========
M/L^3
COLUNA = 6
EXPRESSAO =
==========
L/T^2
MATRIZ DIMENSIONAL (TOTAL)
====== ============
M          1      1      0      1      1      0
L          1      2      1      0     -3      1
T         -2     -2     -1      0      0     -2
##################################################

==================================================
---   PI( 1)   ---


          **                  **
          *                    *
COL  1  = * COL  4 * COL  6 *
          **                  **
```

```
==================================================

==================================================
---   PI(  2)   ---

          **                                  ** (1/3)
          *        4          -1          3   *
COL  2  = * COL  4  * COL  5   * COL  6    *
          **                                  **
==================================================


==================================================
---   PI(  3)   ---

          **                                 ** (1/6)
          *                  -1          3   *
COL  3  = * COL  4 * COL  5   * COL  6    *
          **                                 **
==================================================


MATRIZ ADIMENSIONAL
====== ============
PI( 1)        1     0     0    -1     0    -1
PI( 2)        0     3     0    -4     1    -3
PI( 3)        0     0     6    -1     1    -3
EM RELACAO A QUAL PI
VOCE QUER A FUNCIONAL ?
1

        **                **
        *                  *
COL01 = * COL04 * COL06 * * FUNCAO
        **                **
    **
    *      3          -4                  -3
    * COL02  * COL04   * COL05 * COL06   ,
    **
                                          **
          6          -1                  -3  *
      COL03  * COL04   * COL05 * COL06     *
                                          **
**************************************************
*                  M     L     T                VALOR*
*    M             1     0     0            1.0000000*
*    L             0     1     0            1.2000000*
*    T             0     0     1            1.0000000*
**************************************************
FIM DE EXI02


FINAL : 18-DEC-73  16:34 03.4


PROGRAMA FONTE
  1  S'DECLARO' : M(M),L(L),T(T);

  2  'FUNCAOPI'[M*L/T^2,L/T^2];
```

```
3  'IMPRIMIR'('OP':'MATDIM':'MATDIMP':'OP':'MATADIM');

4  'FUNCAOPI'[%L/T^2,%L^2/T^2,

5              L/T,%,'/T^3,L/T^2];

6  'IMPRIMIR'('OP':'MATDIM':'OP':'MATADIM':'FUNCAO'(COL#1);

7              'MATDEF');

8  'COMENTAR': FIM DE EXE12

9  SSS
```

```
&&&
```

```
EXEMPLO 7 : TRANSFORMACAO DE PROGRAMA CONVERSACIONAL
            EM PROGRAMA ARMAZENADO.

.RUN LAB


&&&
```

```
RENAME NOMEP=FORØ3.DAT
FILES RENAMED:
FORØ3.DAT

TECO NOMEP

5<4DSLS>SHT$$
      $
      'COMENTAR' : TRANSFORMACAO DE PROGRAMA CONVERSACIONAL :
      'COMENTAR' : PARA PROGRAMA ARMAZENADO :
      'COMENTAR' : FIM DO EXEMPLO  "
      $$$
*EX$$
```