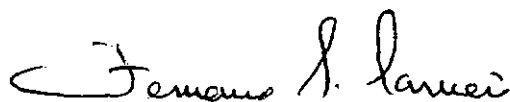


"ANÁLISE DE ESTRUTURAS RETICULADAS PELO MÉTODO
DAS MATRIZES DE TRANSFERÊNCIA"

Eldon Londe Mello

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA (M.Sc.).

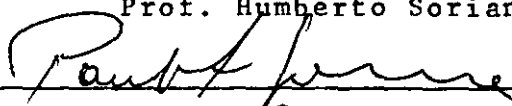
Aprovada por :



Prof. Fernando Lobo Carneiro
Presidente



Prof. Humberto Soriano



Prof. Paulo Alcântara Gomes

RIO DE JANEIRO
ESTADO DA GUANABARA - BRASIL
FEVEREIRO DE 1975

i

À minha esposa

Aos meus pais

AGRADECIMENTOS

- Ao Professor Fernando L.L.B.Carneiro, pela dedicação, orientação e incentivo.
- Ao Professor Alberto Luiz Coimbra, pelo pioneirismo e incentivo aos cursos de Pós-Graduação.
- A todos os Professores da COPPE, pelos valiosos ensinamentos.
- Ao Núcleo de Computação Eletrônica da UFRJ e ao Centro de Processamento de Dados da UnB pela valiosa colaboração.

SINOPSE

São apresentados os conceitos do método das matrizes de transferência na análise estrutural bem como quatro programas automáticos, em FORTRAN, específicos para as estruturas reticuladas mais frequentes na construção civil.

Foram desenvolvidos em um computador IBM 1130 com 8K de memória interna e permitem analisar pórticos simples ou associados de um andar, no plano ou no espaço, vigas contínuas e vigas sobre base elástica.

Além das cargas usuais e variação uniforme de temperatura eles permitem obter as linhas de influência através das deformações impostas às juntas.

Desta forma, esses programas são bastante eficientes na análise de arcos, vigas helicoidais e viadutos transformados em poligonal com grande número de barras.

ABSTRACT

The transfer matrices methods concepts in structural analysis are established as well as four automatic programs, in FORTRAN, turned over the framed structures frequently used in civil construction.

They were developed for an IBM 1130, 8K, internal memory computer and permit to analyse simple or associated frames of one floor, in the plane or in space, continuous beams and beams on continuous linear elastic support.

Besides the usual loading condition and uniform temperature change, they allow us to get the influence lines through imposed displacements at joints.

This way, these programs are efficient enough in the analysis of arches, helicoidal beams and viaducts transformed in a polygonal form with a great number of bars.

ÍNDICE

CAPÍTULO I - INTRODUÇÃO	1
CAPÍTULO II - O MÉTODO DAS MATRIZES DE TRANSFERÊNCIA	
2.1 - Vetor de estado.....	5
2.2 - Matriz de transferência	7
2.3 - Generalização para várias barras	8
2.4 - Matriz fronteira	9
CAPÍTULO III - OBTENÇÃO DAS MATRIZES DE TRANSFERÊNCIA	
3.1 - Integração da equação diferencial	12
3.2 - Equações da Resistência dos Materiais	15
3.3 - Através da matriz de rigidez	17
CAPÍTULO IV - MATRIZ DE TRANSFERÊNCIA DE PÓRTICOS E VIGAS	
4.1 - Viga contínua	21
4.2 - Pórtico plano	25
4.3 - Pórtico no espaço	28
4.4 - Viga sobre base elástica	32
CAPÍTULO V - MATRIZ FRONTEIRA	
5.1 - Ausência de descontinuidade	40
5.2 - Cargas nas juntas	41

5.3 - Deformações impostas	42
5.4 - Apoios elásticos	44
5.5 - Ramificações	46
5.6 - Liberações generalizadas	48

CAPÍTULO VI - ANÁLISE DAS ESTRUTURAS

6.1 - Análise geral	52
6.2 - Viga contínua	55
6.3 - Viga sobre base elástica	59
6.4 - Pórtico plano	60
6.5 - Pórtico no espaço	63

CAPÍTULO VII - DIAGRAMA DE BLOCOS

7.1 - Fluxograma geral	65
7.2 - Estrutura dos dados	82

CAPÍTULO VIII - PROGRAMAS EM FORTRAN

8.1 - Programa VCMT	95
8.2 - Programa VBEMT	118
8.3 - Programa PPMT	136
8.4 - Programa PEMT	158
8.5 - Código de erros	193

APÊNDICE

A - Notações utilizadas no desenvolvimento teórico	195
B - Aplicações dos programas	197
C - Conclusões sobre a eficiência dos programas	221

REFERÊNCIAS BIBLIOGRÁFICAS	245
----------------------------------	-----

CAPÍTULO I

INTRODUÇÃO

Antes do advento dos computadores eletrônicos não havia a preocupação de sintetizar e unificar os processos de resolução de estruturas. Pelo contrário, a tendência dos pesquisadores era a de obter processos cada vez mais simplificados e específicos a determinados tipos de estruturas, com a finalidade precípua de tornar menos complicada e mais rápida sua análise.

Surgiram então vários processos analíticos e gráficos, todos derivados dos dois métodos gerais da Mecânica das Estruturas, o método das Forças e dos Deslocamentos.

Os métodos gerais eram raramente usados porque envolvem um grande volume de cálculo, mesmo para se estabelecer o sistema de equações de compatibilidade de deslocamentos e de equilíbrio.

Montado o sistema de equações restava, finalmente resolvê-lo acarretando cálculos enfadonhos e repetitivos.

Com o advento do computador eletrônico, os pesquisadores voltaram suas atenções para os métodos gerais. A formulação matricial mostrou-se a mais conveniente para os computadores razão pela qual os métodos gerais foram reestruturados e formulados matricialmente. Além disso, ela propicia uma visão global das proprieda-

des estruturais genericamente representadas em compactas equações matriciais.

Naturalmente, os conceitos de idealização estrutural muitas vezes despercebidos ou implícitos numa análise manual tiveram de ser explícitos para permitir uma formulação sistemática das estruturas.

Assim, uma estrutura reticulada é constituída de barras e juntas, os meios contínuos são discretizados em números finitos de elementos. As cargas são consideradas divididas em dois grupos, as que atuam nas barras ou elementos e as que atuam nas juntas ou nós.

O objetivo deste trabalho foi o de apresentar quatro programas capazes de analisar estruturas reticuladas aporticadas, estática ou cinemáticamente determinadas ou indeterminadas, pelo método das matrizes de transferência.

O método das matrizes de transferência, largamente empregado nos domínios da Física, foi introduzido no cálculo estrutural por Falk e Pestel ⁽¹⁾, por volta de 1950.

A característica principal do método é a de se manipular somente com produtos de matrizes de pequenas dimensões. Consiste "em reduzir" a estrutura a uma única "barra" representativa e estabelecer as condições de equilíbrio e de compatibilidade nos apoios extremos fornecendo, portanto, um sistema de equação bem determinado que, resolvido, permite a análise detalhada de toda a estrutura.

Embora o método permita analisar pórticos múltiplos, os programas apresentados foram preparados para resolver pórticos constituídos somente de barras sequenciais, podendo cada junta conter uma ramificação. Os programas para estas estruturas são bastante simples e podem ser processados em computadores de pequeno porte, mesmo para estruturas de elevado número de juntas.

Apesar desta limitação, os programas permitem analisar um número muito variado de estruturas encontradas frequentemente na prática tais como: viadutos, arcos, escadas helicoidais, vigas balcão, vigas contínuas e vigas sobre base elástica, submetidas a cargas permanentes e móveis (linhas de influência), e a variação uniforme de temperatura.

Os capítulos seguintes tratam do desenvolvimento teórico do método e da obtenção das matrizes de transferência e matrizes fronteira para cada uma das quatro estruturas analisadas neste trabalho.

O capítulo VII apresenta os diagramas de blocos para o programa PEMT, pórtico no espaço, que facilitam a compreensão dos programas e permitem sua utilização imediata devido à tentativa de apresentá-los sob forma de "Linguagem Orientada" - uma aproximação grosseira do sistema STRESS da IBM.

Devido a grande semelhança entre os programas de viga contínua de flexão livre (VCMT), viga sobre base elástica (VBEMT) e pórtico plano (PPMT), com o programa PEMT, julgou-se desnecessário incluir seus diagramas de blocos. As diferenças residem, bási

camente, nas sub-rotinas utilizadas por cada programa.

Os diagramas de blocos das sub-rotinas também são desne
cessários porque as matrizes de transferência das barras e as ma
trizes fronteira das juntas estão detalhadamente explicadas no de
senvolvimento teórico.

O capítulo VIII apresenta a programação em linguagem
FORTRAN para os quatro programas.

No apêndice são dadas as notações utilizadas no desenvol
vimento teórico, exemplos de aplicação dos programas e conclusões
gerais.

CAPÍTULO II

O MÉTODO DAS MATRIZES DE TRANSFERÊNCIA

2.1 - Vetor de estado:

Uma seção qualquer de uma barra contém ações e deslocamentos associados.

As ações numa seção de uma barra de pórtico no espaço são a força normal (axial), as forças cortantes segundo y e z , os momentos fletores segundo y e z e o momento torsor.

Essas ações estão representadas na figura 2.1 com a notação genérica A e os deslocamentos associados com a letra D .

O vetor coluna que contém esses deslocamentos e ações é chamado de vetor de estado da seção e é referido ao sistema local obviamente.

Portanto, o vetor de estado da extremidade j da barra i , designado por $\{V\}_{j,i}$ é da forma

$$\{V\}_{j,i} = \{D_1, D_2, D_3, D_4, D_5, D_6, A_1, A_2, A_3, A_4, A_5, A_6\}_{j,i}$$

ou, simbolicamente

$$\{V\}_{j,i} = \{D, A\}_{j,i}$$

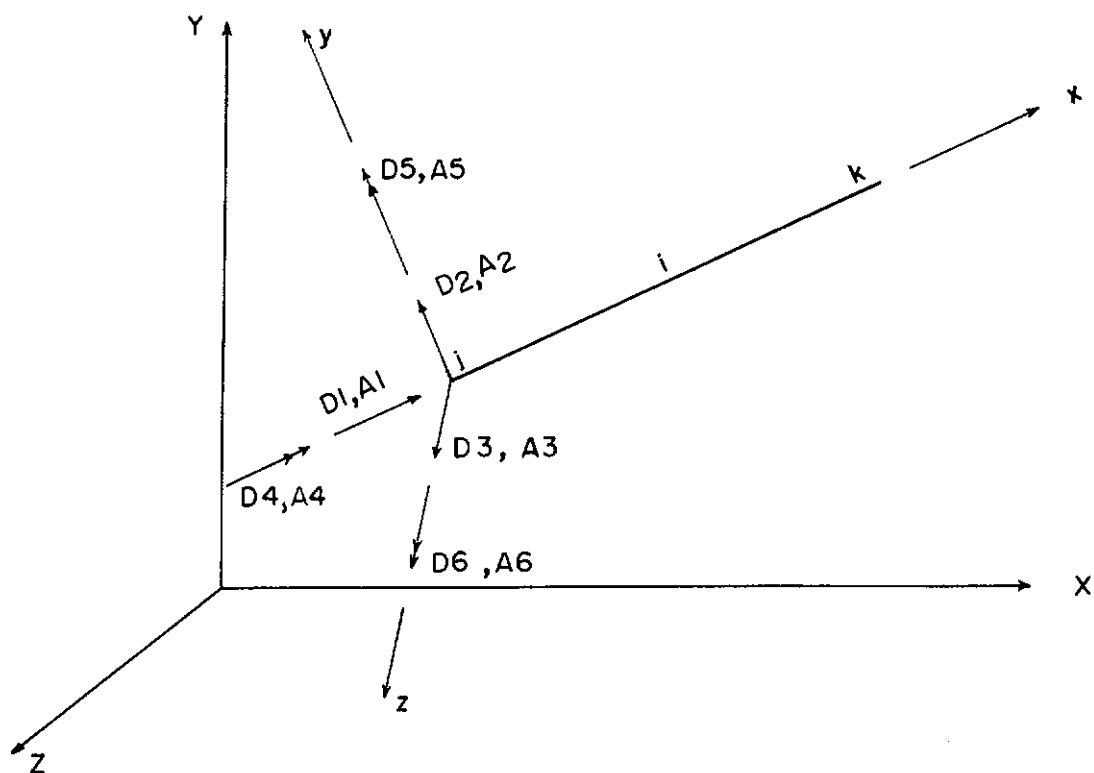


Fig. 2.1

Numa junta o vetor de estado é referido ao sistema global, conforme mostra a figura 2.2. A notação usada é simplesmente $\{V\}_j$, indicando tratar-se do vetor de estado da junta j :

$$\{V\}_j = \{D, A\}_j$$

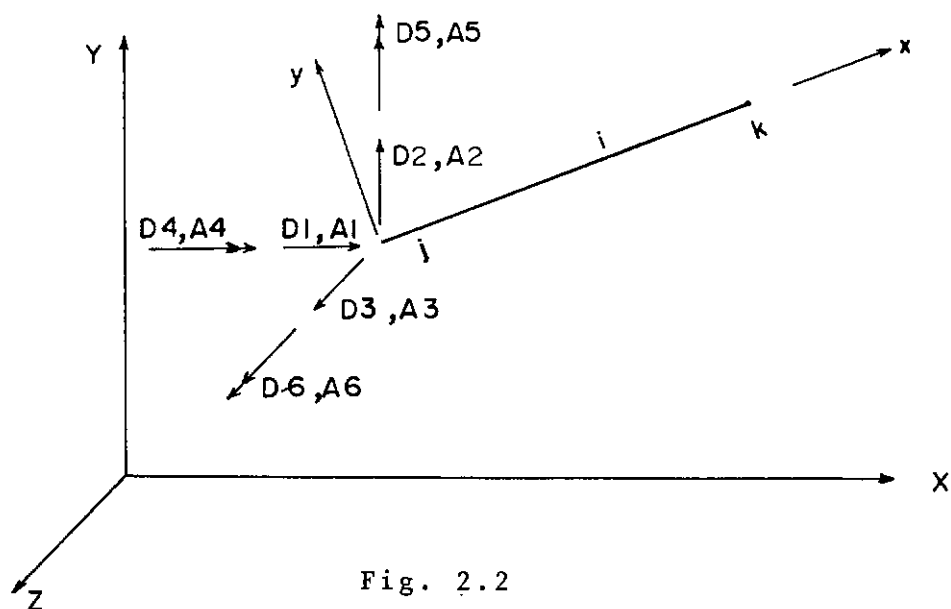


Fig. 2.2

2.2 - Matriz de transferência:

Conhecido o vetor de estado na seção j da barra i da figura 2.3, deseja-se determinar o vetor de estado na seção k dessa barra. O operador que permite estabelecer o vetor na seção k em função do vetor na seção j é chamado operador de transferência porque ele transfere as propriedades de uma seção para outra.

Seja $[T]_i$ esse operador, de modo que:

$$\{V\}_{k,i} = [T]_i \{V\}_{j,i} \quad (2.1)$$

Naturalmente, o operador $[T]$ tem tantas linhas e colunas quantos forem os elementos do vetor de estado.

Resulta, portanto, que o operador é uma matriz quadrada e é conhecido como matriz de transferência.

A equação matricial (2.1) é a equação fundamental do método das matrizes de transferência. Mostrar-se-á nos capítulos seguintes que os elementos da matriz $[T]_i$ são funções das características geométricas e mecânicas da barra i .

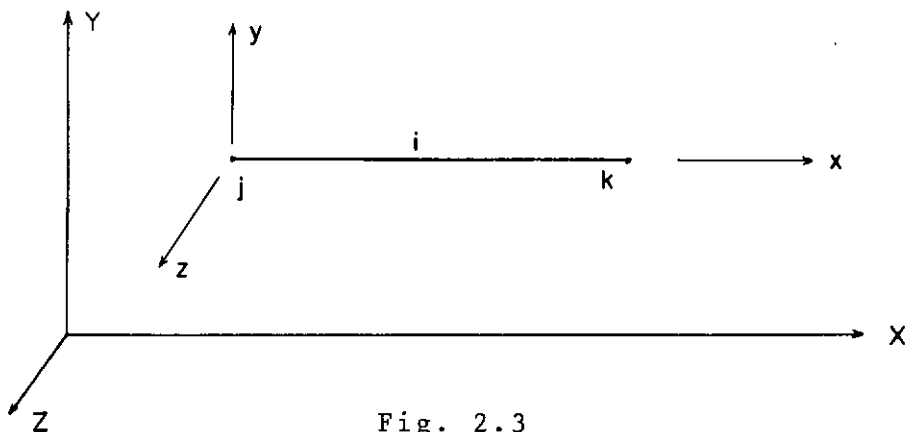


Fig. 2.3

2.3 - Generalização para várias barras:

Considerando-se a barra i da figura 2.4 formada por uma sucessão de n barras de modo a não haver nenhuma descontinuidade entre elas, a aplicação da equação (2.1) a todas essas n barras, conduz a

$$\{V\}_{k,1} = [T]_1 \{V\}_{j,1} \quad (2.2)$$

$$\{V\}_{k,2} = [T]_2 \{V\}_{j,2} \quad (2.3)$$

$$\{V\}_{k,3} = [T]_3 \{V\}_{j,3} \quad (2.4)$$

$$\{V\}_{k,n-1} = [T]_{n-1} \{V\}_{j,n-1} \quad (2.5)$$

$$\{V\}_{k,n} = [T]_n \{V\}_{j,n} \quad (2.6)$$

Mas, não havendo descontinuidade entre as barras é claro que

$$\{V\}_{j,2} = \{V\}_{k,1} \quad (2.7)$$

$$\{V\}_{j,3} = \{V\}_{k,2} \quad (2.8)$$

$$\{V\}_{j,n} = \{V\}_{k,n-1} \quad (2.9)$$

Tendo em vista (2.7), as equações (2.2) e (2.3) permitem escrever:

$$\{V\}_{k,2} = [T]_2 [T]_1 \{V\}_{j,1} \quad (2.10)$$

Substituindo (2.8) em (2.4):

$$\{V\}_{k,3} = [T]_3 \{V\}_{k,2} \text{ ou, conforme } (2.10)$$

$$\{V\}_{k,3} = [T]_3 [T]_2 [T]_1 \{V\}_{j,1}$$

Finalmente, após fazer todas as substituições:

$$\{V\}_{k,n} = [T]_n [T]_{n-1} \dots [T]_3 [T]_2 [T]_1 \{V\}_{j,1}$$

Mas, $\{V\}_{j,1} = \{V\}_{j,i}$ e $\{V\}_{k,n} = \{V\}_{k,i}$, logo:

$$\{V\}_{k,i} = [T]_n [T]_{n-1} \dots [T]_3 [T]_2 [T]_1 \{V\}_{j,i} \quad (2.11)$$

Confrontando agora (2.11) com (2.1), conclui-se que

$$[T]_i = [T]_n [T]_{n-1} \dots [T]_3 [T]_2 [T]_1 \quad (2.12)$$

isto é, a matriz de transferência da barra i é igual ao produto das matrizes de transferência das n barras que a formam, conforme (2.12).

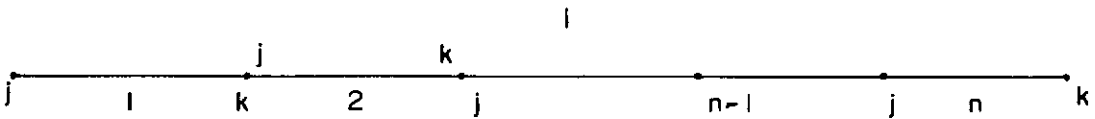


Fig. 2.4

2.4 - Matriz fronteira:

Admitiu-se no item anterior a inexistência de descontinuidades entre as n barras. Considere-se agora a figura 2.5 onde se admitem descontinuidades em todas as seções.

As matrizes de transferência das n barras são também conhecidas podendo-se escrever:

$$\{V\}_{k,1} = [T]_1 \{V\}_{j,1} \quad (2.13)$$

$$\{V\}_{k,2} = [T]_2 \{V\}_{j,2} \quad (2.14)$$

$$\{V\}_{k,3} = [T]_3 \{V\}_{j,3} \quad (2.15)$$

$$\{V\}_{k,n} = [T]_n \{V\}_{j,n} \quad (2.16)$$

É necessário agora definir um operador capaz de relacionar os vetores de estado entre as descontinuidades, da forma:

$$\{V\}_{j,2} = \{VF\}_2 + \{V\}_{k,1} \quad (2.17)$$

O vetor $\{VF\}_2$ é chamado vetor fronteira da seção 2 e permite estabelecer as condições de compatibilidade de deslocamento e de equilíbrio na seção.. As equações (2.13), (2.14) e (2.17) permitem escrever:

$$\{V\}_{k,2} = [T]_2 (\{VF\}_2 + [T]_1 \{V\}_{j,1}) \quad (2.18)$$

Mas $\{V\}_{j,3} = \{VF\}_3 + \{V\}_{k,2}$ ou

$\{V\}_{j,3} = \{VF\}_3 + [T]_2 (\{VF\}_2 + [T]_1 \{V\}_{j,1})$ e a equação (2.15) pode ser escrita na forma:

$$\{V\}_{k,3} = [T]_3 (\{VF\}_3 + [T]_2 (\{VF\}_2 + [T]_1 \{V\}_{j,1}))$$

Procedimento análogo para as barras restantes, fornece finalmente:

$$\{V\}_{k,n} = [T]_n (\{VF\}_n + [T]_{n-1} (\{VF\}_{n-1} + \dots [T]_3 (\{VF\}_3 + [T]_2 (\{VF\}_2 + [T]_1 \{V\}_{j,1}))) \dots) \quad (2.19)$$

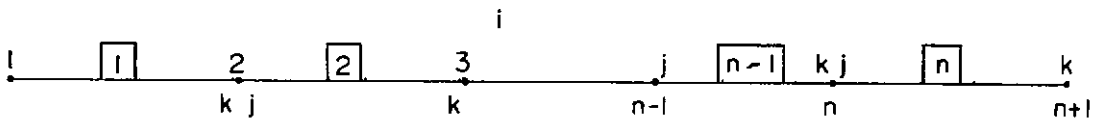


Fig. 2.5

Muitas vezes, ao invés de definir o vetor fronteira como em (2.17), é mais interessante definir a matriz fronteira $[F]$:

$$\{V\}_{j,2} = [F]_2 \{V\}_{k,1} \quad (2.20)$$

Desta forma, a substituição de (2.20) nas equações (2.13) e seguintes, conduz à equação

$$\{V\}_{k,n} = [T]_n [F]_n [T]_{n-1} [F]_{n-1} \dots [T]_3 [F]_3 [T]_2 [F]_2 [T]_1 \{V\}_{j,1}$$

Como $\{V\}_{k,n} = \{V\}_{k,i}$ e $\{V\}_{j,1} = \{V\}_{j,i}$ e tendo-se em conta a equação (2.1):

$$\{V\}_{k,i} = [\tilde{T}]_i \{V\}_{j,i} \quad \text{onde}$$

$$[\tilde{T}]_i = [T]_n [F]_n [T]_{n-1} [F]_{n-1} \dots [T]_3 [F]_3 [T]_2 [F]_2 [T]_1. \quad (2.21)$$

Assim, a matriz de transferência total para a estrutura da figura (2.5), onde há descontinuidade, pode ser obtida por produtos sucessivos das matrizes de transferência e das matrizes fronteiras, conforme (2.21).

CAPÍTULO III

OBTENÇÃO DAS MATRIZES DE TRANSFERÊNCIA

As matrizes de transferência podem ser obtidas de três maneiras diferentes:

- pela integração da equação diferencial;
- pela aplicação das equações da Resistência dos Materiais;
- através da matriz de rigidez.

Neste capítulo, com o objetivo de facilitar a exposição, considera-se um sistema homogêneo constituído de uma barra de viga contínua.

Nos capítulos seguintes serão tratados os sistemas não homogêneos, constituídos de barras de viga contínua, viga sobre base elástica, pórticos planos e pórticos no espaço.

3.1 - Integração da equação diferencial:

A viga representada na figura 3.1 deve satisfazer a equação diferencial:

$$\frac{d^2}{dx^2} (EI_z \frac{d^2 y}{dx^2}) = 0 \quad \text{ou}$$

$$EI_z \frac{d^4 y}{dx^4} = 0 \quad \text{com a hipótese de } EI_z \text{ constante.}$$

Integrando quatro vezes, obtem-se:

$$EI_z \frac{d^3 y}{dx^3} = C_1 \quad (3.1)$$

$$EI_z \frac{d^2 y}{dx^2} = C_1 x + C_2 \quad (3.2)$$

$$EI_z \frac{dy}{dx} = C_1 \frac{x^2}{2} + C_2 \cdot x + C_3 \quad (3.3)$$

$$EI_z y = C_1 \frac{x^3}{6} + C_2 \frac{x^2}{2} + C_3 x + C_4 \quad (3.4)$$

A equação (3.1) representa a força cortante, a qual para $x=0$ vale Q_j , portanto:

$$C_1 = Q_j$$

A equação (3.2) representa o momento fletor o qual para $x=0$ é M_j , portanto:

$$C_2 = M_j$$

Na equação (3.3), $\frac{dy}{dx} = \theta$ cujo valor para $x=0$ é θ_j , portanto:

$$C_3 = EI_z \cdot \theta_j$$

Finalmente, para $x=0$ obtem-se $C_4 = EI_z \cdot y_j$ na equação (3.4)

Substituindo as constantes encontradas nas equações (3.1) a (3.4) e fazendo $x=l$, obtem-se o sistema (3.5)

$$Q_k = Q_j$$

$$M_k = Q_j \cdot l + M_j$$

$$\theta_k = Q_j \cdot \frac{l^2}{2EI_z} + M_j \cdot \frac{l}{EI_z} + \theta_j \quad (3.5)$$

$$y_k = Q_j \cdot \frac{\ell^3}{6EI_z} + M_j \cdot \frac{\ell^2}{2EI_z} + \theta_j \cdot \ell + y_j$$

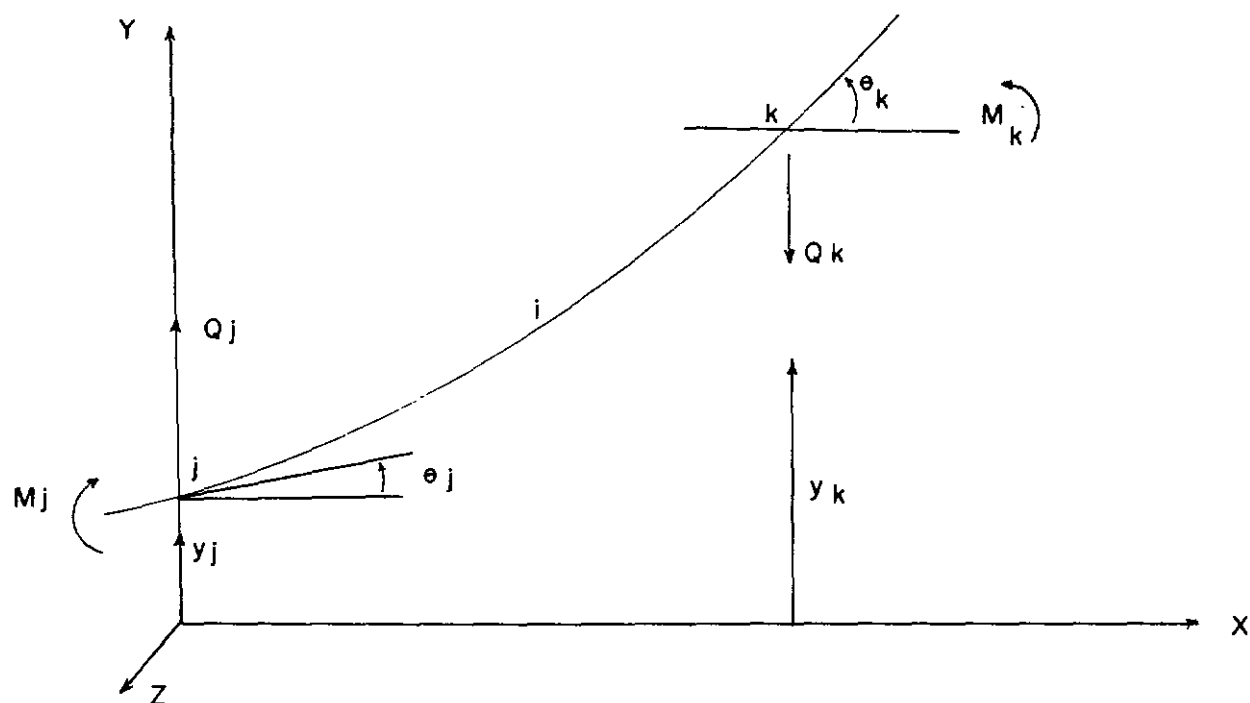


Fig. 3.1

Adotando como referência o triedro direto da figura 3.2, o sistema (3.5) será:

$$Q_k = - Q_j$$

$$M_k = \ell \cdot Q_j - M_j$$

(3.6)

$$\theta_k = \theta_j + \frac{\ell^2}{2EI_z} Q_j - \frac{\ell}{EI_z} M_j$$

$$y_k = y_j + \ell \cdot \theta_j + \frac{\ell^3}{6EI_z} Q_j - \frac{\ell^2}{2EI_z} M_j$$

Considerando os vetores de estado:

$$\{V\}_{j,i} = \{y, \theta, Q, M\}_{j,i}$$

$$\{V\}_{k,i} = \{y, \theta, Q, M\}_{k,i}$$

o sistema (3.6) toma a forma matricial

$$\begin{vmatrix} Y \\ \theta \\ Q \\ M_{K,i} \end{vmatrix} = \begin{vmatrix} 1 & \ell & \ell^3/6EI_z & -\ell^2/2EI_z \\ 0 & 1 & \ell^2/2EI_z & -\ell/EI_z \\ 0 & 0 & -1 & 0 \\ 0 & 0 & \ell & -1 \end{vmatrix} \begin{vmatrix} Y \\ \theta \\ Q \\ M_{j,i} \end{vmatrix} \quad (3.7)$$

Simbolicamente

$$\{V\}_{k,i} = [T]_i \{V\}_{j,i}$$

onde $[T]_i$ representa a matriz de transferência da viga da figura 3.2

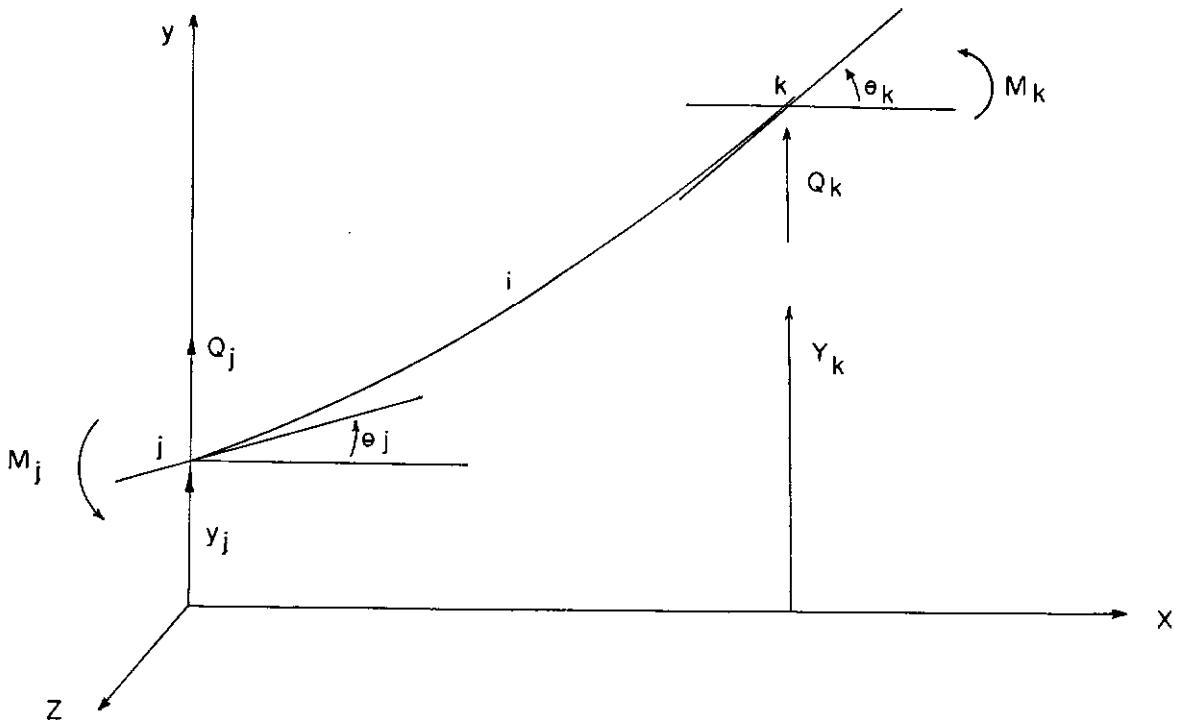


Fig. 3.2

3.2 - Equações da Resistência dos Materiais:

Aplicando as equações de equilíbrio da Estática à viga da figura 3.2, obtém-se:

$$\Sigma F_y = 0: \quad Q_j + Q_k = 0$$

$$\Sigma M_z = 0: \quad M.x + M_j - x.Q_j = 0$$

Portanto, $Q_k = -Q_j$ (3.8)

$$M.x = x.Q_j - M_j \quad (3.9)$$

Da Resistência dos Materiais, sabe-se que:

$$\frac{d}{dx} \left(\frac{dy}{dx} \right) = \frac{d\theta}{dx} = \frac{M}{EI_z}$$

Então, $d\theta = \frac{M}{EI_z} dx$, cuja integral no intervalo $[0, x]$ é

$$\int_0^x d\theta = \theta_x - \theta_0 = \int_0^x \frac{(x.Q_j - M_j)}{EI_z} dx \quad \text{ou}$$

$$\theta_x - \theta_0 = \frac{x^2}{2EI_z} Q_j - \frac{x}{EI_z} M_j \quad (3.10)$$

Mas $\frac{dy}{dx} = \theta$ ou $dy = \theta dx$, cuja integral é

$$\int_0^x dy = y_x - y_0 = \int_0^x \left(\theta_j + \frac{x^2}{2EI_z} Q_j - \frac{x}{EI_z} M_j \right) dx$$

$$\text{ou } y_x = y_j + x\theta_j + \frac{x^3}{6EI_z} Q_j - \frac{x^2}{2EI_z} M_j \quad (3.11)$$

Pondo $x=l$ nas equações (3.8) a (3.11) obtém-se, finalmente:

$$Q_k = -Q_j$$

$$M_k = l.Q_j - M_j$$

$$\theta_k = \theta_j + \frac{l^2}{2EI_z} Q_j - \frac{l}{EI_z} M_j$$

$$y_k = y_j + l.\theta_j + \frac{l^3}{6EI_z} Q_j - \frac{l^2}{2EI_z} M_j$$

sistema exatamente igual ao sistema (3.6) obtido anteriormente.

3.3 - Através da matriz de rigidez:

Seja a viga da figura 3.3 onde se indicam os vetores de estado nas seções j e K:

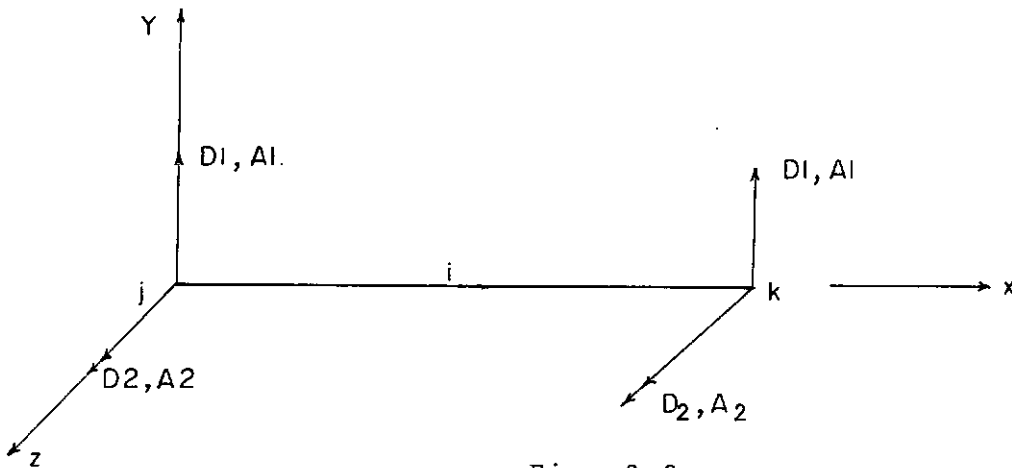


Fig. 3.3

$$\{V\}_{j,i} = \{D_1, D_2, A_1, A_2\}_{j,i} = \{D, A\}_{j,i}$$

onde $\{A\}_j = \{A_1, A_2\}_j$ e $\{D\}_j = \{D_1, D_2\}_j$

Relações idênticas subsistem para a seção K.

A equação fundamental do método das matrizes de transferência

$$\{V\}_{k,i} = [T]_i \{V\}_{j,i}$$

pode ser colocada em forma de submatrizes do seguinte modo:

$$\begin{bmatrix} D \\ \dots \\ A \end{bmatrix}_{k,i} = \begin{bmatrix} T_{DD} & \dots & T_{DA} \\ T_{AD} & & T_{AA} \end{bmatrix} \begin{bmatrix} D \\ \dots \\ A \end{bmatrix}_{j,i} \quad (3.12)$$

em que $|T_{DD}| = \begin{bmatrix} 1 & \ell \\ 0 & 1 \end{bmatrix}$ e $|T_{AA}| = \begin{bmatrix} -1 & 0 \\ \ell & -1 \end{bmatrix}$

$$|T_{AD}| = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \text{ e } |T_{DA}| = \begin{bmatrix} \ell^3/6EI_z & -\ell^2/2EI_z \\ \ell^2/2EI_z & -\ell/EI_z \end{bmatrix}$$

conforme a equação (3.7)

A equação fundamental do método de rigidez ou dos deslocamentos:

$$\{A\} = [S] \{D\}$$

também, em forma de submatrizes é:

$$\begin{bmatrix} A_j \\ \dots \\ A_k \end{bmatrix} = \begin{bmatrix} S_{jj} & S_{jk} \\ \dots & \dots \\ S_{kj} & S_{kk} \end{bmatrix} \begin{bmatrix} D_j \\ \dots \\ D_k \end{bmatrix} \quad (3.13)$$

onde S é a matriz de rigidez referida, também, à viga da figura 3.3.

Desenvolvendo (3.13) obtem-se as equações:

$$\{A\}_j = [S_{jj}] \{D\}_j + [S_{jk}] \{D\}_k \quad (3.14)$$

$$\{A\}_k = [S_{kj}] \{D\}_j + [S_{kk}] \{D\}_k \quad (3.15)$$

Explicitando $\{D\}_k$ da equação (3.14)

$$\{D\}_k = -[S_{jk}]^{-1} [S_{jj}] \{D\}_j + [S_{jk}]^{-1} \{A\}_j \quad (3.16)$$

Levando (3.16) em (3.15) obtem-se:

$$\{A\}_k = |S_{kj}| \{D\}_j + |S_{kk}| (-|S_{jk}|^{-1} |S_{jj}| \{D\}_j + |S_{jk}|^{-1} \{A\}_j)$$

$$\{A\}_k = (|S_{kj}| - |S_{kk}| |S_{jk}|^{-1} |S_{jj}|) \{D\}_j + |S_{kk}| |S_{jk}|^{-1} \{A\}_j \quad (3.17)$$

As equações (3.16) e (3.17) permitem escrever:

$$\begin{bmatrix} D \\ \vdots \\ A \end{bmatrix}_k = \begin{bmatrix} -|S_{jk}|^{-1} |S_{jj}| & \cdots & |S_{jk}|^{-1} \\ \vdots & \ddots & \vdots \\ |S_{kj}| - |S_{kk}| |S_{jk}|^{-1} |S_{jj}| & \cdots & |S_{kk}| |S_{jk}|^{-1} \end{bmatrix} \begin{bmatrix} D \\ \vdots \\ A \end{bmatrix}_j \quad (3.18)$$

As equações (3.12) e (3.18) permitem concluir que:

$$|T_{DD}| = - [S_{jk}]^{-1} [S_{jj}] \quad (3.19)$$

$$|T_{DA}| = [S_{jk}]^{-1} \quad (3.20)$$

$$|T_{AD}| = [S_{kj}] - [S_{kk}] [S_{jk}]^{-1} [S_{jj}] \quad (3.21)$$

$$|T_{AA}| = [S_{kk}] [S_{jk}]^{-1} \quad (3.22)$$

Como se sabe, as submatrizes de rigidez das relações (3.19)a (3.22) são:

$$[S_{jj}] = \begin{bmatrix} 12EI_z/\ell^3 & 6EI_z/\ell^2 \\ 6EI_z/\ell^2 & 4EI_z/\ell \end{bmatrix}$$

$$[S_{jk}] = [S_{kj}] = \begin{bmatrix} -12EI_z/\ell^3 & 6EI_z/\ell^2 \\ -6EI_z/\ell^2 & 2EI_z/\ell \end{bmatrix}$$

$$[S_{kk}] = \begin{bmatrix} 12EI_z/\ell^3 & -6EI_z/\ell^2 \\ -6EI_z/\ell^2 & 4EI_z/\ell \end{bmatrix}$$

Substituindo-as naquelas relações e efetuando os cálculos matriciais, obtém-se finalmente:

$$|T_{DD}| = \begin{bmatrix} 1 & \ell \\ 0 & 1 \end{bmatrix}$$

$$|T_{DA}| = \begin{bmatrix} \ell^3/6EIz & -\ell^2/2EIz \\ \ell^2/2EIz & -\ell/EIz \end{bmatrix}$$

$$[T_{AD}] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad e \quad [T_{AA}] = \begin{bmatrix} -1 & 0 \\ \ell & -1 \end{bmatrix}$$

que coincidem exatamente com (3.12).

CAPÍTULO IV

MATRIZ DE TRANSFERÊNCIA DE PÓRTICOS E VIGAS

A matriz de transferência pode ser obtida de três maneiras diferentes, como se mostrou no capítulo anterior.

Trata-se agora de obter as matrizes de transferência para as barras das estruturas analisadas neste trabalho, vigas contínuas e sobre base elástica, pórticos planos e no espaço, incluindo-se os efeitos produzidos pelas cargas que atuam nessas barras.

A integração direta da equação diferencial é usada apenas para a viga sobre base elástica enquanto que as equações elementares da Resistência dos Materiais são usadas para as demais estruturas, por serem mais simples e objetivas.

As matrizes de transferência dos pórticos, referidas ao sistema global, serão obtidas quando da análise do conjunto de elementos no capítulo VI.

4.1 - Viga Contínua:

Seja a viga representada na figura 4.1 submetida as cargas concentradas P e M de abcissa α e $\tilde{\alpha}$ carga distribuída $p(x)$ ao longo de toda a barra.

Sejam $\{V\}_{j,i} = \{y, \theta, Q, M\}_{j,i}$ e $\{V\}_{s,i} = \{y, \theta, Q, M\}_{s,i}$ os vetores de estado na origem e na seção genérica S de abcissa x ,

respectivamente.

Seja ainda $\{V\} = \{D, A\}$ a representação simbólica do vetor de estado.

Tomando-se a soma das forças segundo y na seção S :

$$\begin{aligned} Q_s + Q_j + P + \int_0^x p(x) dx &= 0 & \text{ou} \\ Q_s &= -Q_j - P - \int_0^x p(x) dx \end{aligned} \quad (4.1)$$

A equação de equilíbrio do momento fletor em S

$$M_s + M_j - Q_j \cdot x - P(x-\alpha) + M - \int_0^x p(x) \cdot x dx = 0$$

fornece:

$$M_s = -M_j + Q_j \cdot x + P(x-\alpha) - M + \int_0^x p(x) x dx \quad (4.2)$$

Da relação $d\theta = \frac{M_s}{EI_z} dx$, obtem-se a rotação na seção S :

$$\int_0^x d\theta = \theta_s - \theta_j = \frac{1}{EI_z} \int_0^x (-M_j + Q_j \cdot x + P(x-\alpha) - M + \int_0^x p(x) \cdot x dx) dx$$

ou

$$\theta_s = \theta_j - \frac{x}{EI_z} M_j + \frac{x^2}{2EI_z} Q_j + \frac{P(x-\alpha)^2}{2EI_z} - \frac{M(x-\alpha)}{EI_z} + \frac{1}{EI_z} \int_0^x \int_0^x p(x) \cdot x dx^2 \quad (4.3)$$

A translação na seção S obtem-se de: $dy = \theta dx$

ou

$$\int_0^x dy = y_s - y_j = \int_0^x \left(\theta_j - \frac{x}{EIz} M_j + \frac{x^2}{2EIz} Q_j + \frac{P(x-\alpha)^2}{2EIz} - \frac{M(x-\alpha)}{EIz} + \frac{1}{EIz} \int_0^x p(x) \cdot x dx^2 \right) dx$$

ou

$$y_s = y_j + \theta_j \cdot x - \frac{x^2}{2EIz} M_j + \frac{x^3}{6EIz} Q_j + \frac{P(x-\alpha)^3}{6EIz} - \frac{M(x-\alpha)^2}{2EIz} + \frac{1}{EIz} \int_0^x \int_0^x \int_0^x p(x) \cdot x dx^3$$

$$p(x) \cdot x dx^3 \quad (4.4)$$

Para $x=\ell$ nas equações (4.1) a (4.4), obtem-se a matriz de transferência para o sistema não homogêneo apresentado na figura 4.1.

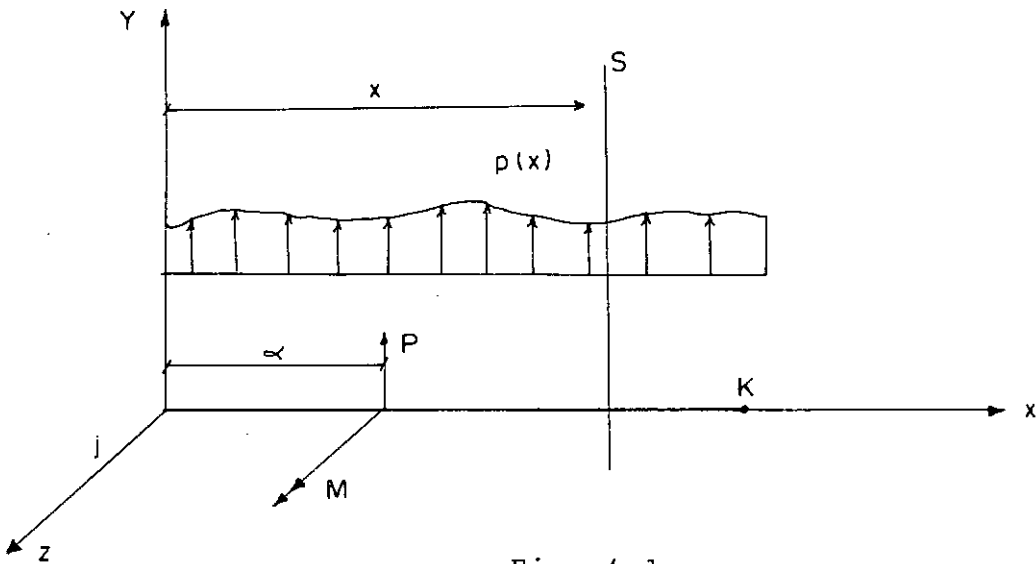


Fig. 4.1

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} = \begin{vmatrix} T_{DD} & T_{DA} & T_{DC} \\ T_{AD} & T_{AA} & T_{AC} \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i} \quad (4.5)$$

As submatrizes $|T_{DD}|$, $|T_{DA}|$, $|T_{AD}|$ e $|T_{AA}|$ são as mesmas já obtidas em (3.12). A equação (4.5) mostra que a matriz de transferência dos sistemas homogêneos, equação (3.12) foi acrescida de uma colu

na para a introdução do vetor de cargas.

$$\{T_c\} = \{T_{DC} \begin{matrix} \vdots \\ T_{AC} \end{matrix} \begin{matrix} \vdots \\ 1 \end{matrix}\} \quad \text{onde}$$

$$\{T_{DC}\} = \begin{vmatrix} \frac{P(-\alpha)^3}{6EIz} - \frac{M(\ell-\alpha)^2}{2EIz} + \frac{1}{EIz} & \int_0^\ell \int_0^\ell \int_0^\ell p(x) \cdot x dx^3 \\ 0 & 0 & 0 \\ \frac{P(\ell-\alpha)^2}{2EIz} - \frac{M(\ell-\alpha)}{EIz} + \frac{1}{EIz} & \int_0^\ell \int_0^\ell p(x) \cdot x dx^2 \\ 0 & 0 \end{vmatrix} \quad (4.6)$$

$$\{T_{AC}\} = \begin{vmatrix} -P - \int_0^\ell p(x) dx \\ 0 \\ P(\ell-\alpha) - M + \int_0^\ell p(x) \cdot x dx \end{vmatrix}$$

Para obedecer às regras do cálculo matricial, ela foi também acrescentada de uma linha, e, o vetor de estado numa seção qualquer passa a ser:

$$\{V\}_{s,i} = \{D \begin{matrix} \vdots \\ A \end{matrix} \begin{matrix} \vdots \\ 1 \end{matrix}\}_{s,i}$$

Porém, poder-se-ia escrever a equação (4.5) na forma:

$$\left| \frac{D}{A} \right|_{k,i} = \begin{vmatrix} T_{DD} & T_{DA} \\ T_{AD} & T_{AA} \end{vmatrix} \left| \frac{D}{A} \right|_{j,i} + \begin{vmatrix} T_{DC} \\ T_{AC} \end{vmatrix}_{k,i} \quad (4.7)$$

sem a necessidade, portanto, de aumentar uma coluna e uma linha na matriz.

A preferência pela forma (4.5) reside no fato de que ela possibilita a análise da estrutura somente com produtos de matrizes,

do mesmo modo como foi mostrado na introdução da matriz fronteira no capítulo II.

Se $p(x)$ for uma carga linearmente distribuída de taxa q_j e q_k nas extremidades j e k , então:

$$p(x) = q_j + \left(\frac{q_k - q_j}{\ell} \right) x$$

Substituindo $p(x)$ em (4.6) e integrando, obtem-se (4.8).

$$\{T_{DC}\} = \left\{ \begin{array}{l} \frac{P(\ell-\alpha)^3}{6EIz} - \frac{M(\ell-\alpha)^2}{2EIz} + \frac{(4qj+qk) \cdot \ell^4}{120EIz} \\ \frac{P(\ell-\alpha)^2}{2EIz} - \frac{M(\ell-\alpha)}{EIz} + \frac{(3qj+qk) \cdot \ell^3}{24EIz} \end{array} \right\} \quad (4.8)$$

$$\{ T_{AC} \} = \begin{vmatrix} -P - (qj + qk) \cdot l / 2 \\ P(l - \alpha) - M + (2qj + qk) l^2 / 6 \end{vmatrix}$$

4.2 - Pórtico plano:

Além das deformações de flexão inclui-se na análise de pórtico plano a deformação oriunda da força normal.

Assim, o vetor de estado numa seção qualquer será:

$$\{V\} = \{x, y, \theta, N, Q, M, 1\}$$

onde x representa a deformação axial e N a força associada, como ilustra a figura 4.2

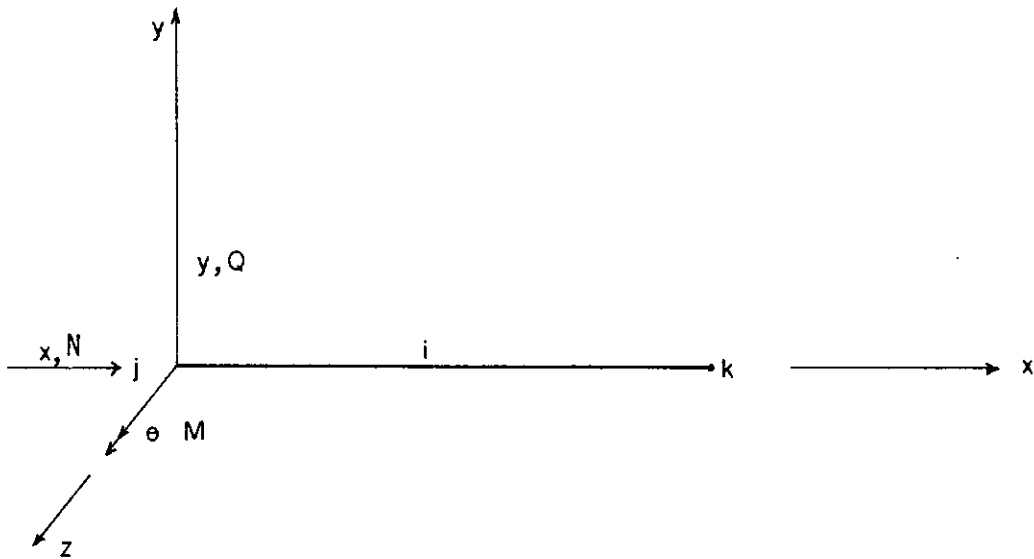


Fig. 4.2

Basta incluir, portanto, na equação (4.5) de viga contínua o efeito da deformação axial e considerar, naturalmente, a influência das cargas que atuam na barra segundo o eixo x .

As cargas consideradas são as indicadas na figura 4.3 onde $p(x)$ é linearmente distribuído, P concentrada de abscissa a e ΔT a variação de temperatura, sendo α o coeficiente de dilatação térmica do material da barra.

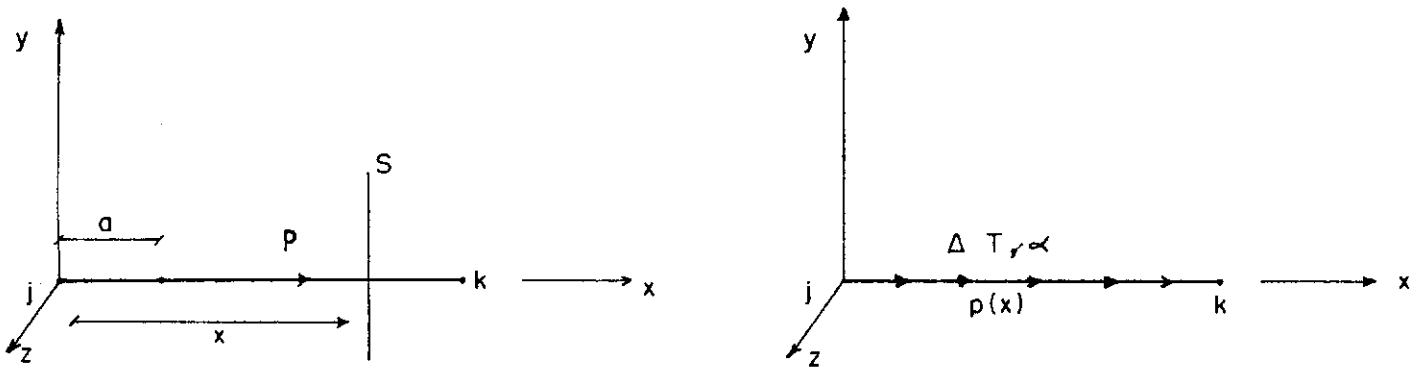


Fig. 4.3

Nestas condições, a equação de equilíbrio na seção S, de abscissa x, é:

$$\begin{aligned} N_s + N_j + P + \int_0^x p(x)dx &= 0 \quad \text{ou} \\ N_s &= -N_j - P - \int_0^x p(x)dx \end{aligned} \quad (4.9)$$

Pela lei de Hooke, a deformação axial vale:

$$d\Delta = \epsilon dx = \frac{\sigma}{E} dx = \frac{N_s}{EAx} dx$$

Considerando a variação de temperatura e 4.9, tem-se:

$$\begin{aligned} \int_0^x d\Delta = x_s - x_j &= \int_0^x \frac{N_s}{EAx} dx + \alpha \Delta T \cdot x, \quad \text{ou} \\ x_s &= x_j - \frac{x}{EAx} N_j - \frac{P(x-a)}{EAx} + \alpha \Delta T \cdot x - \frac{1}{EAx} \int_0^x \int_0^x p(x)dx^2 \end{aligned} \quad (4.10)$$

Como $p(x) = q_j + \left(\frac{q_k - q_j}{\ell}\right) x$, as equações (4.9) e (4.10) podem ser escritas, após a integração, e com $x = \ell$:

$$N_k = -N_j - P - \frac{(q_j + q_k) \cdot \ell}{2} \quad (4.11)$$

$$x_k = x_j - \frac{\ell}{EAx} N_j - \frac{P(\ell - a)}{EAx} + \alpha \Delta T \cdot \ell - \frac{\ell^2}{6EAx} (2q_j + q_k) \quad (4.12)$$

As equações (4.5), (4.11) e (4.12) podem ser colocadas na forma matricial, de acordo com o vetor de estado:

$$\{V\} = \{x, y, \theta, N, Q, M, 1\} = \{D, A, 1\}$$

da seguinte maneira:

$$\begin{bmatrix} D \\ \dots \\ A \\ \dots \\ 1 \end{bmatrix}_{k,i} = \begin{bmatrix} T_{DD} & T_{DA} & T_{DC} \\ \dots & \dots & \dots \\ T_{AD} & T_{AA} & T_{AC} \\ \dots & \dots & \dots \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} D \\ \dots \\ A \\ \dots \\ 1 \end{bmatrix}_{j,i} \quad (4.13)$$

onde

$$|T_{DD}| = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \ell \\ 0 & 0 & 1 \end{bmatrix} \quad |T_{AD}| = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$|T_{DA}| = \begin{bmatrix} -\ell/EAx & 0 & 0 \\ 0 & \ell^3/6EIz & -\ell^2/2EIz \\ 0 & \ell^2/2EIz & -\ell/EIz \end{bmatrix}$$

$$|T_{AA}| = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & \ell & -1 \end{bmatrix}$$

$|T_{DC}|$ e $|T_{AC}|$ são as componentes do vetor de cargas $|T_C| = |T_{DC}, T_{AC}, 1|$ obtidas das relações (4.8) e com a inclusão dos termos de carga das equações (4.11) e (4.12).

4.3 - Pórtico no espaço:

O vetor de estado numa seção qualquer de uma barra de pórtico no espaço é:

$$\{V\} = \{x, y, z, \phi, \theta_y, \theta_z, N, Q_y, Q_z, M_t, M_y, M_z, 1\} = \{D, A, 1\}$$

e se encontra representado na figura 4.4, na extremidade j da barra i .

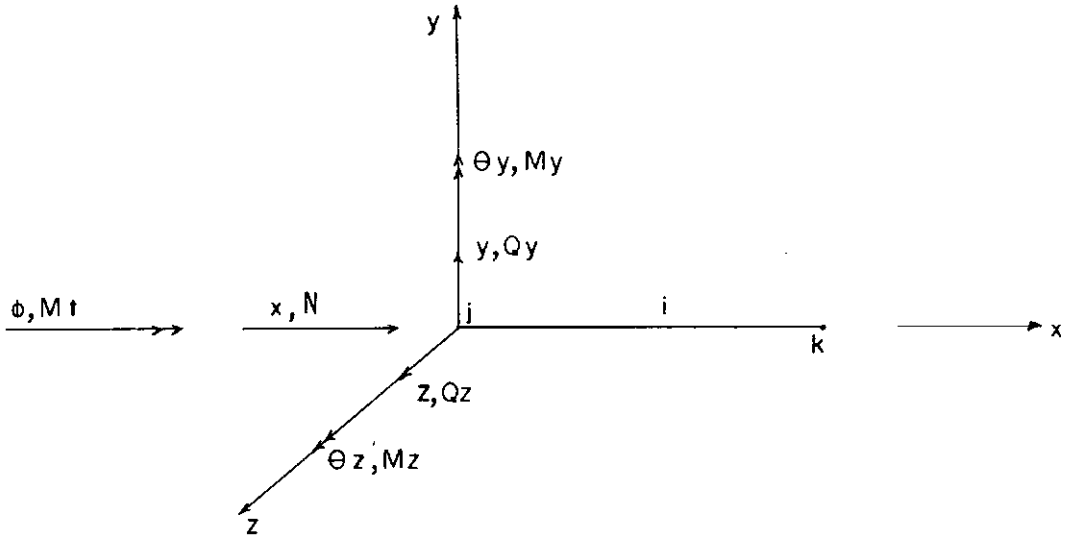


Fig. 4.4

Dessas componentes, apenas ϕ e M_t são novidades em relação ao vetor de estado de pórtico plano, porque representam a deformação de torção e o momento torsor, respectivamente.

As componentes z , Q_z e θ_y , M_y podem ser consideradas através da permutação cíclica dos eixos y e z . Assim sendo, há que considerar agora apenas o efeito de torção e as cargas associadas.

As cargas estão representadas na figura 4.5 onde $\underline{M}$ é o momento concentrado e $\underline{m}$ o momento uniformemente distribuído.

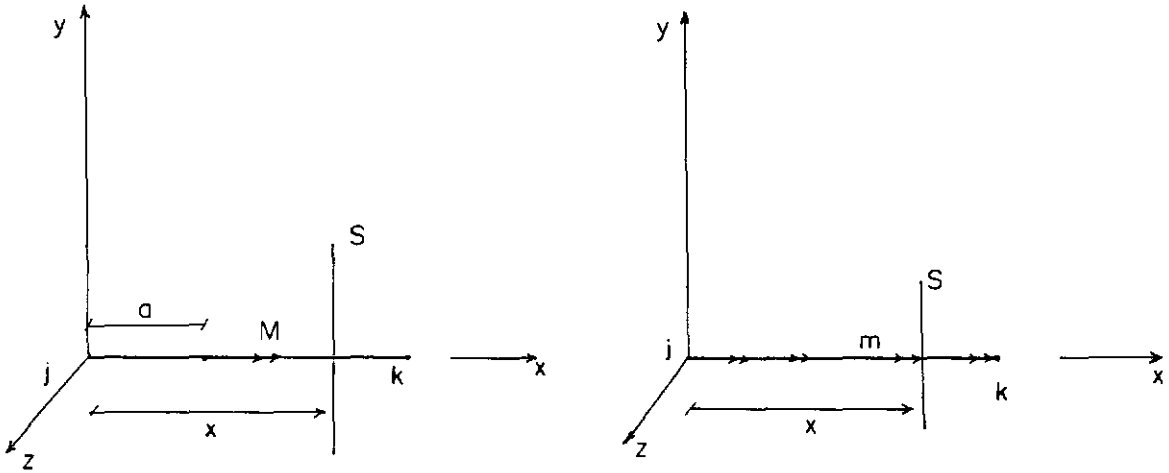


Fig. 4.5

Na seção S há equilíbrio quando

$$M_{ts} + M_{tj} + M + \int_0^x m dx = 0 \quad \text{ou}$$

$$M_{ts} = -M_{tj} - M - m \cdot x \quad (4.14)$$

Da Resistência dos Materiais $d\phi = \frac{M_t}{GIx} dx$

$$\int_0^x d\phi = \phi_s - \phi_j = \int_0^x \frac{(-M_{tj} - M - mx)}{GIx} dx \quad \text{ou}$$

$$\phi_s = \phi_j - \frac{x}{GIx} M_{tj} - \frac{(x-a)}{GIx} M - \frac{mx^2}{2GIx} \quad (4.15)$$

Pondo $x=\ell$ nas equações (4.14) e (4.15), tem-se:

$$M_{tk} = -M_{tj} - M - m \cdot \ell \quad (4.16)$$

$$\phi_k = \phi_j - \frac{\ell}{GIx} M_{tj} - \frac{(\ell-a)}{GIx} M - \frac{m\ell^2}{2GIx}$$

Incluindo-se as equações (4.16) na matriz de pórtico plano (4.13), obedecendo naturalmente o novo vetor de estado, obtem-se a matriz de pórtico no espaço:

$$\begin{bmatrix} D \\ \dots \\ A \\ \dots \\ 1 \end{bmatrix}_{k,i} = \begin{bmatrix} T_{DD} & T_{DA} & T_{DC} \\ T_{AD} & T_{AA} & T_{AC} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} D \\ \dots \\ A \\ \dots \\ 1 \end{bmatrix}_{j,i}$$

onde

$$|T_{DD}| = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \ell \\ 0 & 0 & 1 & 0 & -\ell & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}; \quad |T_{AD}| = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$|T_{DA}| = \begin{bmatrix} -\ell/EAx & 0 & 0 & 0 & 0 & 0 \\ 0 & \ell^3/6EIz & 0 & 0 & 0 & -\ell^2/2EIz \\ 0 & 0 & \ell^3/6EIy & 0 & \ell^2/2EIy & 0 \\ 0 & 0 & 0 & -\ell/GIx & 0 & 0 \\ 0 & 0 & -\ell^2/2EIy & 0 & -\ell/EIy & 0 \\ 0 & \ell^2/2EIz & 0 & 0 & 0 & -\ell/EIz \end{bmatrix}$$

$$|T_{AA}| = \begin{vmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & -\ell & 0 & -1 & 0 \\ 0 & \ell & 0 & 0 & 0 & -1 \end{vmatrix}$$

e $\{T_{DC}\}$ e $\{T_{AC}\}$ são os vetores de carga que incluem todas as cargas previstas para viga contínua e pórtico plano.

4.4 - Viga sobre base elástica:

A viga da figura 4.6 está assente inteiramente num meio elástico capaz de reagir proporcionalmente às deformações y ao longo de seu comprimento ℓ .

Considerando um sistema homogêneo e chamando de K o coeficiente de proporcionalidade, constante ao longo da viga, a equação diferencial do sistema é:

$$\frac{d^2}{dx^2} (EIz \frac{d^2 y}{dx^2}) + K \cdot y = 0$$

ou, com EIz constante

$$EIz \frac{d^4 y}{dx^4} + K \cdot y = 0 \quad (4.17)$$

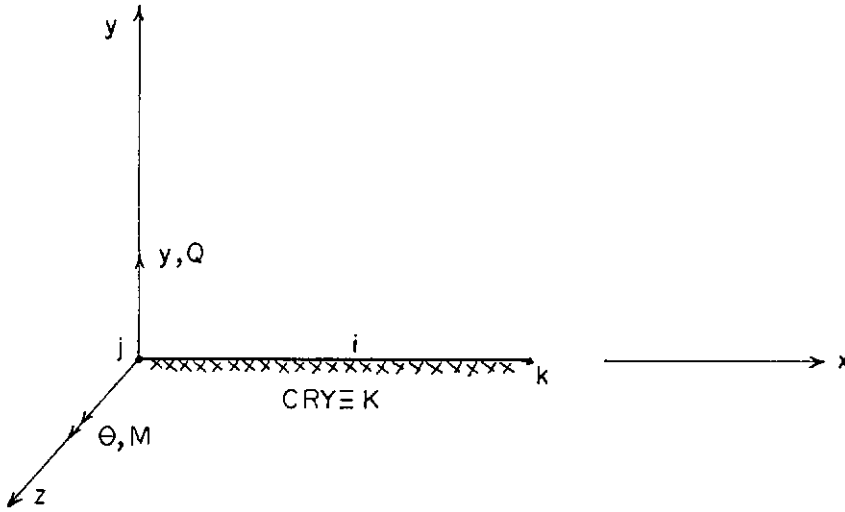


Fig. 4.6

Pondo $\gamma = \sqrt[4]{\frac{K}{4EIz}}$, a solução geral de (4.17) é:

$$y = e^{\gamma x} (A \cos \gamma x + B \sin \gamma x) + e^{-\gamma x} (C \cos \gamma x + D \sin \gamma x) \quad (4.18)$$

Sendo $\{V\}_{j,i} = \{y, \theta, Q, M\}_{j,i}$ o vetor de estado na extremidade j , as constantes de integração A, B, C e D podem ser obtidas em função de suas componentes da seguinte maneira:

$$\text{Para } x=0 \text{ em (4.18), resulta } y_j = A + C \quad (4.19)$$

A derivada de (4.18):

$$\frac{dy}{dx} = \gamma e^{\gamma x} [(A+B) \cos \gamma x + (B-A) \sin \gamma x] + \gamma e^{-\gamma x} [(D-C) \cos \gamma x - (C+D) \sin \gamma x]$$

$$\text{para } x=0, \text{ fornece } \theta_j = (A+B-C+D) \cdot \gamma \quad (4.20)$$

A derivada segunda

$$\frac{d^2 y}{dx^2} = \gamma^2 e^{\gamma x} [2B \cos \gamma x - 2A \sin \gamma x] + \gamma^2 e^{-\gamma x} [2C \sin \gamma x - 2D \cos \gamma x]$$

para $x=0$ é

$$M_j = 2(B-D) \gamma^2 EI_z \quad (4.21)$$

Finalmente, derivando-se uma vez mais obtem-se:

$$\frac{d^3 y}{dx^3} = 2\gamma^3 e^{\gamma x} [(B-A)\cos\gamma x - (A+B)\sin\gamma x] + 2\gamma^3 e^{-\gamma x} [(C+D)\cos\gamma x + (D-C)\sin\gamma x]$$

e, para $x=0$

$$Q_j = 2(B-A+C+D)\gamma^3 EI_z \quad (4.22)$$

A resolução do sistema formado pelas equações (4.19) a (4.22) conduz a

$$A = \frac{1}{2} y_j + \frac{1}{4} \frac{\theta_j}{\gamma} - \frac{1}{8} \frac{Q_j}{\gamma^3 EI_z}$$

$$B = \frac{1}{4} \frac{\theta_j}{\gamma} + \frac{1}{4} \frac{M_j}{\gamma^2 EI_z} + \frac{1}{8} \frac{Q_j}{\gamma^3 EI_z}$$

$$C = \frac{1}{2} y_j - \frac{1}{4} \frac{\theta_j}{\gamma} + \frac{Q_j}{8\gamma^3 EI_z}$$

$$D = \frac{1}{4} \frac{\theta_j}{\gamma} - \frac{1}{4} \frac{M_j}{\gamma^2 EI_z} + \frac{1}{8} \frac{Q_j}{\gamma^3 EI_z}$$

Substituindo essas constantes na equação (4.18) e suas derivadas, com $x = \ell$, obtem-se finalmente:

$$y_k = F_1 \cdot y_j + \frac{F_2}{2\gamma} \theta_j - \frac{F_4}{4\gamma^3 EI_z} Q_j - \frac{F_3}{2\gamma^2 EI_z} M_j$$

$$\theta_k = \gamma F_4 y_j + F_1 \cdot \theta_j + \frac{F_3}{2\gamma EI_z} Q_j - \frac{F_2}{2\gamma EI_z} M_j \quad (4.23)$$

$$Q_k = 2\gamma^3 EI_z F_2 \cdot y_j + 2\gamma^2 EI_z F_3 \theta_j - F_1 Q_j + \gamma F_4 M_j$$

$$M_k = -2\gamma^2 EI_z F_3 y_j + \gamma EI_z F_4 \theta_j + \frac{F_2}{2\gamma} Q_j - F_1 \cdot M_j$$

onde: $F_1 = \text{ch}(\gamma \ell) \cos(\gamma \ell)$

$F_2 = \text{sh}(\gamma \ell) \cos(\gamma \ell) + \text{ch}(\gamma \ell) \text{sen}(\gamma \ell)$

$F_3 = \text{sh}(\gamma \ell) \text{sen}(\gamma \ell)$

$F_4 = \text{sh}(\gamma \ell) \cos(\gamma \ell) - \text{ch}(\gamma \ell) \text{sen}(\gamma \ell)$

Pondo (4.23) na forma matricial:

$$\begin{vmatrix} y \\ \theta \\ Q \\ M \end{vmatrix}_{k,i} = \begin{vmatrix} F_1 & F_2/2\gamma & -F_4/4\gamma^3 EI_z & -F_3/2\gamma^2 EI_z \\ \gamma F_4 & F_1 & F_3/2\gamma EI_z & -F_2/2\gamma EI_z \\ 2\gamma^3 EI_z F_2 & 2\gamma^2 EI_z F_3 & -F_1 & \gamma F_4 \\ -2\gamma^2 EI_z F_3 & EI_z F_4 & F_2/2 & -F_1 \end{vmatrix} \begin{vmatrix} y \\ \theta \\ Q \\ M \end{vmatrix}_{j,i} \quad (4.24)$$

$$\begin{vmatrix} D \\ A \end{vmatrix}_{k,i} = \begin{vmatrix} T_{DD} & T_{DA} \\ T_{AD} & T_{AA} \end{vmatrix} \begin{vmatrix} D \\ A \end{vmatrix}_{j,i} \quad \text{ou } \{V\}_{k,i} = [T] \{V\}_{j,i} \text{ onde}$$

$[T]$ é a matriz de transferência da viga da figura 4.6

Uma solução particular da equação não homogênea

$$EI_z \frac{d^4 y}{dx^4} + K \cdot y = p(x)$$

em que $p(x)$ é a carga linearmente distribuída da figura 4.7

$$p(x) = q_j + \left(\frac{q_k - q_j}{\ell} \right) x \quad \tilde{e}:$$

$$y = e^{\gamma x} (A \cos \gamma x + B \sin \gamma x) + e^{-\gamma x} (C \cos \gamma x + D \sin \gamma x) + \frac{q_j}{K} + \left(\frac{q_k - q_j}{K \cdot \ell} \right) x$$

A determinação das constantes A, B, C e D é feita de maneira análoga à anterior.

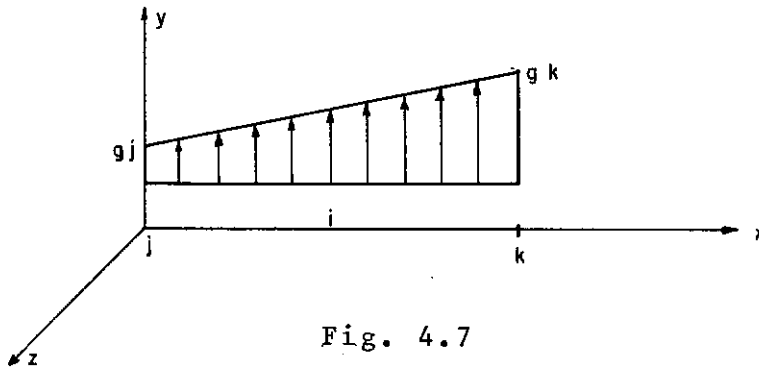


Fig. 4.7

A matriz do sistema não homogêneo será, então, da forma

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} = \begin{vmatrix} T_{DD} & T_{DA} & T_{DC} \\ T_{AD} & T_{AA} & T_{AC} \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i}$$

em que o vetor de carga $\{T_C\} = \{T_{DC}, T_{AC}, 1\}$ tem os componentes

$$\{T_{DC}\} = \begin{vmatrix} \frac{F_2}{2\gamma\ell} \left(\frac{q_j - q_k}{K} \right) + \frac{q_k}{K} - F_1 \cdot \frac{q_j}{K} \\ \frac{F_1 - 1}{\gamma\ell} \left(\frac{q_j - q_k}{K} \right) - F_4 \cdot \frac{q_j}{K} \end{vmatrix}$$

$$\{ T_{AC} \} = \begin{vmatrix} -\frac{2F_3}{\gamma\ell} & \frac{q_k - q_j}{K} & -2F_2 & \frac{q_j}{K} \\ \frac{F_4}{\gamma\ell} & \frac{q_j - q_k}{K} & +2F_3 & \frac{q_j}{K} \end{vmatrix}$$

As cargas concentradas não foram consideradas. É mais interessante, no caso de viga sobre base elástica, considera-las a plicadas diretamente nas juntas porque facilita a obtenção dos diagramas de fletores e cortantes. Porém, devido a modulação da programação, é extremamente fácil inserir nos programas, rotinas para calcular os efeitos de quaisquer outras cargas que não as jã consideradas.

CAPÍTULO V

MATRIZ FRONTEIRA

No capítulo II ficou constatada a necessidade da matriz fronteira para contornar as descontinuidades nas componentes do vetor de estado das juntas.

Sua função é, portanto, estabelecer as condições de compatibilidade de deslocamentos e as condições de equilíbrio das juntas.

É definida, obviamente, em relação ao sistema global de referência da estrutura, de modo que todos os vetores de estado se referem também ao sistema global.

A figura 5.1 mostra um vetor de estado típico de uma barra i de pórtico no espaço na extremidade k :

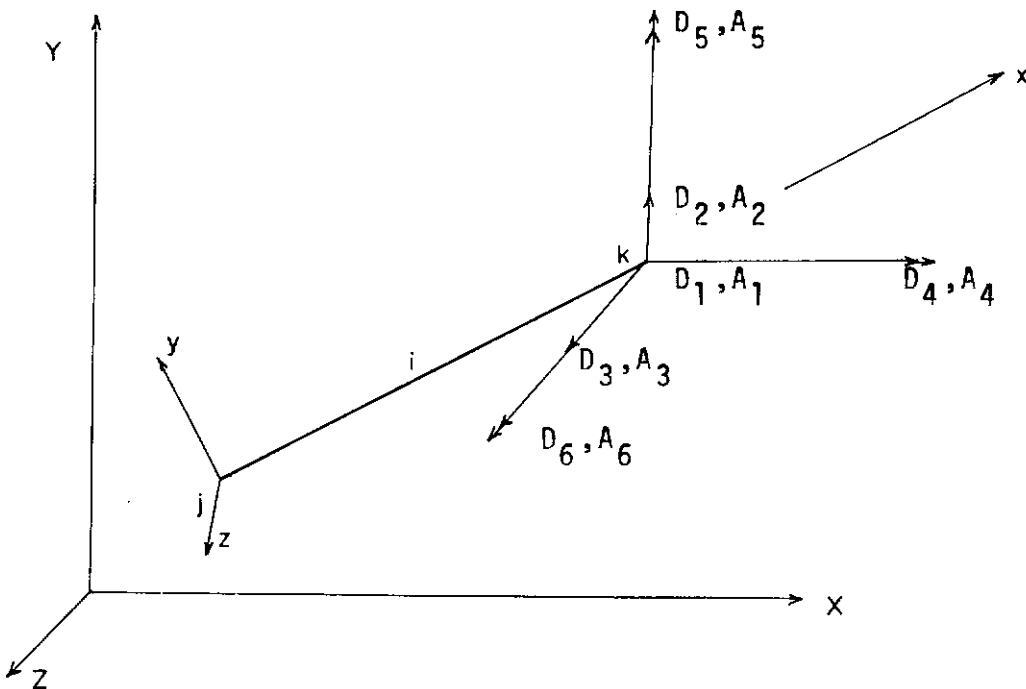


Fig. 5.1

$$\{V\}_{k,i} = \{D_1, D_2, D_3, D_4, D_5, D_6, A_1, A_2, A_3, A_4, A_5, A_6, 1\}_{k,i}$$

ou $\{V\}_{k,i} = \{D, A, 1\}_{k,i}$

onde $\{D\}_{k,i}$ representa os deslocamentos generalizados $D_1, D_2, D_3, D_4, D_5, D_6$ e $\{A\}_{k,i}$ as ações generalizadas A_1, A_2, A_3, A_4, A_5 e A_6 na extremidade k da barra i , no sistema global.

A equação

$$\{V\}_{j,i+1} = [F]_{i+1} \{V\}_{k,i}$$

representa as condições de compatibilidade e de equilíbrio na junta $i + 1$, figura 5.2, e será escrita na forma:

$$\begin{bmatrix} D \\ A \\ 1 \end{bmatrix}_{j,i+1} = \begin{bmatrix} F_{DD} & F_{DA} & F_{DC} \\ F_{AD} & F_{AA} & F_{AC} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} D \\ A \\ 1 \end{bmatrix}_{k,i} \quad (5.1)$$

para facilitar a exposição dos itens aqui estudados. Não é demais frisar que $[F]_{i+1}$ é a matriz fronteira da junta $i + 1$.

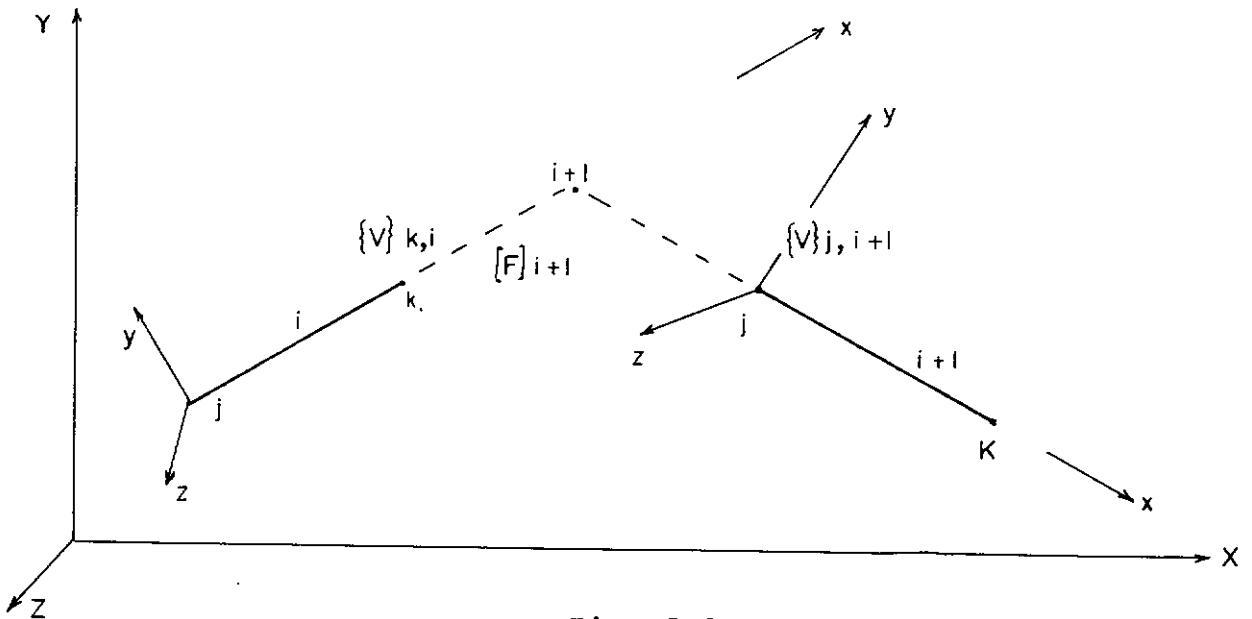


Fig. 5.2

As submatrizes da matriz fronteira $|F|$ serão determinadas a seguir para todos os casos de descontinuidade da junta $i+1$.

5.1 - Ausência de Descontinuidade:

A finalidade deste item é determinar, em definitivo, as submatrizes $|F_{DD}|$ e $|F_{AA}|$ da equação (5.1).

A propósito, de (5.1) obtem-se a equação:

$$\{D\}_{j,i+1} = |F_{DD}| \{D\}_{k,i} + |F_{DA}| \{A\}_{k,i} + \{F_{DC}\}_{i+1} \quad (5.2)$$

Não havendo descontinuidade, a equação matricial de compatibilidade de deslocamentos na junta $i+1$ da figura 5.2 é:

$$\begin{aligned} \{D\}_{j,i+1} - \{D\}_{k,i} &= \{0\} \\ \text{ou} \\ \{D\}_{j,i+1} &= \{D\}_{k,i} \end{aligned} \quad (5.3)$$

Comparando (5.2) e (5.3) tem-se, imediatamente:

$$\begin{aligned} |F_{DD}| &= |I| && \text{(matriz identidade)} \\ |F_{DA}| &= |0| && \text{(matriz nula)} \\ \{F_{DC}\} &= \{0\} && \text{(vetor nulo)} \end{aligned}$$

Também, de (5.1) obtem-se

$$\{A\}_{j,i+1} = |F_{AD}| \{D\}_{k,i} + |F_{AA}| \{A\}_{k,i} + \{F_{AC}\}_{i+1} \quad (5.4)$$

e, a equação de equilíbrio da junta $i+1$ será

$$\{A\}_{j,i+1} + \{A\}_{k,i} = \{0\} \quad \text{ou}$$

$$\{A\}_{j,i+1} = -\{A\}_{k,i} \quad (5.5)$$

Comparando também, (5.4) e (5.5) conclui-se que:

$$\begin{aligned} |F_{AA}| &= -|I| \\ |F_{AD}| &= |0| \\ \{F_{AC}\} &= \{0\} \end{aligned}$$

Em resumo, a equação (5.1) será:

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i+1} = \begin{vmatrix} I & 0 & 0 \\ 0 & -I & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} \quad (5.6)$$

5.2 - Cargas nas Juntas:

O vetor de cargas aplicado na junta $i+1$ da figura 5.3:

$$\{P\}_{i+1} = \{PX, PY, PZ, PXX, PYY, PZZ\}_{i+1}$$

e os vetores de estado nas extremidades das barras $\{A\}_{j,i+1}$ e $\{A\}_{k,i}$ estarão em equilíbrio se

$$\{P\}_{i+1} - \{A\}_{k,i} - \{A\}_{j,i+1} = \{0\} \quad \text{ou seja}$$

$$\text{se} \quad \{A\}_{j,i+1} = -\{A\}_{k,i} + \{P\}_{i+1} \quad (5.7)$$

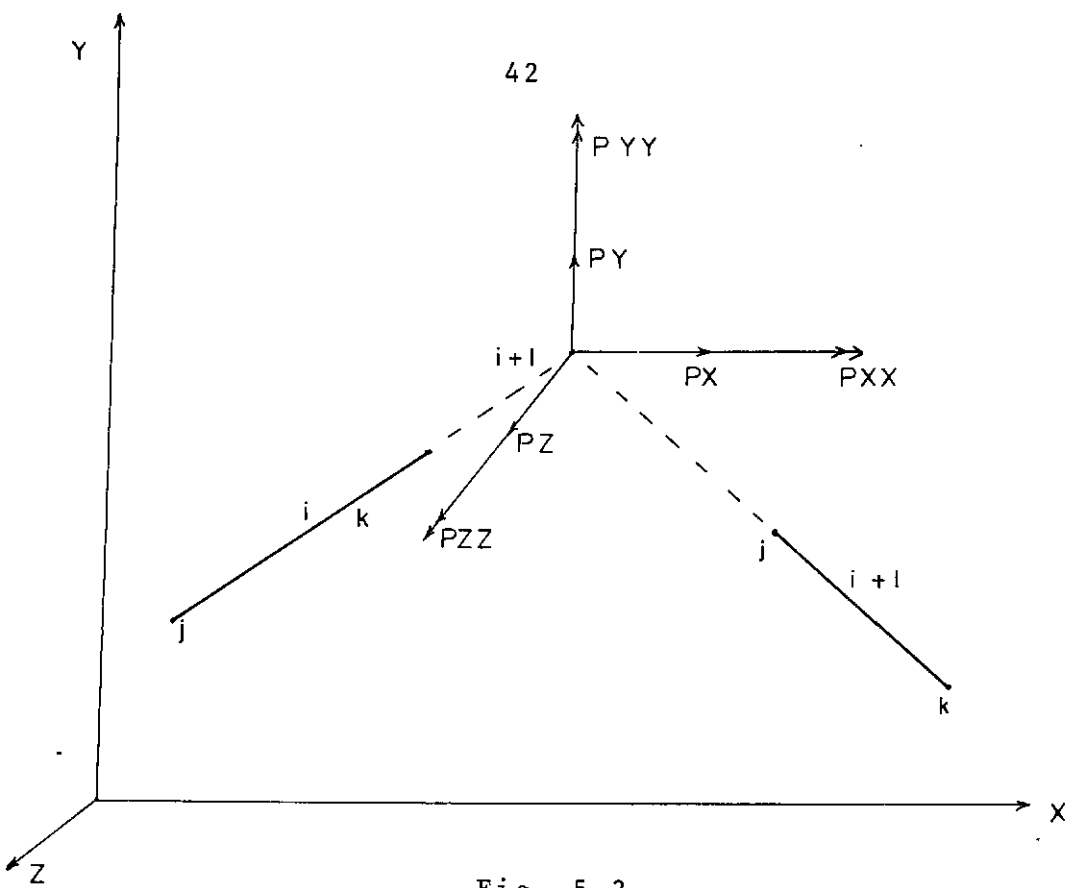


Fig. 5.3

Comparando (5.7) com (5.4), vê-se que:

$\{FAC\} \equiv \{P\}_{i+1}$. portanto

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i+1} = \begin{vmatrix} I & 0 & 0 \\ 0 & -I & P \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} \quad (5.8)$$

5.3 - Deformações Impostas :

Se se impõem à junta $i+1$ da figura 5.4 as deformações generalizadas:

$$\{\Delta\}_{i+1} = \{\Delta X, \Delta Y, \Delta Z, \Delta XX, \Delta YY, \Delta ZZ\}_{i+1} \dots$$

então a equação de compatibilidade será:

$$\{D\}_{j,i+1} - (\{D\}_{k,i} + \{\Delta\}_{i+1}) = \{0\} \quad \text{ou}$$

$$\{D\}_{j,i+1} = \{D\}_{k,i} + \{\Delta\}_{i+1} \quad (5.9)$$

Comparando (5.9) e (5.2) conclui-se que

$$\{F_{DC}\} \equiv \{\Delta\}_{i+1}$$

Logo, a equação (5.8) passa a

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i+1} = \begin{vmatrix} I & 0 & \Delta \\ 0 & -I & P \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} \quad (5.10)$$

Com as deformações impostas, pode-se analisar pórticos planos e no espaço e vigas sobre base elástica para recalques nos apoios extremos.

A justificativa da não inclusão da viga contínua, de flexão livre, será dada no próximo capítulo. Porém, a obtenção das linhas de influência pelo teorema de Land, através de deformações impostas, é válida para as quatro estruturas analisadas neste trabalho.

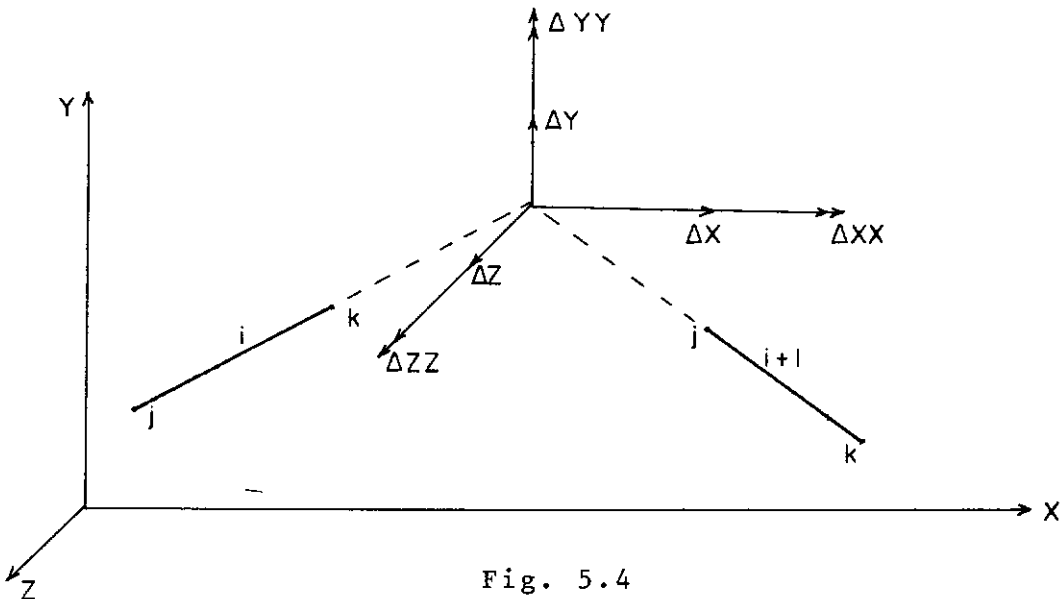


Fig. 5.4

5.4 - Apoios Elásticos:

As reações dos apoios elásticos, indicados na figura 5.5, são proporcionais aos deslocamentos da junta $i+1$.

Denominando de KX , KY , KZ , KXX , KYY , KZZ as constantes de proporcionalidade, também chamadas rigidezas de molas, tem-se:

$$\begin{array}{c|c|c}
 \begin{array}{c} RX \\ RY \\ RZ \\ RXX \\ RYY \\ RZZ \end{array} & = - & \begin{array}{c} KX \\ KY \\ KZ \\ KXX \\ KYY \\ KZZ \end{array}
 \end{array}
 \begin{array}{c} DX \\ DY \\ DZ \\ DXX \\ DYY \\ DZZ \end{array}$$

$$\text{ou } \{R\} = - [K] \{D\} \quad (5.11)$$

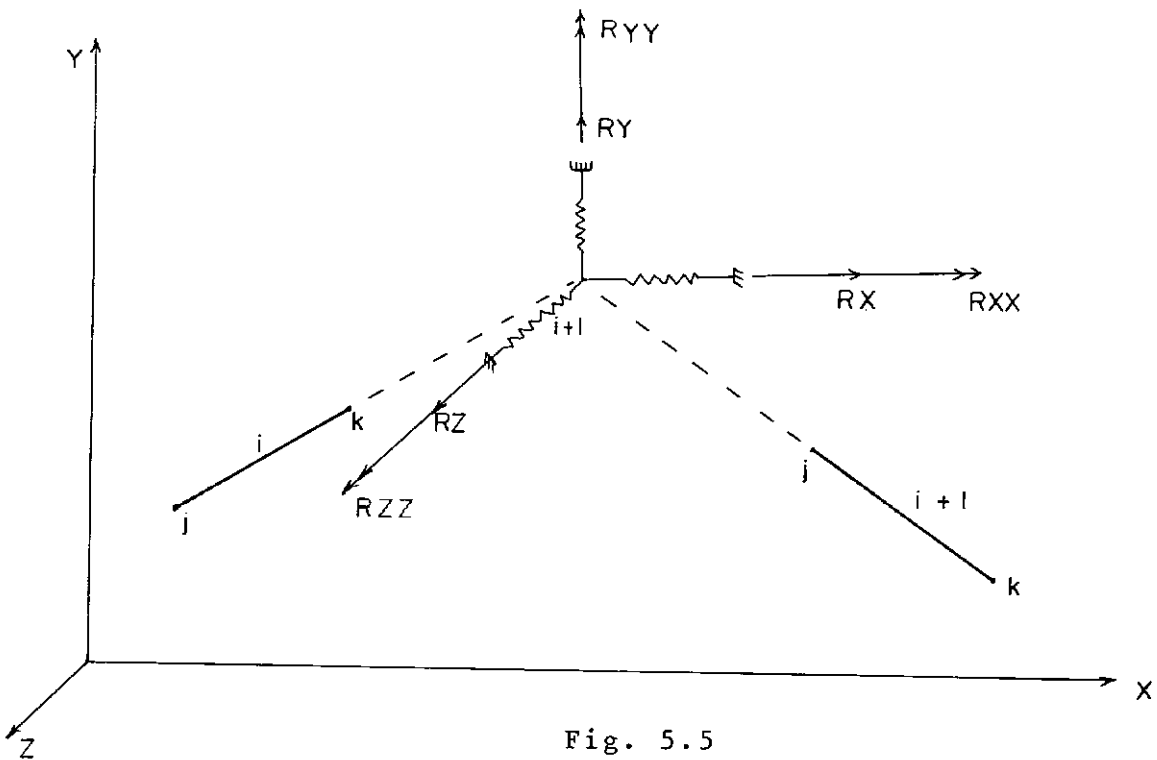


Fig. 5.5

A equação de equilíbrio na junta $i+1$ é, simplesmente:

$$- \{A\}_{k,i} - \{A\}_{j,i+1} + \{R\} = \{0\} \quad \text{ou}$$

$$\{A\}_{j,i+1} = -\{A\}_{k,i} + \{R\} \quad (5.12)$$

Substituindo (5.11) em (5.12) obtém-se:

$$\{A\}_{j,i+1} = -\{A\}_{k,i} - [K] \{D\} \quad (5.13)$$

Confrontando (5.13) com (5.4), vê-se que

$$[F_{AD}] \equiv -[K]$$

e, a equação (5.10) passa a ser da forma:

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i+1} = \begin{vmatrix} I & 0 & \Delta \\ -K & -I & P \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} \quad (5.14)$$

Nota-se que só foi possível contornar a descontinuidade causada pelo vetor de reações dos apoios elásticos, $\{R\}$ na equação (5.12), graças à equação adicional dada por (5.11).

No caso de vigas contínuas com apoios fixos intermediários, não é válida a equação (5.11) não sendo, portanto, possível obter a matriz fronteira da equação (5.14). Uma solução seria a substituição do apoio fixo por uma mola de rigidez infinita.

Outra é a técnica da eliminação das descontinuidades explicada, em detalhes, no próximo capítulo.

5.5 - Ramificações:

Seja a barra r da figura 5.6, chamada ramificação da estrutura constituída pelas barras em série $i-1, i, i+1$.

Admite-se que seja rigidamente conectada à estrutura na junta $i+1$ pela sua extremidade k perfeitamente engastada na extremidade j , isto é:

$$\{D\}_{j,r} = \{0\}$$

Seja $|S|$ a matriz de rigidez da ramificação referida ao sistema global e $\{D\}_{k,r}$ o vetor deslocamento na extremidade k da ramificação. Assim, o efeito da ramificação na junta $i+1$ é dado pela equação fundamental do método dos deslocamentos

$$\{A\}_{k,r} = |S| \{D\}_{k,r} \quad (5.15)$$

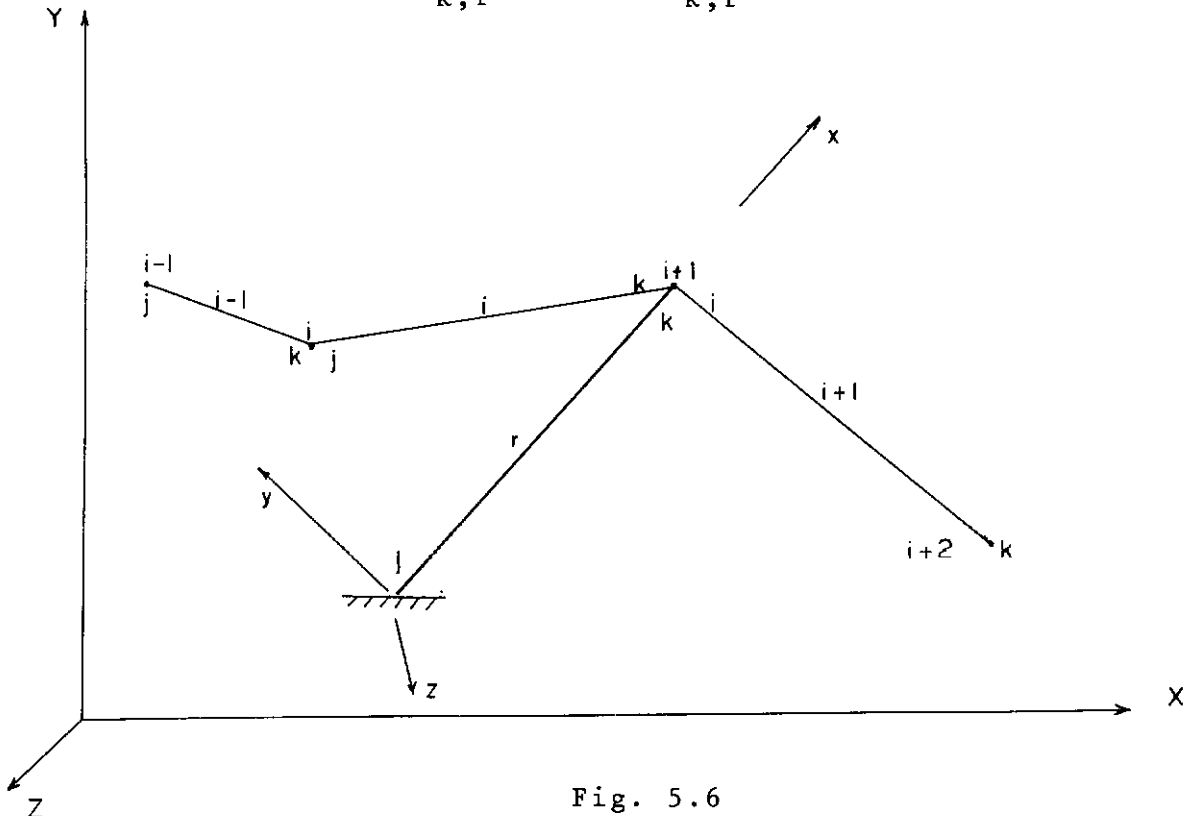


Fig. 5.6

e, a equação de equilíbrio na junta $i+1$ assegura que

$$-\{A\}_{k,i} - \{A\}_{j,i+1} - \{A\}_{k,r} = \{0\} \quad \text{ou}$$

$$\{A\}_{j,i+1} = -\{A\}_{k,i} - \{A\}_{k,r}$$

Substituindo $\{A\}_{k,r}$ pelo seu valor dado em (5.15)

$$\{A\}_{j,i+1} = -\{A\}_{k,i} - |S| \{D\}_{k,r}$$

e, da condição de compatibilidade de deslocamentos na junta $i+1$

$$\{D\}_{k,r} = \{D\}_{k,i} = \{D\}_{j,i+1}$$

obtem-se finalmente:

$$\{A\}_{j,i+1} = -\{A\}_{k,i} - |S| \{D\}_{k,i} \quad (5.16)$$

Comparando (5.16) com (5.4), observa-se que:

$$|F_{AD}| \equiv |S|$$

e, a equação (5.14) será da forma:

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i+1} = \begin{vmatrix} I & 0 & \Delta \\ -K-S & -I & P \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} \quad (5.17)$$

A equação (5.17) mostra que uma ramificação ou mesmo uma subestrutura pode ser substituída por apoios elásticos.

O efeito das cargas que atuam na ramificação é considerado através das ações equivalentes das cargas na junta $i+1$ e é efetuado automaticamente, na programação.

5.6 - Liberações Generalizadas:

A análise de uma estrutura com liberações impostas, pelo método dos deslocamentos, não oferece nenhuma dificuldade.

Infelizmente tal não ocorre com o método das matrizes de transferência.

Enquanto que no método dos deslocamentos a liberação se refere ao sistema local, liberação na barra, no método das matrizes de transferência ela se refere ao sistema global, liberação na junta.

Enquanto que no primeiro método simplesmente se trata de obter a matriz de rigidez da barra com liberação, no segundo é necessário fazer um rearranjo total na matriz de transferência toda vez que se depara com liberações de modo a introduzir no vetor de estado os deslocamentos relativos associados às liberações com a consequente eliminação de igual número de componentes do vetor de estado, escolhidas arbitrariamente ⁽¹⁾. A técnica da redução da matriz, empregada na análise de viga contínua, para contornar as descontinuidades nos apoios fixos pode ser utilizada para as liberações mas requer o conhecimento *à priori*, do número de liberações impostas na junta o que não acontece com a técnica da eliminação.

A necessidade de manter a programação com um mínimo de sofisticação para que os programas possam ser processados em computadores de pequeno porte, conduziu a adotar uma técnica aproximada, com bons resultados ⁽³⁾.

Consiste essencialmente, em admitir a junta como sendo uma

barra de comprimento e inércias infinitamente pequenas.

Seja, por exemplo, uma rótula na viga representada na figura 5.7, onde a barra fictícia f substitui a rótula.

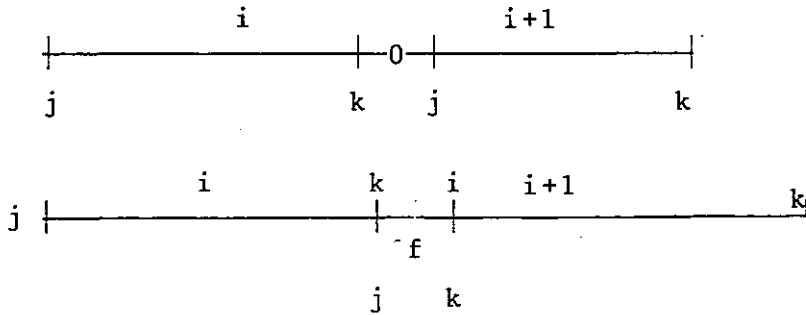


Fig. 5.7

A matriz de transferência da barra f é, equação (3.7):

$$\begin{vmatrix} y \\ \theta \\ Q \\ M \end{vmatrix}_{k,f} = \begin{vmatrix} 1 & \ell & \ell^3/6EI_z & -\ell^2/2EI_z \\ 0 & 1 & \ell^2/2EI_z & -\ell/EI_z \\ 0 & 0 & -1 & 0 \\ 0 & 0 & \ell & -1 \end{vmatrix} \begin{vmatrix} y \\ \theta \\ Q \\ M \end{vmatrix}_{j,f} \quad (5.18)$$

De (5.18)

$$M_k = \ell Q_j - M_j$$

mas $M_k = M_j = 0$ devida à rotula, portanto $\ell = 0$, porque $Q_j \neq 0$.

Para um computador que trabalha com nove algarismos significativos, pode-se tomar:

$$\ell = 0 \approx 10^{-10} \quad \text{e} \quad \ell^2 = 10^{-20} = 0$$

de modo que $\theta_k - \theta_j = -\frac{\ell}{EI_z} M_j$, de (5.18), fornece

$$M_j = - \frac{\theta_k}{\frac{\ell}{EI_z}} \text{ e, como } M_j=0 \text{ e } \theta_k - \theta_j \neq 0 \text{ chega-se à conclusão de que}$$

$$\frac{\ell}{EI_z} = \infty \cong 10^{10} = \gamma$$

$$\text{Portanto, } EI_z = \frac{\ell}{\gamma} = \frac{10^{10}}{10^{10}} = 10^0 \cong 0$$

Dessa forma a equação (5.18) passa a ser:

$$\begin{vmatrix} y \\ \theta \\ Q \\ M_{k,f} \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & \gamma \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{vmatrix} \begin{vmatrix} y \\ \theta \\ Q \\ M_{j,f} \end{vmatrix} \quad (5.19)$$

Encarando agora a matriz de (5.19) como uma matriz fronteira, na forma da equação (5.17), tem-se

$$\begin{vmatrix} D \\ A \\ 1_{j,i+1} \end{vmatrix} = \begin{vmatrix} F_{DD} & F_{DA} & F_{DC} \\ F_{AD} & F_{AA} & F_{AC} \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1_{k,i} \end{vmatrix}$$

onde:

$$|F_{DA}| = \begin{vmatrix} 0 & 0 \\ 0 & \gamma \end{vmatrix}$$

Generalizando o raciocínio para considerar as liberações de pórtico no espaço, a submatriz $|F_{DA}|$ será:

$$|F_{DA}| = \begin{vmatrix} \gamma_x & & & \\ & \gamma_y & & 0 \\ & & \gamma_{xx} & \\ & 0 & & \gamma_{yy} \\ & & & & \gamma_{zz} \end{vmatrix} = |\gamma| = |\infty|$$

e a equação (5.17) tem a forma definitiva:

$$\begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{j,i+1} = \begin{vmatrix} I & \gamma & \Delta \\ -K-S & -I & P \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} D \\ A \\ 1 \end{vmatrix}_{k,i} \quad (5.20)$$

CAPÍTULO VI

ANÁLISE DAS ESTRUTURAS

6.1 - Análise Geral:

De posse das matrizes de transferência das barras e das matrizes fronteira das juntas, pode-se analisar qualquer estrutura apoiada pelo método das matrizes de transferência.

Seja analisar, por exemplo, a viga representada na figura 6.1

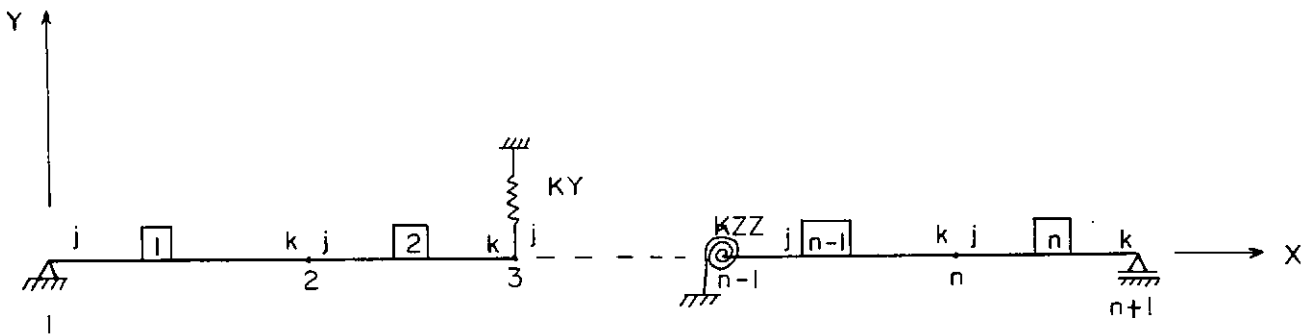


Fig. 6.1

As etapas a seguir são, essencialmente, as seguintes:

- 1 - Percorrer a estrutura da maneira exposta no capítulo II de modo a relacionar os vetores de estado dos apoios extremos.

$$\{V\}_{n+1} = [T]_n [F]_n [T]_{n-1} [F]_{n-1} \dots [T]_2 [F]_2 [T]_1 \{V\}_1 \quad \text{ou}$$

$$\{V\}_{n+1} = [T] \{V\}_1 \quad \text{isto é:}$$

$$\begin{vmatrix} y \\ \theta \\ Q \\ M \\ 1 \end{vmatrix}_{n+1} = \begin{vmatrix} t_{11} & t_{12} & t_{13} & t_{14} & t_{15} \\ t_{21} & t_{22} & t_{23} & t_{24} & t_{25} \\ t_{31} & t_{32} & t_{33} & t_{34} & t_{35} \\ t_{41} & t_{42} & t_{43} & t_{44} & t_{45} \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} y \\ \theta \\ Q \\ M \\ 1 \end{vmatrix}_1 \quad (6.1)$$

2 - Resolver o sistema (6.1) para as condições de contorno dos apoios extremos.

Em cada apoio existem duas componentes nulas ou conhecidas e duas a determinar devendo-se, portanto, resolver duas equações a duas incógnitas:

$$\begin{vmatrix} t_{12} & t_{13} \\ t_{42} & t_{43} \end{vmatrix} \begin{vmatrix} \theta \\ Q \end{vmatrix}_1 + \begin{vmatrix} t_{15} \\ t_{45} \end{vmatrix} = \begin{vmatrix} y \\ M \end{vmatrix}_{n+1} = \begin{vmatrix} 0 \\ 0 \end{vmatrix} \quad \text{ou}$$

$$\begin{vmatrix} \theta \\ Q \end{vmatrix}_1 = - \begin{vmatrix} t_{12} & t_{13} \\ t_{42} & t_{43} \end{vmatrix}^{-1} \begin{vmatrix} t_{15} \\ t_{45} \end{vmatrix} \quad (6.2)$$

Obtido o vetor de estado no apoio 1, determina-se o do apoio n+1 através de (6.1) e (6.2):

$$\begin{vmatrix} \theta \\ Q \end{vmatrix}_{n+1} = \begin{vmatrix} t_{22} & t_{23} \\ t_{32} & t_{33} \end{vmatrix} \begin{vmatrix} \theta \\ Q \end{vmatrix}_1 + \begin{vmatrix} t_{25} \\ t_{35} \end{vmatrix} \quad \text{ou}$$

$$\begin{Bmatrix} \theta \\ Q \end{Bmatrix}_{n+1} = - \begin{bmatrix} t_{22} & t_{23} \\ t_{32} & t_{33} \end{bmatrix} \begin{bmatrix} t_{12} & t_{13} \\ t_{42} & t_{43} \end{bmatrix}^{-1} \begin{Bmatrix} t_{15} \\ t_{45} \end{Bmatrix} + \begin{Bmatrix} t_{25} \\ t_{35} \end{Bmatrix}$$

3 - Percorrer novamente a estrutura de modo a determinar passo a passo os vetores das seções intermediárias:

$$\begin{aligned} \{V\}_{k,1} &= [T]_1 \{V\}_1 \\ \{V\}_{j,2} &= [F]_2 \{V\}_{k,1} \\ &\vdots \\ \{V\}_{j,n} &= [F]_n \{V\}_{k,n-1} \end{aligned}$$

Conclui-se que a ordem da matriz de transferência da estrutura não depende do número de juntas que a compõem e nem do grau de indeterminação estática ou cinemática. A ordem da matriz a ser invertida é igual à metade do número de componentes do vetor de estado.

A equação 6.1 mostra que tudo se passa como se a estrutura pudesse ser reduzida a uma única barra, de extremos perfeitamente definidos. Por isso o método é também chamado de método da redução.

Embora o método possa analisar qualquer estrutura aportica da passível dessa redução, os programas apresentados estão aptos a resolver somente aquelas constituídas de barras em série.

Já foi dito que nesses casos o método é bastante eficiente

possibilitando a análise de estruturas com grande número de juntas em computadores de pequeno porte.

A seguir faz-se a análise detalhada de cada uma das quatro estruturas consideradas neste trabalho.

6.2 - Viga Contínua:

A análise de vigas contínuas com apoios fixos intermediários não pode ser feita somente com o exposto no item anterior devido à descontinuidade da força cortante nos apoios, causada pela reação segundo o eixo Y.

A equação adicional (5.11) permitiu a inclusão das reações dos apoios elásticos na matriz fronteira, contornando assim as descontinuidades nas componentes do vetor de estado.

Nos apoios fixos não mais subsiste a equação (5.11), mas poder-se-ia considerar as reações como se fossem cargas concentradas e proceder à análise exposta no item 6.1.

Tal procedimento só se aplica numa análise manual ⁽⁴⁾ sendo necessário, na análise automática, eliminar essas reações.

A técnica, utilizada neste trabalho ⁽³⁾, dentre as muitas outras possíveis ⁽¹⁾, consiste na redução da ordem da matriz de transferência.

As componentes do vetor de estado

$$\{V\} = \{y, \theta, Q, M\}$$

podem-se reduzir as duas dando origem ao vetor de estado reduzido:

$$\{v\} = \{\theta, M\}$$

cujas componentes são contínuas ao longo de toda a viga.

De fato, nos apoios fixos a translação y é nula e a força cortante pode ser eliminada como segue:

Seja a viga representada na figura 6.2 onde cada vão pode ser formado por diversas barras como as da figura 6.1.

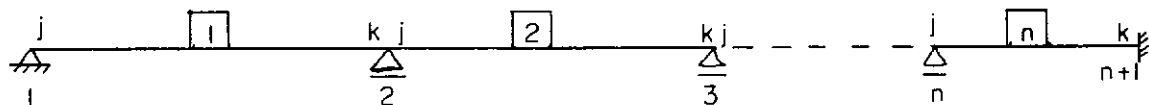


Fig. 6.2

A equação (6.1) para um vão qualquer, com $y_k = y_j = 0$ fornece:

$$0 = t_{12} \theta_j + t_{13} Q_j + t_{14} M_j + t_{15} \quad (6.3)$$

$$\theta_k = t_{22} \theta_j + t_{23} Q_j + t_{24} M_j + t_{25} \quad (6.4)$$

$$Q_k = t_{32} \theta_j + t_{33} Q_j + t_{34} M_j + t_{35} \quad (6.5)$$

$$M_k = t_{42} \theta_j + t_{43} Q_j + t_{44} M_j + t_{45} \quad (6.6)$$

De (6.3) obtém-se

$$Q_j = -(t_{12}\theta_j + t_{14}M_j + t_{15})/t_{13} \quad (6.7)$$

que pode ser eliminado por simples substituição em (6.4) e (6.6):

$$\theta_k = (t_{22} - \frac{t_{12}}{t_{13}} t_{23})\theta_j + (t_{24} - \frac{t_{14}}{t_{13}} t_{23})M_j + t_{25} - \frac{t_{15}}{t_{13}} t_{23} \quad (6.8)$$

$$M_k = (t_{42} - \frac{t_{12}}{t_{13}} t_{43})\theta_j + (t_{44} - \frac{t_{14}}{t_{13}} t_{43})M_j + t_{45} - \frac{t_{15}}{t_{13}} t_{43}$$

Pondo o sistema (6.8) na forma:

$$\begin{vmatrix} \theta \\ M \\ 1 \end{vmatrix}_k = \begin{vmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \theta \\ M \\ 1 \end{vmatrix}_j$$

obtem-se a matriz de transferência reduzida que será representada simbolicamente por:

$$\{v\}_k = |t| \{v\}_j \quad (6.9)$$

A matriz fronteira (5.20) reduz-se também a

$$\begin{vmatrix} \theta \\ M \\ 1 \end{vmatrix}_j = \begin{vmatrix} 1 & 0 & \Delta ZZ \\ -KZZ & -1 & PZZ \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \theta \\ M \\ 1 \end{vmatrix}_k \quad \text{ou}$$

$$\{v\}_j = |f| \{v\}_k \quad (6.10)$$

As equações (6.4) e (6.10) permitem agora obter o vetor de estado reduzido $\{v\}_{n+1}$ em função de $\{v\}_1$ da viga da figura 6.2:

$$\{v\}_{n+1} = |t|_n |f|_n \dots \dots \dots |t|_2 |f|_2 |t|_1 \{v\}_1 \quad \text{ou}$$

$$\{v\}_{n+1} = [t] \{v\}_1 \quad (6.11)$$

As condições de contorno aplicadas à equação (6.11) fornecem componentes incógnitas nos apoios extremos. A solução de (6.11) consiste em resolver uma equação a uma incógnita, porque as forças cortantes são obtidas das equações (6.7) e (6.5), nessa ordem, uma vez obtidas as componentes do vetor de estado $\{v\}_1$. Dessa forma obtém-se o vetor de estado completo no apoio 1:

$$\{V\}_1 = \{0, \theta, Q, M\}_1$$

e a análise prossegue normalmente de acordo com a etapa 3 do item 6.1 para as juntas livres até que se encontre um apoio fixo quando então, se utiliza de (6.7) para se obter a força cortante.

As reações das juntas podem ser calculadas facilmente através da equação de equilíbrio segundo Y, figura 6.3:

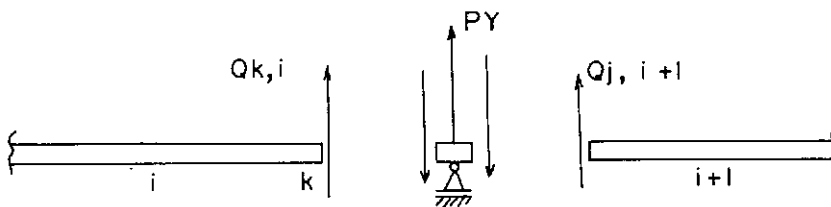


Fig. 6.3

$$R - Q_{k,i} - Q_{j,i+1} + PY = 0 \quad \text{ou}$$

$$R = Q_{k,i} + Q_{j,i+1} - PY$$

A matriz fronteira reduzida (6.10) permite concluir que a única deformação imposta possível nos apoios fixos é a rotação ΔZ

podendo-se, portanto, obter as linhas de influência de momentos fletores pelo teorema de Land.

6.3 - Viga sobre Base Elástica:

A análise de viga sobre base elástica enquadra-se perfeitamente no disposto no item 6.1 porque não se admite apoios fixos intermediários, figura 6.4:

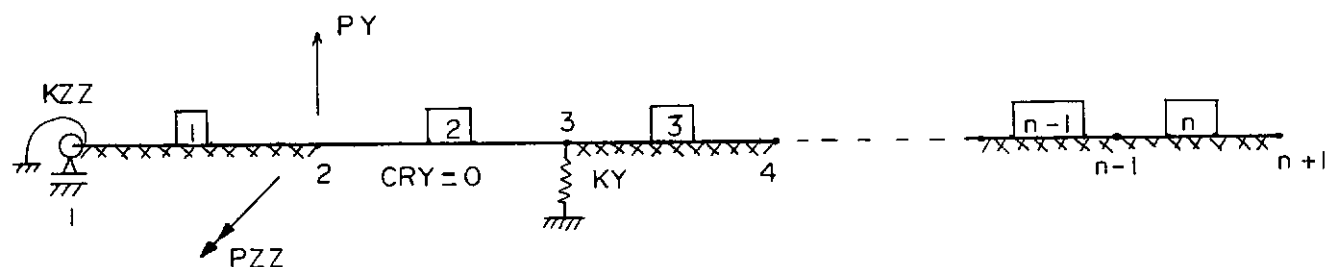


Fig. 6.4

Assim a matriz fronteira (5.20) é válida para qualquer junta com exceção da submatriz $|F_{AD}|$ que é simplesmente

$$|F_{AD}| = - |K|$$

porque não existe ramificação.

Qualquer barra pode estar não assente em meio elástico, como a barra 2, bastando especificar um coeficiente de recalque da base (CRY) igual a 0 (zero).

6.4 - Pórtico Plano:

A análise de pórticos planos se faz da maneira indicada no item 6.1 porque não se admite, neste trabalho, apoios fixos, intermediários. É necessário, no entanto, obter a matriz de transferência no sistema global o que pode ser feito com a matriz de rotação de eixos a duas dimensões.

Seja a figura 6.5 onde se representa uma barra i cujo eixo local x faz um ângulo α com o global X .

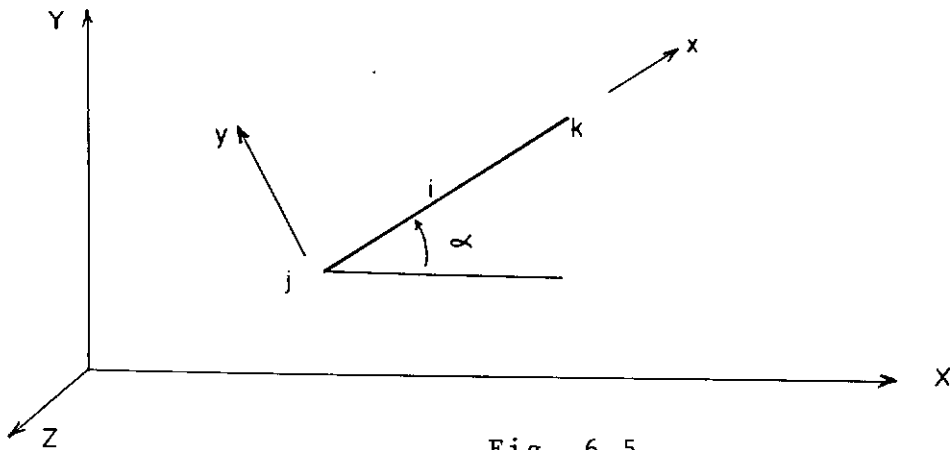


Fig. 6.5

O comprimento da barra é determinado em função das coordenadas das juntas j e k :

$$\ell_i = \sqrt{(X_k - X_j)^2 + (Y_k - Y_j)^2}$$

Os cossenos diretores resultam de:

$$\cos \alpha = (X_k - X_j)/\ell_i \quad \text{e} \quad \cos(90 - \alpha) = \sin \alpha = (Y_k - Y_j)/\ell_i$$

os quais formam a matriz de rotação:

$$|R|_i = \begin{bmatrix} C_x & C_y & 0 \\ -C_y & C_x & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

onde $C_x = \cos \alpha$ e $C_y = \cos (90 - \alpha)$.

A matriz R permite obter o vetor de estado no sistema local em função do mesmo no sistema global pela equação:

$$\{V\}^L = |R| \{V\}^G \quad (6.12)$$

Seja $|T|_i^L$ a matriz de transferência da barra i no sistema local. Então:

$$\{V\}_{k,i}^L = |T|_i^L \{V\}_{j,i}^L \quad \text{ou}$$

$$\begin{bmatrix} D \\ A \\ 1 \end{bmatrix}_{k,i}^L = \begin{bmatrix} T_{DD} & T_{DA} & T_{DC} \\ T_{AD} & T_{AA} & T_{AC} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} D \\ A \\ 1 \end{bmatrix}_{j,i}^L \quad (6.13)$$

Colocando a equação (6.13) na forma:

$$\begin{bmatrix} D \\ A \end{bmatrix}_{k,i}^L = \begin{bmatrix} T_{DD} & T_{DA} \\ T_{AD} & T_{AA} \end{bmatrix}_i^L \begin{bmatrix} D \\ A \end{bmatrix}_{j,i}^G + \begin{bmatrix} T_{DC} \\ T_{AC} \end{bmatrix}_i^L \quad (6.14)$$

e aplicando a relação (6.12), tem-se:

$$\begin{bmatrix} |R| \{D\} \\ |R| \{A\} \end{bmatrix}_{k,i} = \begin{bmatrix} T_{DD} & T_{DA} \\ T_{AD} & T_{AA} \end{bmatrix}_i^L \begin{bmatrix} |R| \{D\} \\ |R| \{A\} \end{bmatrix}_{j,i}^G + \begin{bmatrix} |R| \{T_{DC}\} \\ |R| \{T_{AC}\} \end{bmatrix}_i^G \quad \text{ou}$$

$$\begin{bmatrix} |R| \{0\} \\ 0 |R| \{A\} \end{bmatrix}_i \begin{bmatrix} D \\ A \end{bmatrix}_{k,i}^G = \begin{bmatrix} T_{DD} & T_{DA} \\ T_{AC} & T_{AA} \end{bmatrix}_i^L \begin{bmatrix} |R| \{0\} \\ 0 |R| \{A\} \end{bmatrix}_i \begin{bmatrix} D \\ A \end{bmatrix}_{j,i}^G + \begin{bmatrix} |R| \{0\} \\ 0 |R| \{A\} \end{bmatrix}_i \begin{bmatrix} T_{DC} \\ T_{AC} \end{bmatrix}_i^G$$

Pondo

$$\begin{bmatrix} |R| & |Q| \\ |Q| & |R| \end{bmatrix} = |R_T|$$

$$|R_T|_i \{V\}_{k,i}^G = |T|_i^L |R_T|_i \{V\}_{j,i}^G + |R_T|_i \{T_C\}_i^G \quad \text{ou}$$

$$|R_T|_i^{-1} |R_T|_i \{V\}_{k,i}^G = |R_T|_i^{-1} |T|_i^L |R_T|_i \{V\}_{j,i}^G + |R_T|_i^{-1} |R_T|_i \{T_C\}_i^G$$

Mas $|R_T|$ é ortogonal, logo

$$|R_T|^{-1} = |R_T|^T$$

portanto:

$$\{V\}_{k,i}^G = |T|_i^G \{V\}_{j,i}^G + \{T_C\}_i^G \quad (6.15)$$

onde:

$$|T|_i^G = |R_T|_i^T |T|_i^L |R_T|_i \quad (6.16)$$

$$|T_C|_i^G = |R_T|_i^T |T_C|_i^L$$

Incluindo o vetor de cargas $\{T_C\}_i^G$ na matriz $|T|_i^G$, como em (6.13) vem

$$\{V\}_{k,i}^G = |T|_i^G \{V\}_{j,i}^G \quad (6.17)$$

Se a barra i da figura 6.5 for uma ramificação, sua matriz de rigidez para os graus de liberdade da extremidade K , no sistema local é (2)

$$[S]_{k,r}^L = \begin{bmatrix} EAx/\ell & 0 & 0 \\ 0 & 12EIz/\ell^3 & -6EIz/\ell^2 \\ 0 & -6EIz/\ell^2 & 4EIz/\ell \end{bmatrix} \begin{matrix} L \\ \\ k,r \end{matrix}$$

A matriz de rigidez no sistema global pode ser obtida de (6.16):

$$[S]_{k,r}^G = [R]_i^T [S]_{k,r}^L [R]_i \quad (6.18)$$

6.5 - Pórtico no Espaço:

A análise de pórtico no espaço é inteiramente análoga à de pórticos planos.

A matriz de rotação em três dimensões não é, porém, tão simples como a anterior porque uma barra pode assumir uma posição qualquer no espaço.

Todos os casos possíveis estão detalhadamente estudados em GERE ⁽²⁾ e não há, portanto, necessidade de abordá-los novamente.

CAPÍTULO VII

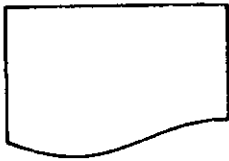
DIAGRAMA DE BLOCOS

Os fluxogramas dos quatro programas são semelhantes de modo que serão apresentados apenas os fluxogramas de p^orticos no espaço PEMT.

As convenções adotadas são as seguintes:



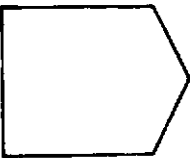
Leitura de dados de cartões



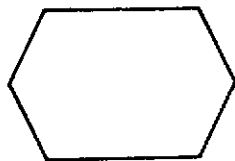
Impressão de relatórios: cabeçalhos
dados lidos e resultados



Processamento de programas e sub-ro
tinas e informações gerais



Decisão, testes



Controle interativo, loop.



Memória auxiliar, disco magnético



Permutações



Conector

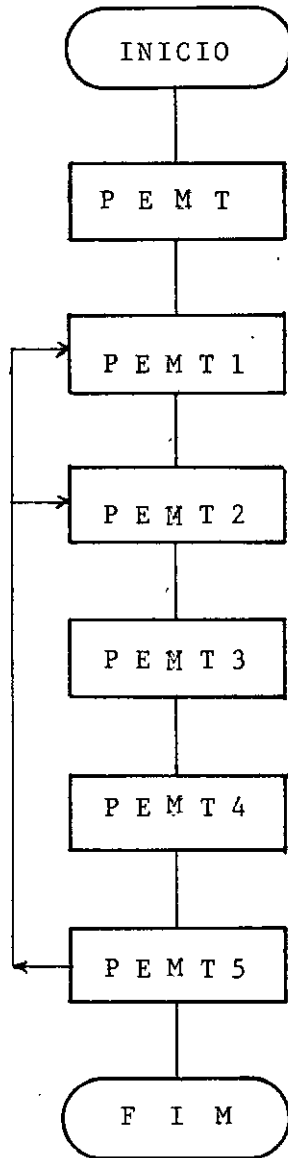


Início ou término do processamento

7.1 - Fluxograma Geral:

Todos os programas estão modulados no sentido de que cada módulo executa uma tarefa específica, e dimensionado para ser processado no sistema IBM-1130 com 8K palavras de memória interna.

Esses módulos ou blocos estão interrelacionados pela declaração CALL LINK do FORTRAN e são executados na sequência indicada.



PEMT - Solicita do operador do S.O.-1130 a seleção das unidades de entrada-saída através das chaves 1 e 2 do painel.

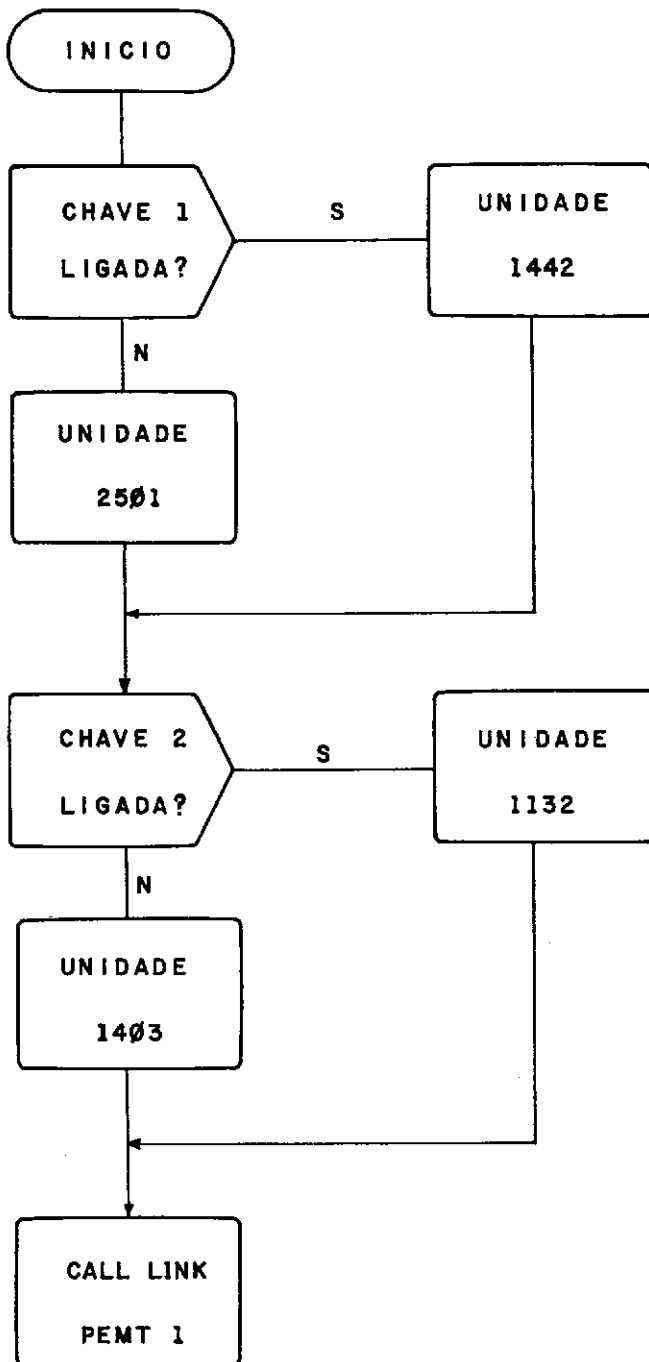
PEMT1 - Procede à leitura dos dados de identificação da estrutura bem como dos dados de sua topologia.

PEMT2 - Procede à leitura do título do carregamento e dos dados de carga.

- PEMT3 - Realiza a primeira passagem ao longo da estrutura e estabelece as condições de contorno.
- PEMT4 - Realiza a segunda passagem, imprimindo os resultados.
- PEMT5 - Faz as ligações com PEMT1 para análise de nova estrutura ou com PEMT2 para novo carregamento ou finaliza o processamento, conforme solicitado.

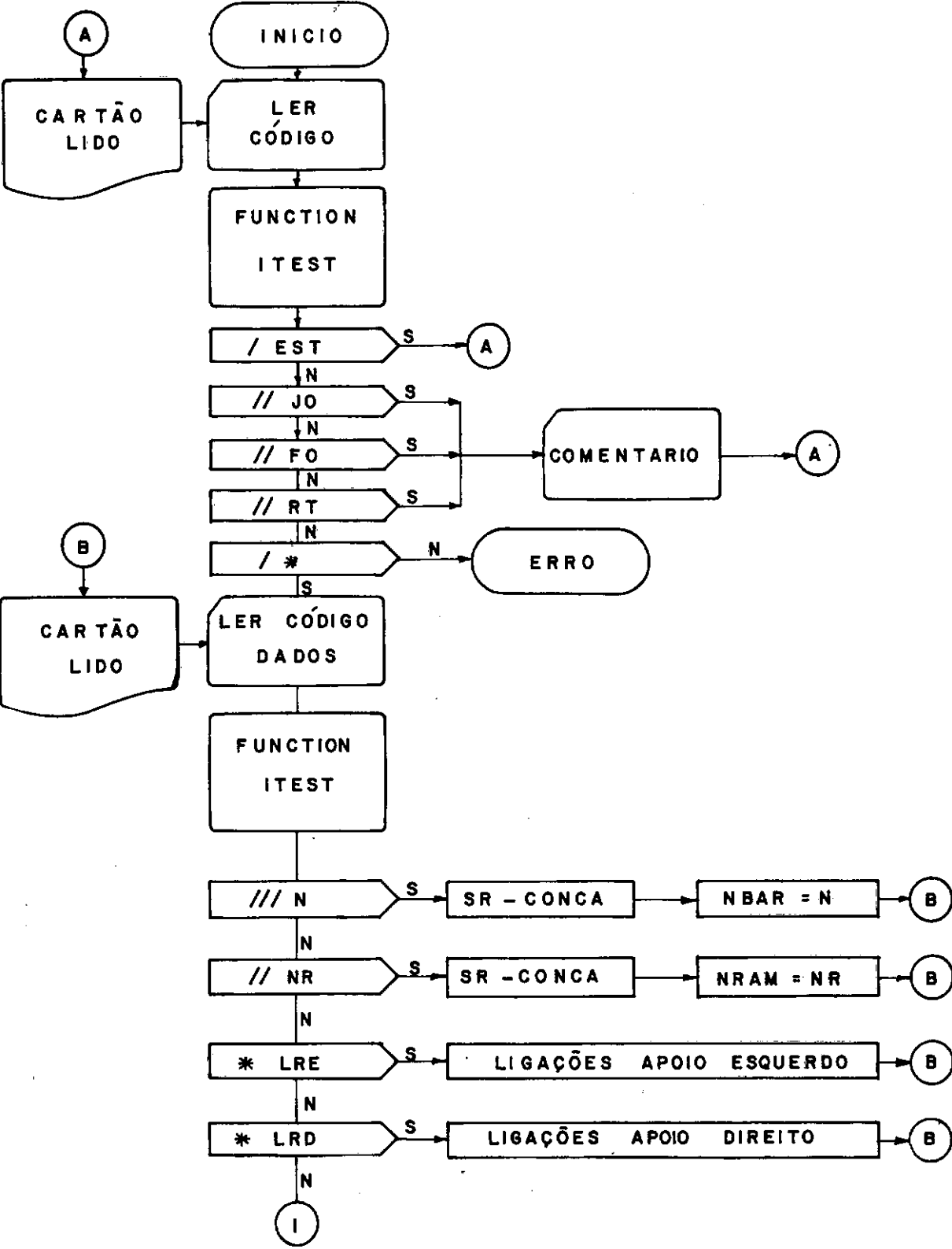
A seguir são dados os fluxogramas de cada módulo.

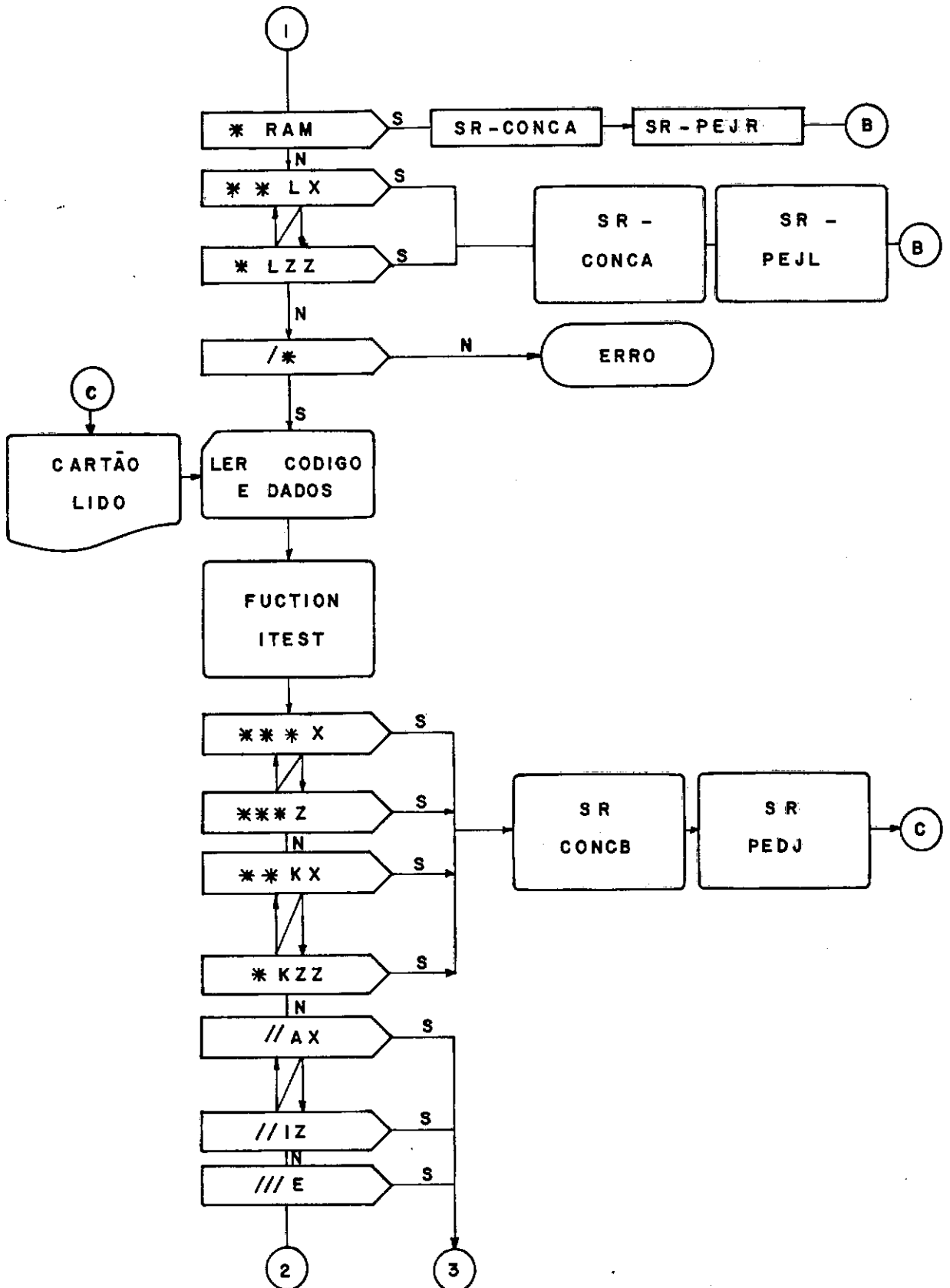
7.1.1 - PEMT

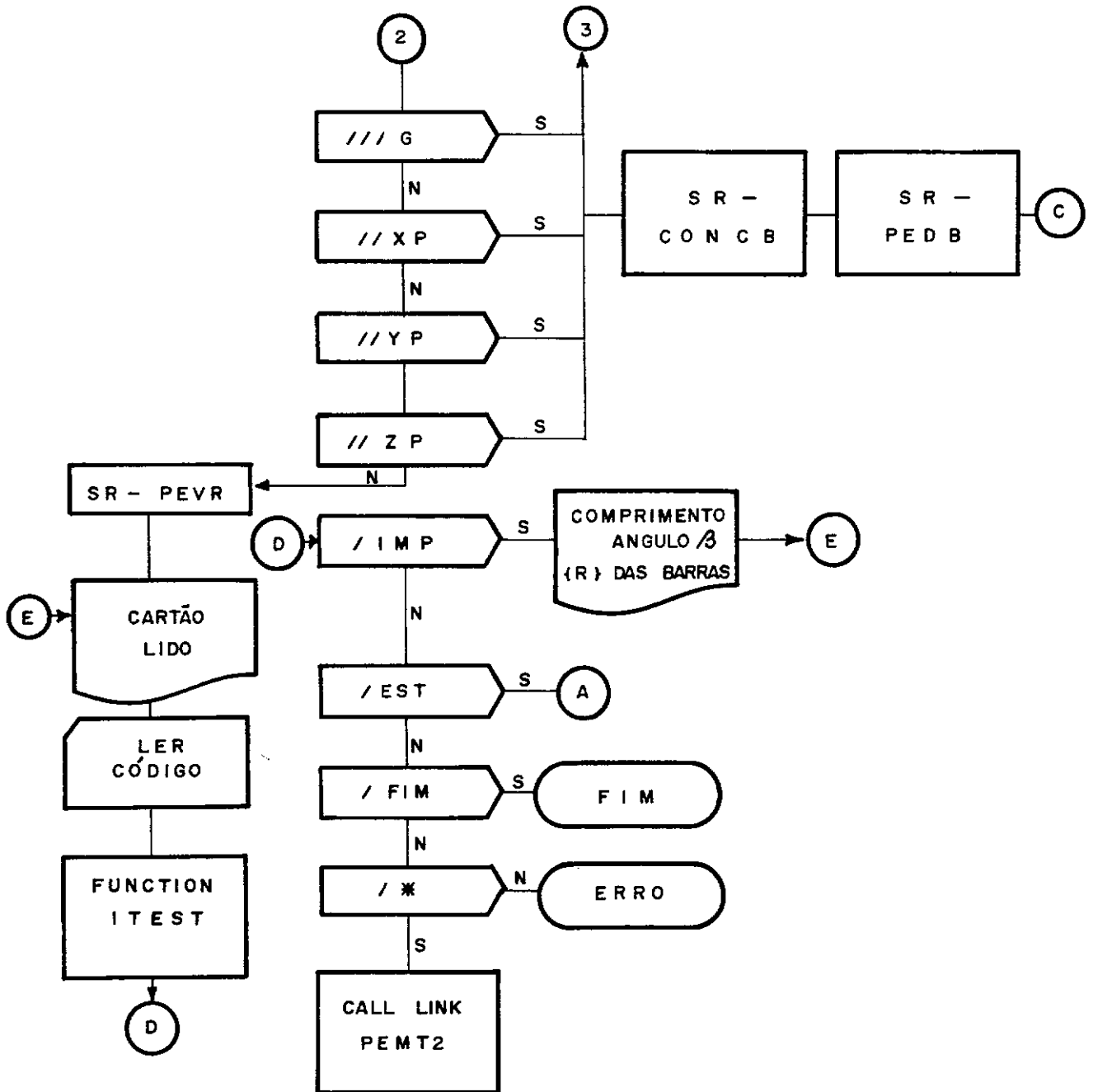


Sub-rotina utilizada: DATSW (FORTRAN)

7.1.2 - PENT 1







Descrição das sub-rotinas:

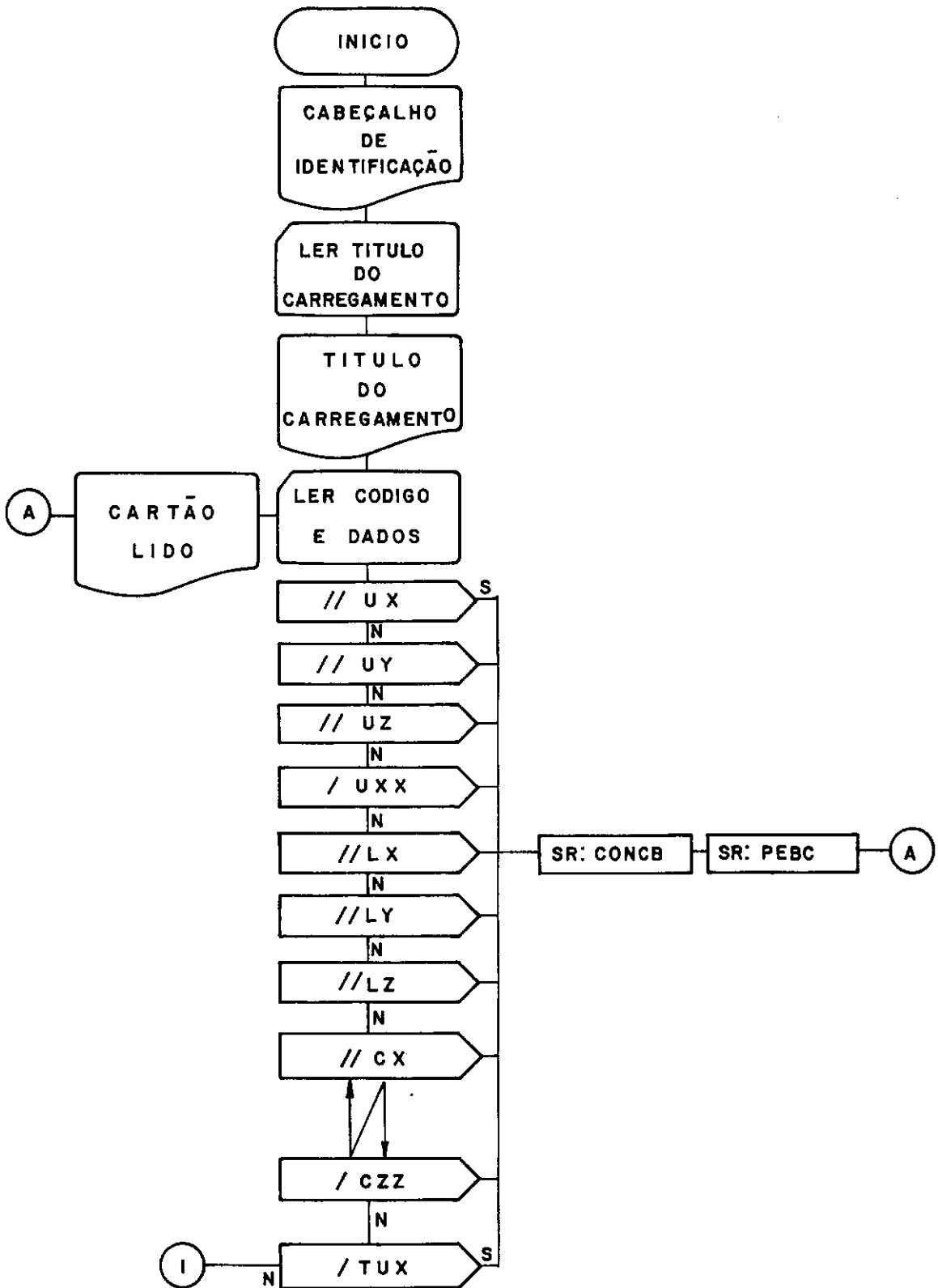
- ITEST é uma função sub-programa que interpreta os códigos escritos nas 4 primeiras colunas do cartão.

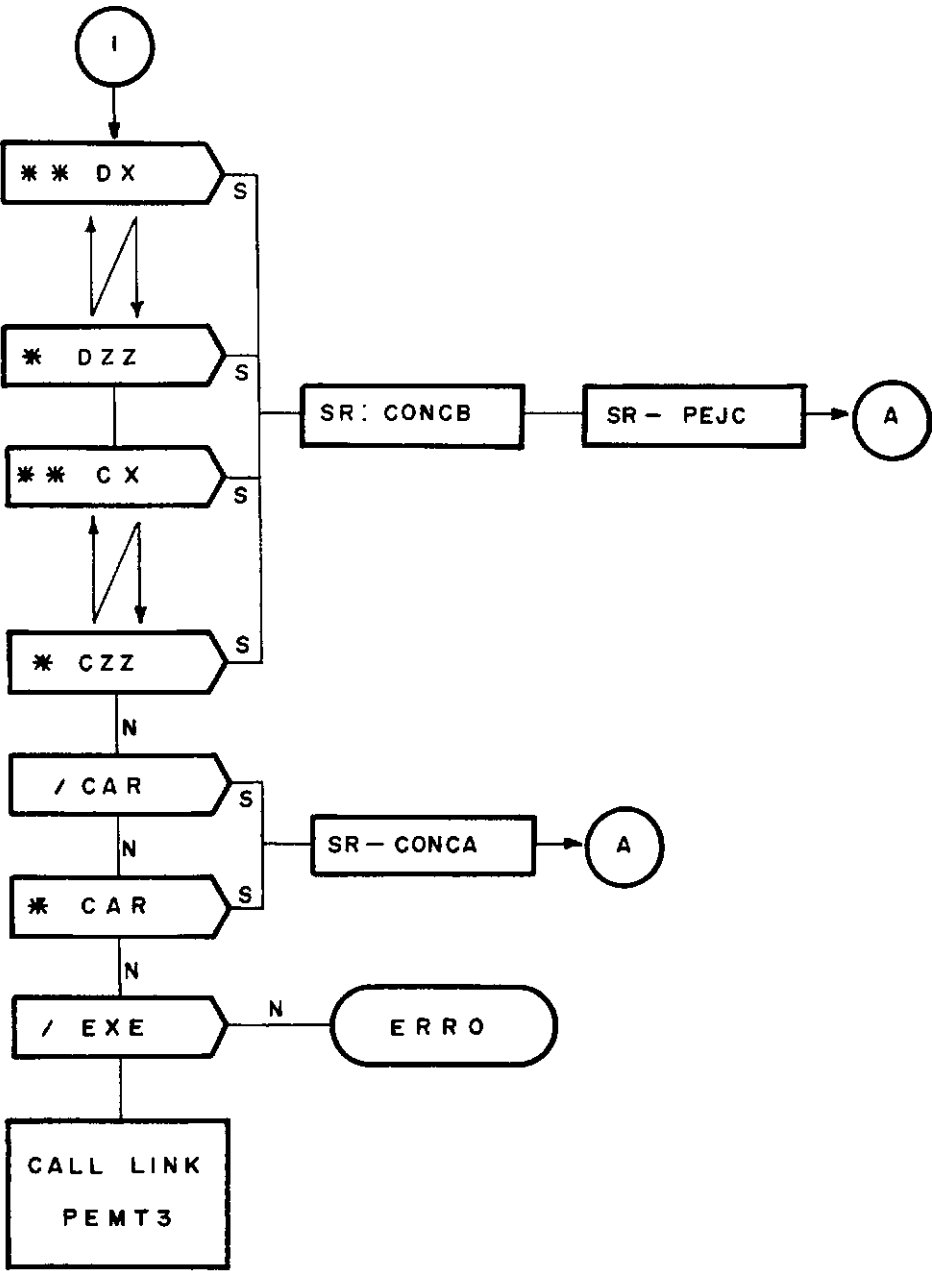
Se o código não for identificado o programa emite uma mensagem de erro no próprio formulário.
- CONCA é uma sub-rotina que faz a consistência dos dados a partir da coluna 6 (seis) do cartão.

Ver os códigos de erros no final do capítulo.
- CONCB é também uma sub-rotina de consistência que atua nas colunas 6 (seis), 11 (onze) e, as vezes na coluna 16 (dezesesseis).

Ver códigos de erros no final do capítulo.
- PEJR esta sub-rotina arquiva as juntas que tenham ramificação.
- PEJL arquiva as juntas com liberações generalizadas.
- PEDJ arquiva os dados referentes às juntas.
- PEDB arquiva os dados referentes às barras, principais e/ou ramificações.
- PEVR esta sub-rotina forma o vetor-rotação para todas as barras, principais ou não.

7.1.3 - PENT 2





Descrição das sub-rotinas:

- PEBC

esta sub-rotina forma o vetor de cargas $\{T_C\}$ da matriz de transferência para as barras carregadas, principais e/ou ramificações.

Se a barra carregada for uma ramificação, o vetor $\{T_C\}$ assim formado será colocado, posteriormente, no vetor de cargas das juntas $\{F_C\}$ da matriz fronteira.

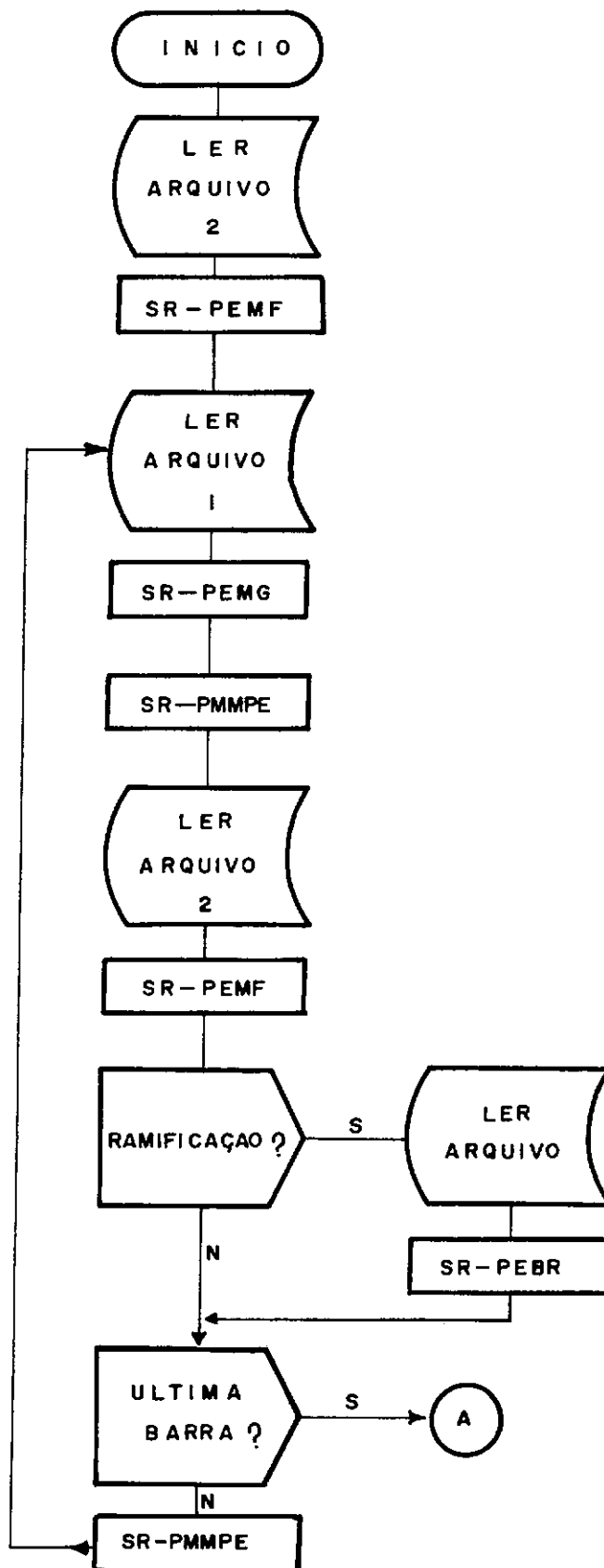
Esta sub-rotina chama ainda as seguintes:

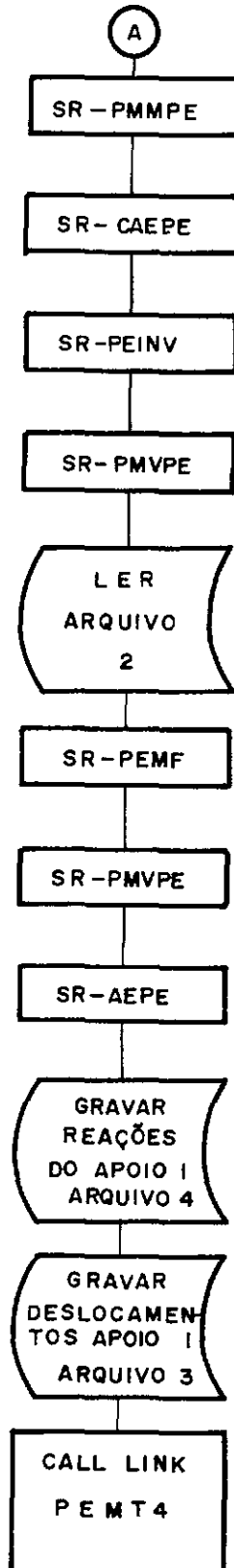
- PELIN sub-rotina para cargas lineares e/ou uniformes nas barras.
- PEUXX sub-rotina para momento torsor uniformemente distribuído na barra.
- PETUX sub-rotina para variação uniforme de temperatura.
- PEFCB sub-rotina para cargas concentradas nas barras.
- PEMCB sub-rotina para momentos concentrados nas barras.

- PEJC

esta sub-rotina forma o vetor de cargas $\{F_C\}$ da matriz fronteira para as juntas carregadas.

7.1.4 - PEMT 3

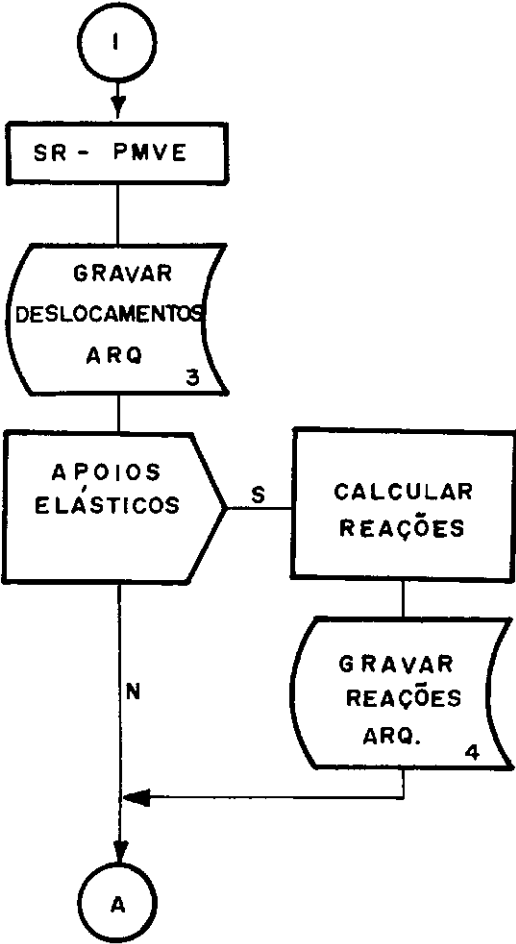
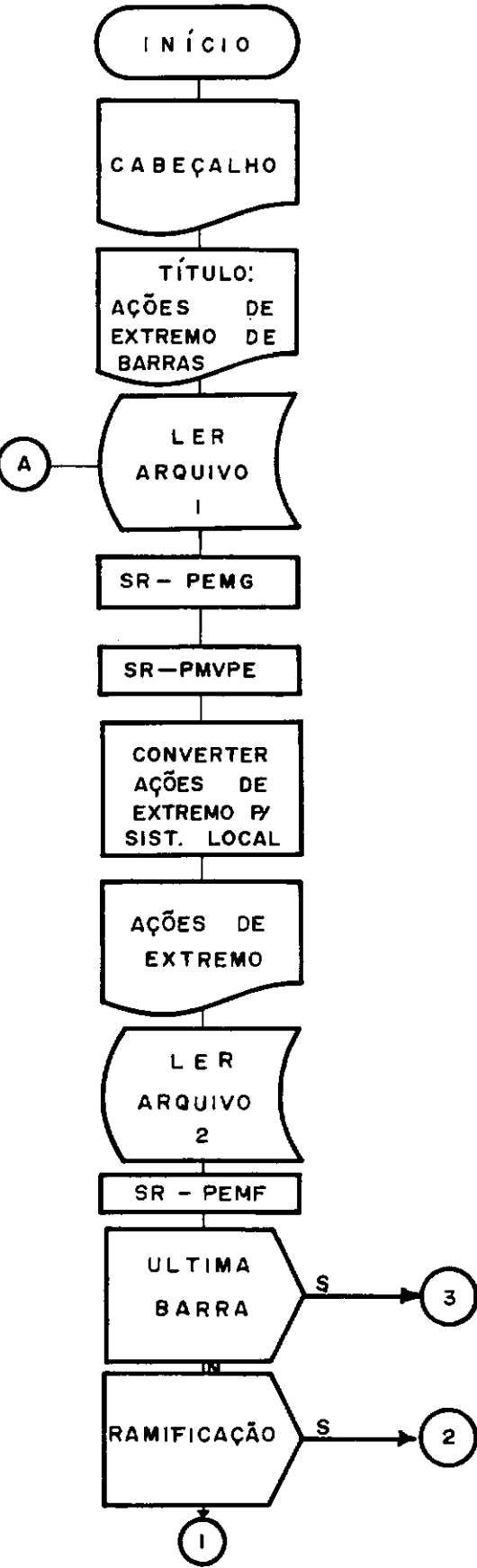


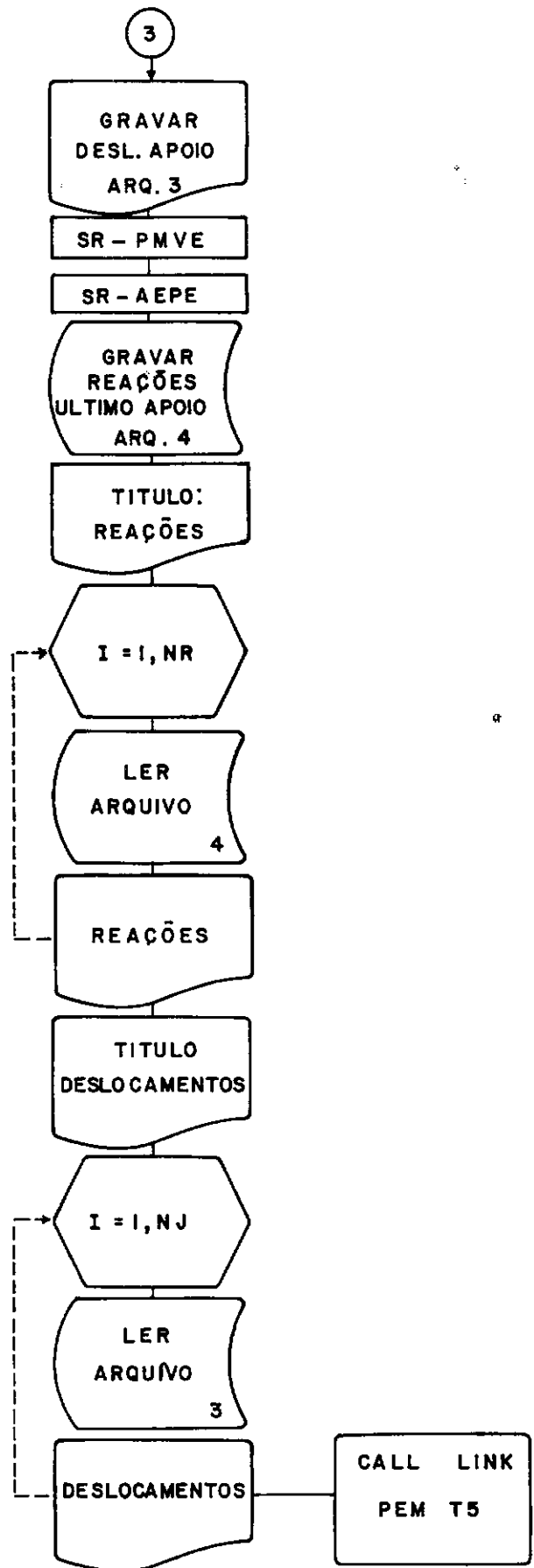


Descrição das sub-rotinas:

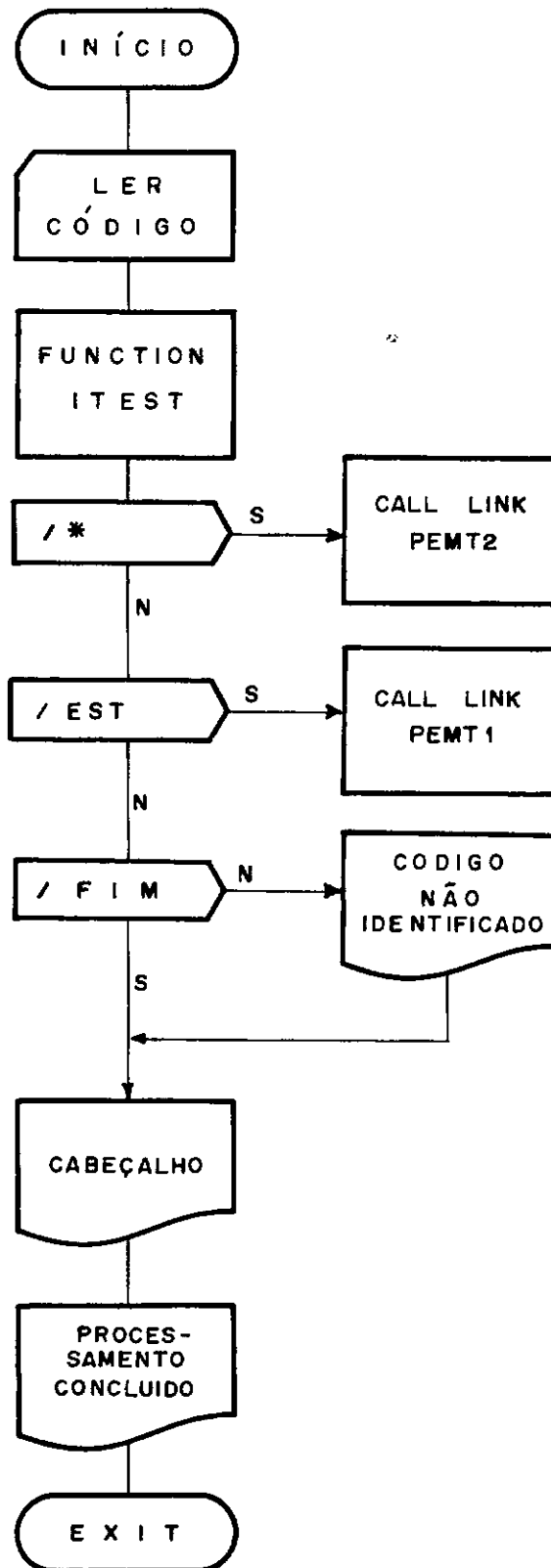
- PEMF sub-rotina para formar a matriz fronteira da junta.
- PEMG sub-rotina para formar a matriz de transferência das barras, no sistema global.
- PMMPE sub-rotina para produto de matrizes.
- PEBR sub-rotina para formar a matriz de rigidez da ramificação, sistema global, e incluí-la na submatriz $[FAD]$ da matriz fronteira.
- CAEPE sub-rotina que estabelece as condições de contorno dos apoios extremos.
- PEINV sub-rotina para inversão da matriz (6x6) do sistema.
- PMVPE sub-rotina para fazer o produto de matrizes por vetor.
- AEPE sub-rotina para calcular as reações dos apoios extremos.

7.1.5 - PENT 4





7.1.6 - PENT 5



7.2 - Estrutura dos dados:

7.2.1 - Considerações gerais:

Os dados para os programas estão organizados em 4 blocos:

- bloco de identificação da estrutura;
- bloco de definição preliminar;
- bloco de definição complementar;
- bloco de definição das cargas.

A ordem dos blocos não pode ser alterada e a delimitação de cada um é feita através dos cartões controle conforme mostra o fluxograma de entrada dos dados.

A ordem dos cartões em cada bloco é qualquer, podendo os mesmos serem repetidos ou suprimidos de acordo com a lógica estrutural. Cada cartão é identificado por um código mnemônico de 4 caracteres colocado, obrigatoriamente, a partir da 1a. coluna do cartão. A 5a. coluna do cartão é usada para complementar o código, quando necessário, através dos caracteres "=" (igual a) e " " (branco).

Os campos têm uma formatação rígida e todos os dados, inclusive os inteiros, devem conter o ponto decimal. Dessa forma, os dados inteiros não precisam ser justificados à direita do campo.

Para facilitar a exposição dos layouts de cartões, os dados inteiros serão representados por N e os reais por D.

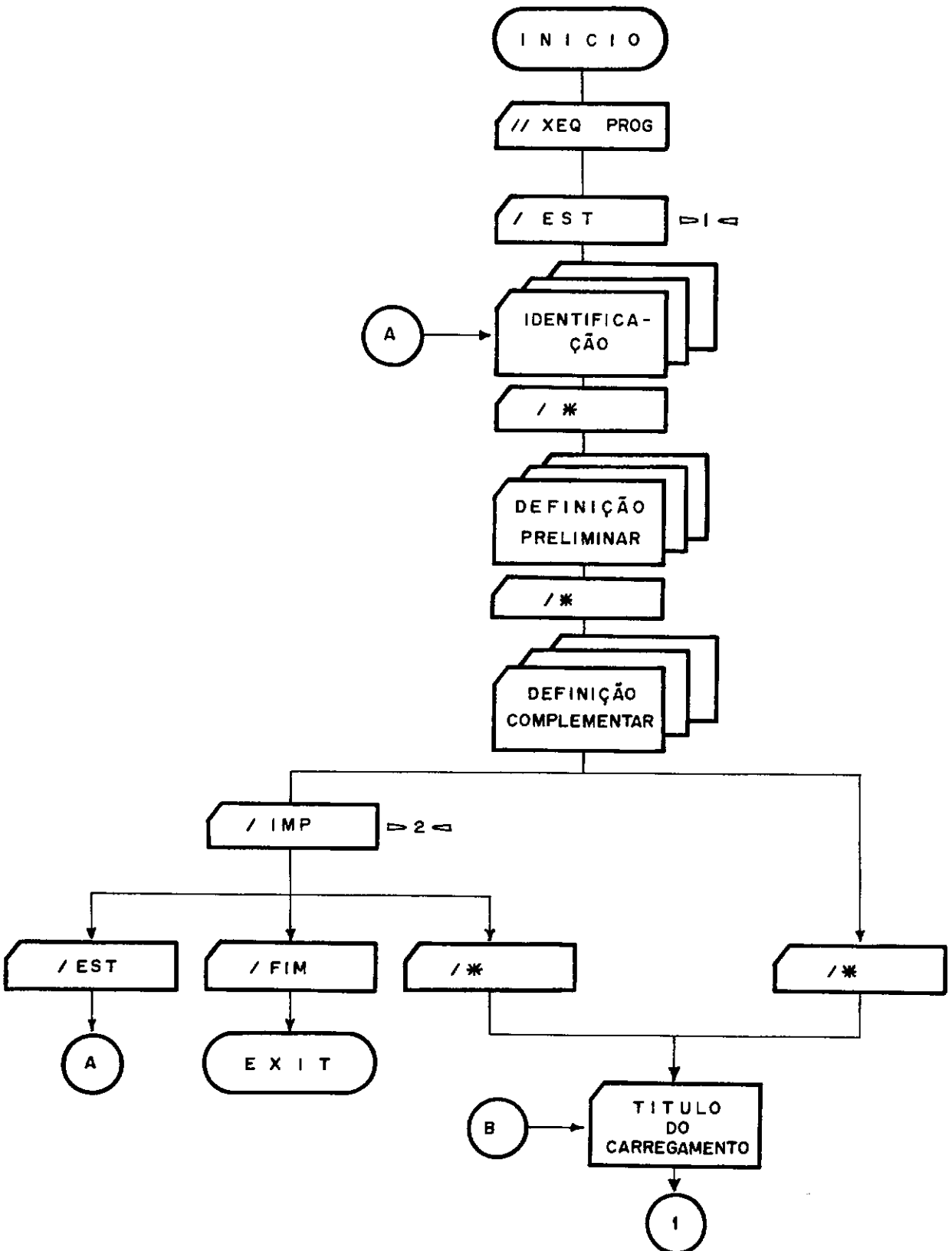
Alguns códigos usam as variáveis N1, N2 e N3 com o mesmo significado da declaração DO do FORTRAN:

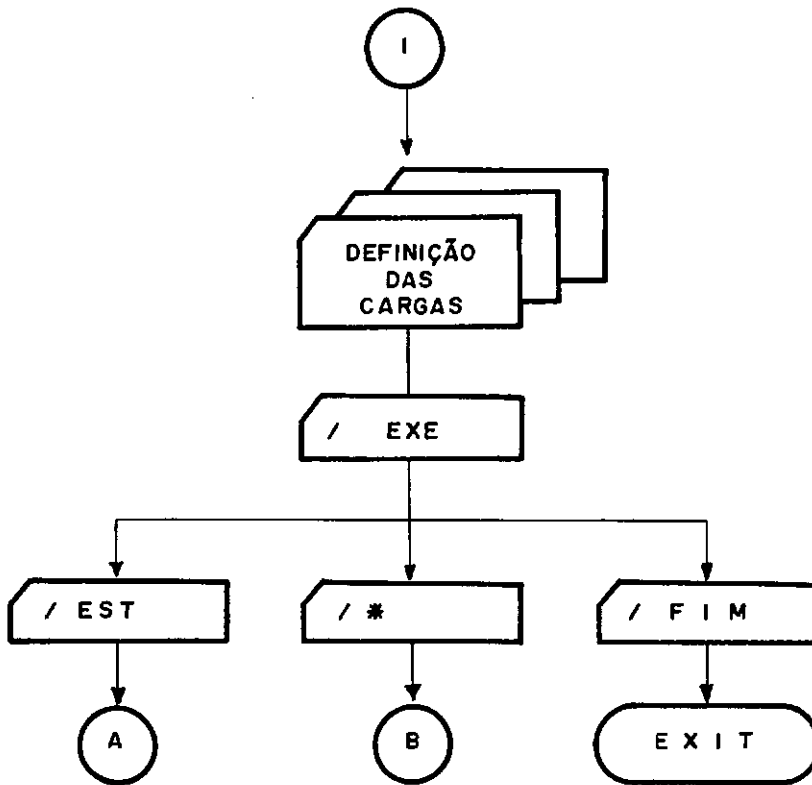
```
DO 55 I = N1, N2, N3.
```

A maioria dos códigos usa apenas N1 e N2, com N3 = 1.

Todos os códigos usam os caracteres "/" (barra) e "*" (asterístico) como prefixos. Com exceção dos códigos de identificação e de controle, o prefixo "/" indica tratar-se de dados referentes às barras, e o prefixo "*", referentes às juntas da estrutura.

FLUXOGRAMA DE ENTRADA DOS DADOS



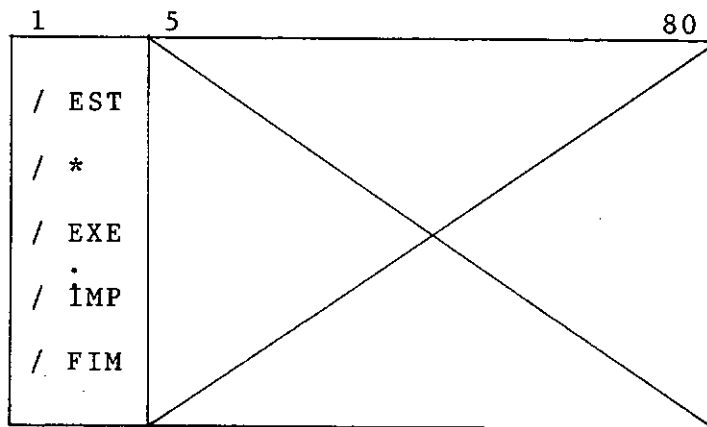


▷1◁ Nesta posição, este cartão controle é opcional.

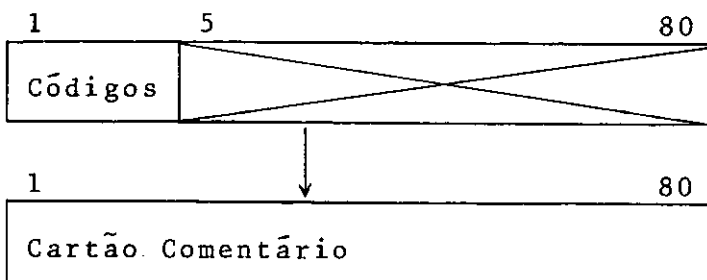
Nas demais posições, este e os outros cartões controles desviam o programa para as etapas indicadas pelos conectores.

▷2◁ Este cartão controle só é disponível nos programas PPMT e PEMT. Ele ordena a impressão, no formulário, dos comprimentos das barras e das componentes do vetor rotação.

7.2.2 - Layout dos cartões controle:



7.2.3 - Layout do bloco de identificação:



Os cartões de identificação são opcionais e, se usados de vem ser seguidos de um cartão comentário.

Os códigos disponíveis são: //JO, //FO, //RT. O conteúdo do cartão comentário é impresso no cabeçalho.

7.2.4 - Layout do bloco de definição preliminar:

	1	5 6	11	16	21	76
▷ 1 ◁	///N	N.				
▷ 2 ◁	//NR	N.				
▷ 3 ◁	*LRE	N.	N.	N.	N.	
	*LRD	N.	N.	N.	N.	
▷ 4 ◁	**SY	N.	N.	N.	N.	N.
	**SY	= N1.	N2.	N3.		
▷ 5 ◁	*SZZ	N.	N.			
▷ 6 ◁	*RAM	N.	N.	N.	N.	N.
	*RAM	= N1.	N2.	N3.		
▷ 7 ◁	**LX	N.	N.	N.	N.	N.
						
	*LZZ					

1 - Número de barras da estrutura (barras principais)

Uso = todos os programas (obrigatório).

2 - Número de ramificações

Uso = PPMT e PEMT (opcional)

- 3 - Lista de restrições dos apoios esquerdo e direito.

Uso = VBEMT, PPMT e PEMT

Default = assume-se engaste perfeito.

- 4 - Juntas suportes segundo Y

Uso = VCMT (obrigatório).

- 5 - Juntas suportes segundo ZZ.

Uso = VCMT

Default = assume-se rotação livre nos apoios extremos.

- 6 - Juntas com ramificação.

Uso = PPMT e PEMT (opcional)

- 7 - Liberações generalizadas nas juntas.

Uso = todos os programas (opcional).

7.2.5 - Layout do bloco de definição complementar:

	1	5 6	11	16	26	36	66 75
▷ 1 ◁	***X ***Y ***Z		N1. N2. D.		D. D.	D.	
	idem	=	N1. N2. D.				
▷ 2 ◁	**KX  *KZZ		N1. N2. D.		D. D.	D.	
	idem	=	N1. N2. D.				
▷ 3 ◁	///L ///L		N1. N2. D.		D. D.	D.	
		=	N1. N2. D.				
▷ 4 ◁	//AX //IX //IY //IZ ///E ///G		N1. N2. D.		D. D.	D.	
	idem	=	N1. N2. D.				
▷ 5 ◁	//XP //YP //ZP		N1. N2. D.		D. D.	D.	
	idem	=	N1. N2. D.				
▷ 6 ◁	/CRY /CRZ		N1. N2. D.		D. D.	D.	
	idem	=	N1. N2. D.				

1 - Coordenadas de todas as juntas.

Uso = PPMT e PEMT

Default = assume-se coordenada nula.

2 - Rigidezes de "molas" dos apoios elásticos.

Uso = todos os programas (opcional).

3 - Comprimento das barras.

Uso = VCMT e VBEMT

4 - Propriedades prismáticas e módulos de elasticidade.

Uso = todos os programas (obrigatório).

5 - Coordenadas de um ponto P para o cálculo do ângulo α (vide Gere-Weaver).

Uso = PEMT (opcional).

6 - Coeficientes de recalques da base elástica.

Uso = VBEMT (obrigatório).

Implantação = possível nos outros programas.

7.2.6 - Layout do bloco de definição das cargas:

	1	5	6	11	16	26	36	66	75
▷ 1 ◁	//UX //UY //UZ /UXX			N1.	N2.	D.	D.	D.	D.
	idem	=		N1.	N2.	D.			
▷ 2 ◁	//LX //LY //LZ			N1.	N2.	DJ.	DK.	DJ.	DK.
	idem	=		N1.	N2.	DJ.	DK.		
▷ 3 ◁	//CX  /CZZ			N1.	N2.	D.	A.	D.	A.
	idem	=		N1.	N2.	D.	A.		
▷ 4 ◁	/TUX /TUX			N1.	N2.	D.	α .	D.	α .
		=		N1.	N2.	D.	α .		
▷ 5 ◁	**DX  *DZZ			N1.	N2.	D.	D.	D.	D.
	idem	=		N1.	N2.	D.			
▷ 6 ◁	**CX  *CZZ			N1.	N2.	D.	D.	D.	D.
	idem	=		N1.	N2.	D.			
▷ 7 ◁	/CAR *CAR			N1.	N2.	N3.			

1 - Cargas uniformemente distribuidas.

Uso = todos os programas

2 - Cargas linearmente distribuidas.

Uso = todos os programas

3 - Cargas concentradas nas barras.

Uso = VCMT, PPMT e PEMT

4 - Variação uniforme de temperatura.

Uso = PPMT e PEMT

5 - Deformações impostas às juntas.

Uso = todos os programas

6 - Cargas nas juntas.

Uso = todos os programas

7 - Barras e/ou juntas carregadas.

Permitem alternar as cargas de um carregamento completo dado inicialmente.

As barras e/ou juntas especificadas serão carregadas enquanto as demais não.

Uso = todos os programas

CAPÍTULO VIII

PROGRAMAS EM FORTRAN

Os programas foram codificados em FORTRAN para o Computador IBM-1130, com 8K "palavras" de memória interna e um disco magnético. Eles permitem selecionar, através das chaves do Console, as seguintes unidades periféricas:

Leitoras IBM - 1442 e 2501

Impressoras IBM - 1132 e 1403.

O operador é instruído através de uma mensagem no Console para a seleção das unidades.

O disco magnético é usado para armazenar os dados de entrada, os deslocamentos e as reações. Somente o programa VCMT armazena as matrizes de Transferência dos vãos para, posteriormente, serem reduzidas com a eliminação da força cortante pela condição de deslocamento vertical nulo nos apoios fixos.

Com o objetivo de maximizar a capacidade de análise, os programas foram codificados inteiramente independentes um do outro ocupando, portanto, a área estritamente necessária para seu processamento. A única variação na demanda de memória interna ocorre com o dimensionamento dos vetores de controle de barras carregadas MC () e de juntas carregadas JC(). Atualmente os programas estão dimensionados para 200 barras e 201 juntas, havendo, porém, disponibilidade de memória se se desejar um número maior de barras. Além disso, o grande número de sub-rotinas na programação permite o uso, eficiente, do controle LOCAL para a redução da demanda de memória

Interna. Resta ainda a possibilidade de uma maior segmentação dos programas através de um número maior de declaração CALL LINK (não se pesquisou a capacidade máxima).

Com a finalidade de reduzir os erros de truncamento, to dos os programas utilizam do controle EXTENDED PRECISION.

PAGE 1 PR-VCMT

// JOB

PR-VCMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVCMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(TYPEWRITER)

INTEGER RE,W

COMMON RE,W

WRITE(1,11)

11 FORMAT(/1X'LIGAR CHAVE 1 P/1442,DESLIGAR P/2501READER'//1X'LIGAR C

*HAVE 2 P/1132,DESLIGAR P/1403PRINTER'//1X'POSICIONAR CHAVES E DAR

*START')

PAUSE

RE=2

W=3

CALL DATSW(1,I)

CALL DATSW(2,J)

GOTO(12,13),I

13 RE=8

12 GOTO(16,17),J

17 W=5

16 CALL LINK(VCMT1)

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

IOCS

CORE REQUIREMENTS FOR VCMT

COMMON 2 VARIABLES 2 PROGRAM 146

END OF COMPILATION

PAGE 1 PR-VCMT1

// JOB

PR-VCMT1

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVCMT1

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),DI(2),DD(15),DT(6)

DIMENSION LR(2),INP1(24),INP2(24),INP4(24),KOM(4)

COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP

DEFINE FILE 1(200,24,U,LDT),2(201,26,U,LDF),3(200,76,U,LDP)

DEFINE FILE 4(201,7,U,LDR),5(201,6,U,LDD)

DATA INP1//',E','S','T','/',',','/',',','O','/',',','/',',','F','O','/',',','/',',',

*'R','T','/',',','/',',','/',',','/',',','/',',','/',',','/',',',

DATA INP2//',N','/',',','/',',','/',',','/',',','/',',','/',',',

*'L','Y','/',',','/',',','/',',','/',',','/',',','/',',',

DATA INP4//',L','/',',','/',',','/',',','/',',','/',',','/',',',

*'K','Y','/',',','/',',','/',',','/',',','/',',','/',',',

DATA LR/2*0/,DT/6*0./

NBAR=0

100 DO 100 I=1,4

LCAB(I)=2

CALL CABV(I)

LCAB(I)=1

C-----LER DADOS DE IDENTIFICACAO

101 READ(RE,102)KOM

102 FORMAT(4A1)

WRITE(W,171)KOM

171 FORMAT(/1X,4A1)

I=0

DO 110 J=1,21,4

K=J+3

I=I+1

IF(ITEST(INP1,J,K,KOM,1))110,103,110

103 GOTO(101,115,125,135,145,113),I

110 CONTINUE

104 WRITE(W,106)KOM

106 FORMAT(/20X'NAO EXISTE O CODIGO ('4A1')'////)

CALL EXIT

115 READ(RE,107)OBRA

107 FORMAT(20A4)

WRITE(W,155)OBRA

155 FORMAT(1X,20A4)

LCAB(2)=1

```

125  GOTO 101
      READ(RE,107)FULA
      WRITE(W,155)FULA
      LCAB(3)=1
      GOTO 101
135  READ(RE,107)RESP
      WRITE(W,155)RESP
      LCAB(4)=1
      GOTO 101
145  LCAB(1)=2
      GOTO 101
127  WRITE(W,128)KOM,ID,(DD(K),K=1,N)
128  FORMAT(1X,4A1,A1,15F5.0)
C-----LER DADOS DE DEFINICAO
113  READ(RE,114)KOM,ID,DD
114  FORMAT(4A1,A1,15F5.0)
      I=0
      DO 140 J=1,21,4
      K=J+3
      I=I+1
      IF(ITEST(INP2,J,K,KOM,1))140,116,140
116  GOTO(165,205,205,225,225,195),I
140  CONTINUE
      GOTO 104
195  WRITE(W,171)KOM
      GOTO 245
165  CALL CONCA(DD,1,200,IER,2,N)
      IF(IER)121,121,198
121  NBAR=DD(1)
      NJ=NBAR+1
      WRITE(W,112)KOM,DD(1)
112  FORMAT(1X,4A1,1X,F5.0)
      DO 130 I=1,NJ
130  WRITE(2,'I')LR,DT
      GOTO 113
205  K=I-1
      CALL CONCA(DD,15,NJ,IER,4,N)
      IF(IER)122,122,198
122  CALL VCJS(DD,ID,K,N)
      GOTO 127
225  K=I-3
      CALL CONCA(DD,15,NJ,IER,4,N)
      IF(IER)123,123,198
123  CALL VCLIB(DD,K,N)
      GOTO 127
129  WRITE(W,132)KOM,ID,DI,(DT(J),J=1,INC)
132  FORMAT(1X,4A1,A1,2F5.0,6F15.2)
      GOTO 245
131  WRITE(W,133)KOM,ID,DI,(DT(J),J=1,INC)
133  FORMAT(1X,4A1,A1,2F5.0,6E15.6)
C-----LER DADOS DE DEFINICAO COMPLEMENTAR

```


PAGE 3 PR-VCMT1

```
245 READ(RE,117)KOM,ID,DI,DT
117 FORMAT(4A1,A1,2F5.0,6F10.0)
    I=0
    DO 150 J=1,21,4
    K=J+3
    I=I+1
    IF(1TEST(INP4,J,K,KOM,1))150,118,150
118 GOTO(255,255,255,285,285,305),I
150 CONTINUE
    GOTO 104
255 CALL CONCB(DI,ID,INC,IER,NBAR,3)
    IF(1ER)124,124,199
124 CALL VCDB(DT,DI,I,INC)
    GOTO(129,131,131),I
285 K=I-1
    CALL CONCB(DI,ID,INC,IER,NJ,4)
    IF(1ER)126,126,199
126 CALL VCDJ(DT,DI,K,INC)
    GOTO 131
305 WRITE(W,171)KOM
    CALL LINK(VCMT2)
198 WRITE(W,196)KOM,DD,IER
196 FORMAT(/5X,4A1,15F5.0//5X'ERRO='15)
    CALL EXIT
199 WRITE(W,197)KOM,ID,DI,IER
197 FORMAT(/5X,4A1,A1,2F5.0//5X'ERRO='15)
    CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR VCMT1
COMMON 188 VARIABLES 196 PROGRAM 758

END OF COMPILATION

PAGE 1 PR-VCMT2

// JOB

PR-VCMT2

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVCMT2

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION DD(6),DI(2),KOM(4),LCAB(4),OBRA(20),FULA(20),RESP(20)

DIMENSION TITUL(20),MC(200),JC(201),INP5(44),D(3)

COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,MC,JC,TITUL

EQUIVALENCE (D(1),DI(1)),(DD(1),D(3))

DEFINE FILE 1(200,24,U,LDT),2(201,26,U,LDF),3(200,76,U,LDP)

DEFINE FILE 4(201,7,U,LDR),5(201,6,U,LDD)

DATA INP5/'/','/','/','U','Y','/','/','/','L','Y','/','/','/','C','Y','/','/','C',

'Z','Z','','*','C','Y','*','C','Z','Z','*','*','D','Y','*','D','Z'

'Z','/','E','X','E','/','C','A','R','','C','A','R'/

DATA KJ,KM/0,0/

C-----ZERAR CONTROLES DE BARRAS E JUNTAS CARREGADAS

NJ=NBAR+1

DO 200 I=1,NBAR

MC(I)=0

200 JC(I)=0

JC(NJ)=0

C-----IMPRIMIR CABECALHO DE IDENTIFICACAO

CALL CABV(1)

C-----LER E IMPRIMIR TITULO DO CARREGAMENTO

READ(RE,201)TITUL

WRITE(W,202)TITUL

201 FORMAT(20A4)

202 FORMAT(7/1X'*LIST. DADOS* '20A4/1X,80('='))

GOTO 203

223 WRITE(W,224)KOM,ID,DI,(DD(J),J=1,INC)

224 FORMAT(1X,4A1,A1,2F5.0,6F15.2)

C-----LER DADOS DO CARREGAMENTO

203 READ(RE,204)KOM,ID,DI,DD

204 FORMAT(4A1,A1,2F5.0,6F10.0)

I=0

DO 210 J=1,41,4

K=J+3

I=I+1

IF(I TEST(INP5,J,K,KOM,1))210,206,210

206 GOTO(315,315,315,315,325,325,325,325,335,345,345),I

210 CONTINUE

WRITE(W,207)KOM

PAGE 2 PR-VCMT2

```
207 FORMAT(/1X'NAO EXISTE O CODIGO ('4A1')'///)
CALL EXIT
315 CALL CONCB(DI,ID,INC,IER,NBAR,3)
IF(IER)221,221,298
221 CALL VCCB(DD,DI,I,KOM,INC)
GOTO 203
325 CALL CONCB(DI,ID,INC,IER,NJ,4)
IF(IER)222,222,298
222 CALL VCCJ(DD,DI,I,INC)
GOTO 223
345 CALL CONCA(D,3,NJ,IER,4,N)
IF(IER)226,226,298
226 IF(N-2)227,228,229
227 IER=7
GOTO 298
C-----INICIALIZAR CONTROLES DE CARGAS CONFORME /CAR *CAR
228 N3=1
GOTO 236
229 N3=D(3)
236 N1=D(1)
N2=D(2)
IF(I-10)231,231,232
231 IF(KM)241,241,242
241 DO 220 J=1,NBAR
220 MC(J)=0
KM=1
242 DO 230 J=N1,N2,N3
230 MC(J)=1
GOTO 233
232 IF(KJ)243,243,244
243 DO 240 J=1,NJ
240 JC(J)=0
KJ=1
244 DO 250 J=N1,N2,N3
250 JC(J)=1
233 WRITE(W,234)KOM,ID,(D(K),K=1,N)
234 FORMAT(1X,4A1,A1,3F5.0)
GOTO 203
335 WRITE(W,211)KOM
211 FORMAT(/1X,4A1)
CALL LINK(VCMT3)
298 WRITE(W,299)KOM,ID,DI,IER
299 FORMAT(/5X,4A1,A1,2F5.0//5X'ERRO='I5)
CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR VCMT2

COMMON 648 VARIABLES 128 PROGRAM 560

PAGE 1 PR-VCMT3

// JOB

PR-VCMT3

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVCMT3

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION LR(2),DMF(8),DMT(8),T(5,5),F(5,5),P(5,5),LCAB(4),VE(5)

DIMENSION V(5),VR(2),OBRA(20),FULA(20),RESP(20),KOM(4),TITUL(20)

DIMENSION MC(200),JC(201),INP6(12)

COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,MC,JC,TITUL

DEFINE FILE 1(200,24,U,LDT),2(201,26,U,LDF),3(200,76,U,LDP)

DEFINE FILE 4(201,7,U,LDR),5(201,6,U,LDD)

DATA INP6/' ','*',' ',' ',' ',' ',' ',' ','E','S','T',' ',' ','F',' ','I',' ','M'/'

DATA VE/4*0./,LDF,LDT,LDP,LDR,LDD/2,1,1,1,1/

NJ=NBAR+1

VE(5)=1.

C-----PRIMEIRA PASSAGEM PELA ESTRUTURA

401 INDF=JC(LDF)

INDT=MC(LDT)

READ(1'LDT)DMT

READ(2'LDF)LR,DMF

DO 520 I=1,3

IF(DMT(I))491,491,520

520 CONTINUE

CALL FMT(DMT,P,INDT)

402 IF(LR(1))403,403,404

C-----FORMAR MATRIZ FRONTEIRA 5X5

403 CALL FMT(DMF,F,INDF,LR)

CALL PRMM(F,P,T,5)

INDT=MC(LDT)

READ(1'LDT)DMT

DO 530 I=1,3

IF(DMT(I))491,491,530

530 CONTINUE

CALL FMT(DMT,F,INDT)

CALL PRMM(F,T,P,5)

INDF=JC(LDF)

READ(2'LDF)LR,DMF

GOTO 402

C-----ARQUIVAR M. TRANSF. DO VAO LDT

404 WRITE(3'LDP)P,LDT

IF(LDT-NJ)401,406,406

406 LDP=1

C-----SEGUNDA PASSAGEM P/ ELIMINAR DESCONTINUIDADES NOS APOIOS FIXOS

READ(3,LDP)T,JUNTA

C-----REDUZIR T(5,5) EM P(3,3)

CALL MTRED(T,P)

413 IF(JUNTA=NJ)411,412,412

411 READ(2,JUNTA)LR,DMF

IND=JC(JUNTA)

C-----FORMAR M. FRONTEIRA REDUZIDA. F(3,3)

CALL FMFR(DMF,F,IND)

CALL PRMM(F,P,T,3)

READ(3,LDP)P,JUNTA

CALL MTRED(P,F)

CALL PRMM(F,T,P,3)

GOTO 413

412 IND=JC(NJ)

C-----CONDICOES DE CONTORNO

J=2

K=1

I=2

READ(2,NJ)LR,DMF

CALL FMFR(DMF,F,IND)

F(2,2)=1.

IF(LR(2))512,512,408

512 F(2,3)=-F(2,3)

F(2,1)=-F(2,1)

GOTO 407

408 F(2,3)=0.

F(1,3)=-F(1,3)

J=1

407 CALL PRMM(F,P,T,3)

LDF=1

IND=JC(LDF)

READ(2,LDF)LR,DMF

CALL FMFR(DMF,F,IND)

F(2,2)=1.

IF(LR(2))409,409,511

511 F(2,3)=0.

K=2

I=4

C-----RESOLUCAO DO SISTEMA RESULTANTE

409 CALL PRMM(T,F,P,3)

VE(I)=-P(J,3)/P(J,K)

C-----TERCEIRA PASSAGEM PELA ESTRUTURA

LDT=1

LDP=1

C-----IMPRIMIR TITULO DO CARREGAMENTO

WRITE(W,421)TITUL

421 FORMAT(/,1X,'*RESULTADOS* '20A4/1X,80('='))

WRITE(W,422)

422 FORMAT(/,1X,'ACOES NAS EXTREMIDADES DAS BARRAS'/)

WRITE(W,423)

```

423  FORMAT(1X'BARRA'2(3X'CORTANTE'4X'M.FLETOR'))
      CALL FMF(DMF,F,IND,LR)
      F(3,3)=1.
      F(4,4)=1.
424  IF(F(1,3))414,414,416
416  VE(3)=0.
      F(1,3)=0.
414  IF(F(2,4))417,417,418
418  VE(4)=0.
      F(2,4)=0.
417  IF(IND)484,484,486
484  DMF(5)=0.
      DMF(6)=0.
486  IF(LDF=NJ)426,426,427
426  CALL PRMV(F,VE,V,5)
      WRITE(5'LDD)V(1),V(2)
C-----REACOES NOS APOIOS FIXOS E ELASTICOS
      IF(LR(1))428,428,429
428  IF(DMF(3))446,446,447
446  IF(DMF(4))433,433,447
447  VR(1)=-DMF(3)*V(1)
      VR(2)=-DMF(4)*V(2)
      GOTO 431
429  READ(3'LDP)P
C-----FORCA CORTANTE NA EXTREMIDADE J DO VAO
      V(3)=- (P(1,2)*V(2)+P(1,4)*V(4)+P(1,5))/P(1,3)
      VR(1)=VE(3)+V(3)-DMF(5)
      VR(2)=-DMF(4)*V(2)
      V(1)=0.
      IF(LR(2))431,431,432
432  VR(2)=V(4)-DMF(6)
431  LD=LDF-1
      WRITE(4'LDR)LD,VR
433  IND=MC(LDT)
      READ(1'LDT)DMT
      CALL FMT(DMT,T,IND)
      CALL PRMV(T,V,VE,5)
      LD=LDT-1
C-----ACOES DE EXTREMO DE BARRAS
      WRITE(W,405)LD,V(3),V(4),VE(3),VE(4)
405  FORMAT(1X,I5,2(F11.2,1X,F11.2))
      IND=JC(LDF)
      READ(2'LDF)LR,DMF
      CALL FMF(DMF,F,IND,LR)
      GOTO 424
C-----REACOES E DESLOCAMENTOS JUNTA NJ
427  VR(1)=VE(3)-DMF(5)
      VR(2)=-DMF(4)*VE(2)
      WRITE(5'LDD)VE(1),VE(2)
      IF(LR(2))487,487,488
488  VR(2)=VE(4)-DMF(6)

```

```

487 LD=LDF-1
WRITE(4,LDR)LD,VR
NJR=LDR-1
WRITE(W,415)
415 FORMAT(/,1X'REACOES'/)
WRITE(W,425)
425 FORMAT(1X'JUNTA'4X'REACAO Y'4X'REACAO Z')
DO 400 I=1,NJR
READ(4,I)NJU,VR
WRITE(W,435)NJU,VR
435 FORMAT(1X,I5,2F12.2)
400 CONTINUE
WRITE(W,445)
445 FORMAT(/,1X'DESLOCAMENTOS GENERALIZADOS'/)
WRITE(W,455)
455 FORMAT(1X'JUNTA'4X'DESLOC Y'4X'DESLOC Z')
DO 410 I=1,NJ
READ(5,I)V(1),V(2)
WRITE(W,465)I,V(1),V(2)
465 FORMAT(1X,I5,2F12.4)
410 CONTINUE
C-----LER CODIGO DE DECISAO
READ(RE,475)KOM
475 FORMAT(4A1)
I=0
DO 420 J=1,9,4
K=J+3
I=I+1
IF( ITEST(INP6,J,K,KOM,1))420,477,420
477 GOTO(481,482,483),I
420 CONTINUE
C-----FINALIZAR PROCESSAMENTO
WRITE(W,492)KOM
492 FORMAT(/5X'CODIGO ('4A1') NAO IDENTIFICADO')
GOTO 493
483 WRITE(W,501)KOM
501 FORMAT(/,1X,4A1)
493 CALL CABV(1)
WRITE(W,502)
502 FORMAT(/1X'PROCESSAMENTO CONCLUIDO'//1X'HR. MAQUINA INICIAL='//1X
*'HR. MAQUINA FINAL ='//1X'TEMPO PROCESSAMENTO=' HRS.* MIN.*
**'/)
WRITE(W,503)
503 FORMAT('1')
CALL EXIT
C-----PROCESSAR NOVO CARREGAMENTO
481 CALL LINK(VCMT2)
C-----PROCESSAR NOVA ESTRUTURA
482 CALL LINK(VCMT1)
491 IER=6
LD=LDT-1

```

PAGE 5 PR-VCMT3

```
494 WRITE(W,494)LD,IER  
    FORMAT(/5X'BARRA='I5,3X'ERRO='I5)  
    CALL EXIT  
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR VCMT3
COMMON 648 VARIABLES 392 PROGRAM 1330

END OF COMPILATION

PAGE 1 SR-VCMT

// JOB

SR-VCMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR SR-CABV - CABECALHO - VCMT E VBEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE CABV(NE)

INTEGER RE,W

DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20)

COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP

GOTO(11,12),NE

11 WRITE(W,14)

14 FORMAT('1'// 5X'ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA'/)

GOTO 15

12 WRITE(W,16)

16 FORMAT('1'//5X'ANALISE VIGA BASE ELASTICA M. MATRIZES TRANSFERENCI

*A'/)

15 DO 100 I=1,4

IF(LCAB(I)-1)102,102,100

102 GOTO(105,115,125,135),I

105 WRITE(W,106)

106 FORMAT(5X'ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO'/)

GOTO 100

115 WRITE(W,107)OBRA

107 FORMAT(5X,20A4)

GOTO 100

125 WRITE(W,107)FULA

GOTO 100

135 WRITE(W,107)RESP

100 CONTINUE

WRITE(W,111)

111 FORMAT(1X,40('='))

RETURN

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR CABV

COMMON 188 VARIABLES 2 PROGRAM 198

RELATIVE ENTRY POINT ADDRESS IS 006E (HEX)

END OF COMPILATION

PAGE 2 SR=VCMT

// FOR SR-VCJS - ARQUIVA JUNTAS SUPORTES

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE VCJS(DD, ID, IND, N)
      DIMENSION DD(15), LR(2)
      IDD=ID/8
      IF( (IDD-2056)101,102,101
102   DO 100 I=1,N
      J=DD(I)
      READ(2,J)LR
      LR(IND)=1
      WRITE(2,J)LR
100   CONTINUE
      RETURN
101   ID=32320
      N1=DD(1)
      N2=DD(2)
      N3=DD(3)
      IF(N3)103,103,104
103   N3=1
104   DO 110 I=N1,N2,N3
      READ(2,I)LR
      LR(IND)=1
      WRITE(2,I)LR
110   CONTINUE
      RETURN
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCJS
COMMON 0 VARIABLES 12 PROGRAM 152

RELATIVE ENTRY POINT ADDRESS IS 0011 (HEX)

END OF COMPILATION

// FOR SR-VCLIB - LIBERACOES GENERALIZADAS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE VCLIB(DD, IND, N)
      DIMENSION DD(15), LR(2), DMF(8)
      DO 100 I=1,N
      J=DD(I)
      READ(2,J)LR,DMF
      DMF(IND)=1.E+15
      WRITE(2,J)LR,DMF
```

PAGE 3 SR-VCMT

100 CONTINUE
RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCLIB
COMMON 0 VARIABLES 30 PROGRAM 68

RELATIVE ENTRY POINT ADDRESS IS 0023 (HEX)

END OF COMPILATION

// FOR SR-VCDB - DADOS DAS BARRAS - VCMT E VBEMT

*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE VCDB(DD,DI,NV,INC)
DIMENSION DD(6),DI(2),DMT(8)
N1=DI(1)
N2=DI(2)
K=0
DO 100 I=N1,N2
K=1+K*INC
READ(1,I)DMT
DMT(NV)=DD(K)
WRITE(1,I)DMT
100 CONTINUE
INC=K
RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCDB
COMMON 0 VARIABLES 30 PROGRAM 90

RELATIVE ENTRY POINT ADDRESS IS 0020 (HEX)

END OF COMPILATION

// FOR SR-VCDJ - DADOS DAS JUNTAS - ARQUIVO

*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE VCDJ(DD,DI,NV,INC)
DIMENSION DD(6),DI(2),LR(2),DMF(8)

PAGE 4 SR-VCMT

```
      N1=DI(1)
      N2=DI(2)
      K=0
      DO 100 I=N1,N2
      K=1+K*INC
      READ(2,I)LR,DMF
      DMF(NV)=DD(K)
      WRITE(2,I)LR,DMF
100   CONTINUE
      INC=K
      RETURN
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCDJ
COMMON 0 VARIABLES 32 PROGRAM 98

RELATIVE ENTRY POINT ADDRESS IS 0023 (HEX)

END OF COMPILATION

// FOR SR-VCCB - CARGAS NAS BARRAS - ROTINA PRINCIPAL

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE VCCB(DD, ID, DI, INDS, KOM, INC)
      INTEGER RE, W
      DIMENSION DD(6), DI(2), DMT(8), LCAB(4), OBRA(20), FULA(20), RESP(20)
      DIMENSION MC(200), KOM(4)
      COMMON RE, W, NBAR, LCAB, OBRA, FULA, RESP, MC
      N1=DI(1)
      N2=DI(2)
      K=0
      IF((N2-N1-2)*INC)101,101,102
102   IF(INDS-1)101,101,103
101   DO 100 I=N1,N2
      K=1+K*INC
      READ(1,I)DMT
      DO 120 J=1,3
      IF(DMT(J))121,121,120
120   CONTINUE
      IF(MC(I))106,106,107
106   DO 110 J=5,8
110   DMT(J)=0.
      MC(I)=1
107   GOTO(115,125,135,145), INDS
C-----CARGA UNIFORME
115   CALL VCUY(DMT, DD, K)
```

PAGE 5 SR-VCMT

```
      GOTO 109
C-----CARGA LINEAR
125    CALL VCLY(DMT,DD,K)
      GOTO 109
C-----CARGA CONCENTRADA
135    CALL VCCY(DMT,DD,K,INC,N1,N2)
      GOTO 109
C-----MOMENTO APLICADO
145    CALL VC2Z(DMT,DD,K,INC,N1,N2)
109    WRITE(1,'I')DMT
100    CONTINUE
      WRITE(W,112)KOM,ID,DI,(DD(J),J=1,K)
112    FORMAT(1X,4A1,A1,2F5.0,6F15.2)
      RETURN
103    IER=5
      WRITE(W,104)KOM,ID,DI,IER
104    FORMAT(//5X,4A1,A1,2F5.0//5X'ERRO='I5)
      CALL EXIT
121    IER=6
      WRITE(W,122)I,IER
122    FORMAT(/5X'BARRA='I5,3X'ERRO='I5)
      CALL EXIT
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCCB
COMMON 388 VARIABLES 32 PROGRAM 318

RELATIVE ENTRY POINT ADDRESS IS 0053 (HEX)

END OF COMPILATION

// FOR SR-VCUY - CARGA UNIFORME - VCMT E VBEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE VCUY(DMT,DD,K)
      DIMENSION DMT(8),DD(6)
      AUX=DD(K)*DMT(1)
      DMT(7)=DMT(7)-AUX
      AUX=AUX*DMT(1)/2.
      DMT(8)=DMT(8)+AUX
      AUX=(AUX*DMT(1))/(3.*DMT(2)*DMT(3))
      DMT(6)=DMT(6)+AUX
      DMT(5)=DMT(5)+(AUX*DMT(1))/4.
      RETURN
      END
```

PAGE 6 SR-VCMT

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCUY
COMMON 0 VARIABLES 14 PROGRAM 116

RELATIVE ENTRY POINT ADDRESS IS 0017 (HEX)

END OF COMPILATION

```
// FOR SR-VCLY - CARGA LINEAR - VCMT E VBEMT
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
      SUBROUTINE VCLY(DMT,DD,K)
      DIMENSION DD(6),DMT(8)
      A=DD(K)
      K=K+1
      B=DD(K)
      AUX=DMT(2)*DMT(3)
      XL=DMT(1)
      DMT(7)=DMT(7)-(A+B)*XL/2.
      XL=XL*XL
      DMT(8)=DMT(8)+(2.*A+B)*XL/6.
      DMT(6)=DMT(6)+((3.*A+B)*XL*DMT(1))/(24.*AUX)
      XL=XL*XL
      DMT(5)=DMT(5)+((4.*A+B)*XL)/(120.*AUX)
      RETURN
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCLY
COMMON 0 VARIABLES 24 PROGRAM 176

RELATIVE ENTRY POINT ADDRESS IS 002B (HEX)

END OF COMPILATION

```
// FOR SR-VCCY - CARGA CONCENTRADA
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
      SUBROUTINE VCCY(DMT,DD,K,INC,N1,N2)
      DIMENSION DMT(8),DD(6)
      NV=1
      J=K-1
```

PAGE 7 SR-VCMT

```
101 IF((N2-N1)*INC)101,101,102
102 NV=3
DO 100 I=1,NV
J=J+1
IF(DD(J))104,106,104
104 P=DD(J)
J=J+1
A=DMT(1)-DD(J)
DMT(7)=DMT(7)-P
DMT(8)=DMT(8)+P*A
AUX=DMT(2)*DMT(3)
DMT(6)=DMT(6)+(P*A*A)/(2.*AUX)
DMT(5)=DMT(5)+(P*A*A*A)/(6.*AUX)
K=J
100 CONTINUE
106 RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCCY
COMMON 0 VARIABLES 24 PROGRAM 194

RELATIVE ENTRY POINT ADDRESS IS 0020 (HEX)

END OF COMPILATION

// FOR SR-VCZZ = MOMENTO APLICADO

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE VCZZ(DMT,DD,K,INC,N1,N2)
DIMENSION DMT(8),DD(6)
NV=1
J=K-1
IF((N2-N1)*INC)101,101,102
101 NV=3
102 DO 100 I=1,NV
J=J+1
IF(DD(J))104,106,104
104 XM=DD(J)
J=J+1
A=DMT(1)-DD(J)
DMT(8)=DMT(8)-XM
AUX=(XM*A)/(DMT(2)*DMT(3))
DMT(6)=DMT(6)-AUX
DMT(5)=DMT(5)-AUX*A/2.
K=J
100 CONTINUE
```

PAGE 8 SR-VCMT

106 RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCZZ
COMMON 0 VARIABLES 22 PROGRAM 166

RELATIVE ENTRY POINT ADDRESS IS 001B (HEX)

END OF COMPILATION

// FOR SR-VCCJ - DESLOCAMENTOS IMPOSTOS E CARGAS NAS JUNTAS

*ONE WORD INTEGERS
*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE VCCJ(DD,DI,NV,INC)
  INTEGER RE,W
  DIMENSION DD(6),DI(2),DMF(8),LR(2),LCAB(4),OBRA(20),FULA(20)
  DIMENSION RESP(20),MC(200),JC(201)
  COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,MC,JC
  N1=DI(1)
  N2=DI(2)
  K=0
  DO 100 I=N1,N2
    K=1+K*INC
    READ(2,I)LR,DMF
    IF(JC(I))106,106,107
106  DO 110 J=5,8
110  DMF(J)=0.
    JC(I)=1
107  DMF(NV)=DMF(NV)+DD(K)
    WRITE(2,I)LR,DMF
100  CONTINUE
    INC=K
    RETURN
  END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VCCJ
COMMON 588 VARIABLES 34 PROGRAM 148

RELATIVE ENTRY POINT ADDRESS IS 002A (HEX)

END OF COMPILATION

PAGE 9 SR-VCMT

// FOR SR-FMT - MATRIZ TRANSFERENCIA - FLEXAO LIVRE - VCMT E VBEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE FMT(DM,T,IND)
      DIMENSION DM(8),T(5,5)
      DO 100 I=1,5
      DO 110 J=1,5
110    T(I,J)=0.
100    T(I,I)=1.
      T(3,3)=-1.
      T(4,4)=-1.
      AUX=DM(1)
      T(1,2)=AUX
      T(4,3)=AUX
      AUX=AUX/(DM(2)*DM(3))
      T(2,4)=-AUX
      T(2,3)=AUX*DM(1)/2.
      T(1,4)=-T(2,3)
      T(1,3)=T(2,3)*DM(1)/3.
      IF(IND)108,108,109
C-----VETOR DE CARGAS
109    DO 120 I=1,4
      K=I+4
120    T(I,5)=DM(K)
108    RETURN
      END
```

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR FMT

COMMON 0 VARIABLES 18 PROGRAM 216

RELATIVE ENTRY POINT ADDRESS IS 0021 (HEX)

END OF COMPILE

// FOR SR-FMF - MATRIZ FRONTEIRA

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE FMF(DM,F,IND,LR)
      DIMENSION DM(8),F(5,5),LR(2)
      DO 100 I=1,5
      DO 110 J=1,5
110    F(I,J)=0.
100    F(I,I)=1.
      F(3,3)=-1.
      F(4,4)=-1.
```

PAGE 10 SR-VCMT

```
102 IF(IND)101,101,102
    F(1,5)=DM(7)
    F(2,5)=DM(8)
    IF(LR(1))103,103,104
103 F(3,5)=DM(5)
104 IF(LR(2))106,106,101
106 F(4,5)=DM(6)
101 F(1,3)=DM(1)
    F(2,4)=DM(2)
    F(3,1)=-DM(3)
    F(4,2)=-DM(4)
    RETURN
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR FMF
COMMON 0 VARIABLES 10 PROGRAM 182

RELATIVE ENTRY POINT ADDRESS IS 0012 (HEX)

END OF COMPILATION

// FOR SR=MTRED - MATRIZ TRANSF. REDUZIDA - VCMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE MTRED(T,P)
DIMENSION T(5,5),P(5,5)
AUX=T(2,3)/T(1,3)
P(1,1)=-AUX*T(1,2)+T(2,2)
P(1,2)=T(2,4)-AUX*T(1,4)
P(1,3)=T(2,5)-AUX*T(1,5)
AUX=T(4,3)/T(1,3)
P(2,1)=-AUX*T(1,2)+T(4,2)
P(2,2)=-AUX*T(1,4)+T(4,4)
P(2,3)=T(4,5)-AUX*T(1,5)
P(3,1)=0.
P(3,2)=0.
P(3,3)=1.
RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR MTRED
COMMON 0 VARIABLES 18 PROGRAM 164

PAGE 11 SR-VCMT

RELATIVE ENTRY POINT ADDRESS IS 0018 (HEX)

END OF COMPILATION

// FOR SR-FMFR - MATRIZ FRONTEIRA REDUZIDA - VCMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE FMFR(DM,F,IND)
  DIMENSION DM(8),F(5,5)
  DO 100 I=1,3
  DO 100 J=1,3
100  F(I,J)=0.
  F(1,1)=1.
  F(3,3)=1.
  F(2,1)=-DM(4)
  F(2,2)=-1.
  IF(IND)101,101,102
102  F(1,3)=DM(8)
  F(2,3)=DM(6)
101  RETURN
END
```

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR FMFR

COMMON 0 VARIABLES 8 PROGRAM 112

RELATIVE ENTRY POINT ADDRESS IS 0010 (HEX)

END OF COMPILATION

// FOR SR-PRMM - PRODUTO MATRIZ X MATRIZ - VCMT E VBEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE PRMM(A,B,C,N)
  DIMENSION A(5,5),B(5,5),C(5,5)
  DO 100 I=1,N
  DO 100 J=1,N
  SOMA=0.
  DO 110 K=1,N
110  SOMA=SOMA+A(I,K)*B(K,J)
100  C(I,J)=SOMA
  RETURN
END
```

FEATURES SUPPORTED

ONE WORD INTEGERS

PAGE 12 SR-VCMT

EXTENDED PRECISION

CORE REQUIREMENTS FOR PRMM
COMMON 0 VARIABLES 8 PROGRAM 100

RELATIVE ENTRY POINT ADDRESS IS 000C (HEX)

END OF COMPILATION

// FOR SR-PRMV - PRODUTO MATRIZ X VETOR - VCMT E VBEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE PRMV(A,VE,V,N)
      DIMENSION A(5,5),VE(5),V(5)
      DO 100 I=1,N
      SOMA=0.
      DO 110 J=1,N
110    SOMA=SOMA+A(I,J)*VE(J)
100    V(I)=SOMA
      RETURN
      END
```

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR PRMV
COMMON 0 VARIABLES 8 PROGRAM 82

RELATIVE ENTRY POINT ADDRESS IS 000C (HEX)

END OF COMPILATION

PAGE 1 PR-VBEMT

// JOB

PR-VBEMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVBEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(TYPEWRITER)

INTEGER RE,W

COMMON RE,W

WRITE(1,11)

11 FORMAT(/1X'LIGAR CHAVE 1 P/1442,DESLIGAR P/2501READER'//1X'LIGAR C

*HAVE 2 P/1132,DESLIGAR P/1403PRINTER'//1X'POSICIONAR CHAVES E DAR

*START')

PAUSE

RE=2

W=3

CALL DATSW(1,I)

CALL DATSW(2,J)

GOTO(12,13),I

13 RE=8

12 GOTO(16,17),J

17 W=5

16 CALL LINK(VBET1)

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

IOCS

CORE REQUIREMENTS FOR VBEMT

COMMON 2 VARIABLES 2 PROGRAM 146

END OF COMPILATION

PR-VBET1

```
LOG DRIVE      CART SPEC      CART AVAIL      PHY DRIVE
  0000         0005         0005         0000
```

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVBET1

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)
```

INTEGERS REVIEW

```

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),DI(2),DD(15),DT(6)

```

DIMENSION LRF(2),LRD(2),INP1(24),INP2(24),INP3(28),KOM(4)

COMMON RE, W, NBAR, LCAB, OBRA, FULA, RESP, LRE, LRD

```

COMMON /REVIEW/NDARS,ECAS,OBRAV,SEAR,REST,VERE,VERB
DEFINE FILE 1(200,24,U,LDT),2(201,25,U,LDF),3(201,06,U,LDD)

```

DEFINE FILE 4(201,07,U,LDR)

DATA INP17/'', 'EFT', 'S', 'T', '/', '/', 'J', 'O', '/', '/', 'F', 'O', '/', '/',

'R','I','/' , '' , '/' , '*' , '/' , '*' , '/'

DATA INP2/'/', '/', '/', '/', 'N', '*', 'L', 'R', 'E', '*', 'L', 'R', 'D', '*', '*',

*'L','Y','#','L','Z','Z','/' '#',' ','/'

DATA INP3 '/' '/' '/' '/' 'L' '/' '/' '/' 'I' 'Z' '/' '/' '/' 'E' '/' '/' 'C'.

'R','Y','','*','K','Y','*','K','Z','Z','/','*',' ',' '/

DATA IAE/0/,DT/6*0./

NBAR=0

DO 100 I=1.4

100 LCAB(I)=2

C-----INICIALIZAR LISTA RESTRICAO APOIOS EXTREMOS

C-----CODIGOS FIXO = 1 LIVRE = 0

DO 110 I=1.2

$$\text{LRF}(1) = 1$$
$$\text{LRD}(1) = 1$$

110 ERDT17-1
CALL CAR

```
CALL CABV(2)
1CAB(1)=1
```

DADOS DE IDENTIFICACAO

```
101 READ(RF,102)KOM
```

```
101 READ (RLE,102)
102 FORMAT(4A1)
```

```

102      FORMAT('4A17')
      WRITE(W*103)KOM

```

103 FORMAT(/1X,4A1)

100

```
DO 120 J=1,21,4
```

234

7 8 9 10 11

```
IF (ITEST(INP1,J,K,KOM,1))120,104,120
```

104 GOTO(101,115,125,135,145,113),I

120 CONTINUE

```

120 CONTINUE
105 WRITE(W*106)KOM

```

```
106 FORMAT(//20X'NAO EXISTE O CODIGO ('4A1')'///)
```

FORMAT: 77
CALL EXIT

```

115 READ(RE,107)OBRA
    WRITE(W,108)OBRA
107 FORMAT(20A4)
108 FORMAT(1X,20A4)
    LCAB(2)=1
    GOTO 101
125 READ(RE,107)FULA
    WRITE(W,108)FULA
    LCAB(3)=1
    GOTO 101
135 READ(RE,107)RESP
    WRITE(W,108)RESP
    LCAB(4)=1
    GOTO 101
145 LCAB(1)=2
    GOTO 101
127 WRITE(W,128)KOM,(DD(K),K=1,N)
128 FORMAT(1X,4A1,1X,15F5.0)
C-----LER DADOS DE DEFINICAO
113 READ(RE,114)KOM,DD
114 FORMAT(4A1,1X,15F5.0)
    I=0
    DO 140 J=1,21,4
        K=J+3
        I=I+1
        IF(ITEST(INP2,J,K,KOM,1))140,116,140
116 GOTO(165,205,215,225,225,195),I
140 CONTINUE
    GOTO 105
195 WRITE(W,103)KOM
    GOTO 245
165 CALL CONCA(DD,1,200,IER,2,N)
    IF(IER)121,121,198
121 NBAR=DD(1)
    NJ=NBAR+1
    WRITE(W,112)KOM,DD(1)
112 FORMAT(1X,4A1,1X,2F5.0)
    DO 130 I=1,NJ
130 WRITE(2,'I')IAE,DT
    GOTO 113
205 DO 150 I=1,2
150 LRE(I)=DD(I)
    WRITE(W,112)KOM,DD(1),DD(2)
    GOTO 113
215 DO 160 I=1,2
160 LRD(I)=DD(I)
    WRITE(W,112)KOM,DD(1),DD(2)
    GOTO 113
225 K=I-3
    CALL CONCA(DD,15,NJ,IER,4,N)
    IF(IER)123,123,198

```

PAGE 3 PR=VBET1

```
123 CALL VBLIB(DD,K,N)
    GOTO 127
129 WRITE(W,132)KOM,ID,DI,(DT(J),J=1,INC)
132 FORMAT(1X,4A1,A1,2F5.0,6F15.2)
    GOTO 245
131 WRITE(W,133)KOM,ID,DI,(DT(J),J=1,INC)
133 FORMAT(1X,4A1,A1,2F5.0,6E15.6)
C-----LER DADOS DE DEFINICAO COMPLEMENTAR
245 READ(RE,117)KOM,ID,DI,DT
117 FORMAT(4A1,A1,2F5.0,6F10.0)
    I=0
    DO 170 J=1,25,4
      K=J+3
      I=I+1
      IF(I*TEST(INP3,J,K,KOM,1))170,118,170
118 GOTO(255,255,255,255,265,265,275),I
170 CONTINUE
    GOTO 105
255 CALL CONCB(DI,ID,INC,IER,NBAR,3)
    IF(IER)124,124,199
124 CALL VCDB(DT,DI,I,INC)
    GOTO(129,131,131,131),I
265 K=I-2
    CALL CONCB(DI,ID,INC,IER,NJ,4)
    IF(IER)126,126,199
126 CALL VBDJ(DT,DI,K,INC)
    GOTO 131
275 WRITE(W,103) KOM
    CALL LINK(VBET2)
198 WRITE(W,196)KOM,DD,IER
196 FORMAT(/5X,4A1,15F5.0//5X'ERRO='15)
    CALL EXIT
199 WRITE(W,197)KOM,ID,DI,IER
197 FORMAT(/75X,4A1,A1,2F5.0//5X'ERRO='15)
    CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR VBET1
COMMON 192 VARIABLES 192 PROGRAM 852

END OF COMPILATION

PR-VBET2

```
LOG DRIVE      CART SPEC      CART AVAIL      PHY DRIVE
  0000         0005         0005         0000
```

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVBET2

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)
```

INTEGERS REW

```

DIMENSION DD(6),DI(2),KOM(4),LCAB(4),OBRA(20),FULA(20),RESP(20)

```

DIMENSION TITUL(20),MC(200),JC(201),INP4(36),LRE(2),LRD(2),D(3)

```

DIMENSION LEN(2),KAC(2)

```

COMMON RE, W, NBAR, LCAB, OBRA, FULA, RESP, LRE, LRD, LEN, KAC, MC, JC, TITUL

EQUIVALENCE (D(1),DI(1)),(DD(1),D(3))

```
DEFINE FILE 1(200,24,U,LDT),2(201,25,U,LDF),3(201,06,U,LDD):
```

DEFINE FILE 4(201,07,U,LDR)

DATA INP477/1,49/1,YU,15YR,1/1,1/1,1L,1Y,1*,1*,1D,1Y,1*,1D,

'Z', 'Z', '', '*', 'Č', 'Y', '*', 'Č', 'Z', 'Z', '/', 'È', 'X', 'È', '/', 'Č', 'Á

'R', 'Č', 'A', 'R' / KJ, KM / Ō, Ō /

C-----ZÉRAR CONTROLES DE BARRAS E JUNTAS CARREGADAS

NJ=NBAR+1

```
DO 200 I=1,NBAR
```

$$MC(1) = 0$$
200 $J_C(i) = 0$

JC(NJ)=0

C-----IMPRIMIR CABECALHO DE IDENTIFICACAO

CALL CARV(2)

C-----LER E IMPRIMIR TITULO DO CARREGAMENTO

READ(RE,201)TITUL

201 READ (RE, 201)
FORMAT (20A4)

WRITE(W,202)TITUL

```
202 WRITE(W,202)TITLE
      FORMAT(//1X'*LIST. DADOS* ',20A4/1X,80('=#'))
```

FORMAT 7
GOTO 203

```
223 WRITE(W,224)KOM,ID,DI,(DD(J),J=1,INC)
```

```
WRITE(NEWUNIT,*)Z2,Z4,KUM,I0,D1,(BBR3),J,S  
FORMAT(1X,4A1,A1,2F5.0,6F15.2)
```

C- - - - - LER DADOS DO CARREGAMENTO

READ(RE,204)KOM,IP,PI,PD

```
203 READ(NE,204)KUM,IB,DB,DBD  
204 FORMAT(4A1,A1,2F5.0,6F10.0)
```

FOR
I=0

DO 210 J=1,33,4

DC 21
K=J+3

$$I = I + 1$$

```
IF ITEST(INP4,J,K,KOM,1))210,206,210
```

206 GOTO(315,315,325,325,325,325,335,345,345),I

208 60707519
210 CONTINUE

CONTINUE
WRITE (W, 207) KOM

```

PAGE      2      PR=VBET2
207  FORMAT(/1X'NAO EXISTE O CODIGO  ('4A1')'////)
      CALL EXIT
315  CALL CONCB(DI,ID,INC,IER,NBAR,3)
      IF(IER)221,221,298
221  CALL VBCB(DD,ID,DI,I,KOM,INC)
      GOTO 203
325  K=I+2
      CALL CONCB(DI,ID,INC,IER,NJ,4)
      IF(IER)222,222,298
222  CALL VBCJ(DD,DI,K,INC)
      GOTO 223
345  CALL CONCA(D,3,NJ,IER,4,N)
      IF(IER)226,226,298
226  IF(N=2)227,228,229
227  IER=7
      GOTO 298
C-----INICIALIZAR CONTROLES DE CARGAS CONFORME /CAR  *CAR
228  N3=1
      GOTO 236
229  N3=D(3)
236  N1=D(1)
      N2=D(2)
      IF(I=8)231,231,232
231  IF(KM)241,241,242
241  DO 220 J=1,NBAR
220  MC(J)=0
      KM=1
242  DO 230 J=N1,N2,N3
230  MC(J)=1
      GOTO 233
232  IF(KJ)243,243,244
243  DO 240 J=1,NJ
240  JC(J)=0
      KJ=1
244  DO 250 J=N1,N2,N3
250  JC(J)=1
233  WRITE(W,234)KOM,ID,(D(K),K=1,N)
234  FORMAT(1X,4A1,A1,3F5.0)
      GOTO 203
335  WRITE(W,211)KOM
211  FORMAT(/1X,4A1)
      CALL LINK(VBET3)
298  WRITE(W,299)KOM,ID,DI,IER
299  FORMAT(/5X,4A1,A1,2F5.0//5X'ERRO='I5)
      CALL EXIT
      END

```

```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

```

PAGE 1 PR-VBET3

// JOB

PR-VBET3

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVBET3

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION LRE(2),LRD(2),DMF(8),DMT(8),T(5,5),F(5,5),P(5,5),VE(5)

DIMENSION V(5),VR(2),LCAB(4),OBRA(20),FULA(20),RESP(20),TITUL(20)

DIMENSION LEN(2),MC(200),JC(201),KAC(2),LR(2)

COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC,TITUL

DEFINE FILE 1(200,24,U,LDT),2(201,25,U,LDF),3(201,06,U,LDD)

DEFINE FILE 4(201,07,U,LDR)

DATA VE/4*0.,1./,LDT,LDF,LDD,LDR/1,1,1,1/,IAE/0/

NJ=NBAR+1

C-----PRIMEIRA PASSAGEM PELA ESTRUTURA

IND=JC(1)

READ(2,LDF)IAE,DMF

CALL FMFVB(DMF,F,IND)

DO 500 I=1,2

K=I+2

F(K,K)=1.

IF(LRE(I))500,500,452

452 F(K,5)=0.

500 CONTINUE

401 IND=MC(LDT)

READ(1,LDT)DMT

DO 520 I=1,3

IF(DMT(I))491,491,520

520 CONTINUE

IF(DMT(4))402,402,403

402 CALL FMT(DMT,T,IND)

GOTO 404

403 CALL FMTVB(DMT,T,IND)

404 CALL PRMM(T,F,P,5)

IND=JC(LDF)

READ(2,LDF)IAE,DMF

CALL FMFVB(DMF,T,IND)

IF(LDT=NBAR)451,451,406

451 CALL PRMM(T,P,F,5)

GOTO 401

406 DO 510 I=1,2

K=I+2

T(K,K)=1.

```

507 IF(LRD(1))507,507,453
    T(K,5)=-T(K,5)
    T(K,1)=-T(K,1)
    GOTO 510
453 T(K,5)=0.
    T(1,5)=-T(1,5)
510 CONTINUE
    CALL PRMM(T,P,F,5)
C-----CONDICOES DE CONTORNO
    CALL CAEBV(F,T,VE)
C-----RESOLUCAO DO SISTEMA RESULTANTE
    CALL VBINV(T)
    CALL PRMV(T,VE,V,2)
C-----REARRANJAR VETOR DE ESTADO DA JUNTA 1
    DO 410 I=1,4
410  VE(I)=0.
    DO 430 I=1,2
    KA=KAC(I)
430  VE(KA)=-V(I)
C-----DESLOCAMENTOS E REACOES (APOIOS ELASTICOS ) JUNTA 1
    LDF=1
    LDT=1
    IND=JC(LDF)
    READ(2,LDF)IAE,DMF
    DO 440 I=1,2
440  LR(I)=LRE(I)
    CALL AEVB(DMF,IND,VE,K,VR,LR)
    IF(IAE+K)409,409,411
411  LD=LDF-1
    WRITE(4,LDR)LD,VR
C-----IMPRIMIR TITULO DO CARREGAMENTO
409  WRITE(W,405)TITUL
405  FORMAT(/1X'*RESULTADOS* ',20A4/1X,80('='))
    WRITE(W,415)
415  FORMAT(/1X'ACOES NAS EXTREMIDADES DAS BARRAS'/)
    WRITE(W,425)
425  FORMAT(1X'BARRA'2(3X'CORTANTE'4X'M.FLETOR'))
C-----SEGUNDA PASSAGEM PELA ESTRUTURA - RESULTADOS
421  IND=MC(LDT)
    WRITE(3,LDD)VE(1),VE(2)
    READ(1,LDT)DMT
    IF(DMT(4))412,412,413
412  CALL FMT(DMT,T,IND)
    GOTO 414
413  CALL FMTVB(DMT,T,IND)
414  CALL PRMV(T,VE,V,5)
    LD=LDT-1
C-----ACOES DE EXTREMO DE BARRAS
    WRITE(W,435)LD,VE(3),VE(4),V(3),V(4)
435  FORMAT(1X,I5,2(F11.2,1X,F11.2))
    IF(LDT-NBAR)416,416,417

```

PAGE 3 PR=VBET3

```
416 IND=JC(LDF)
    READ(2,LDF)IAE,DMF
    CALL FMFVB(DMF,F,IND)
    CALL PRMV(F,V,VE,5)
C-----REACOES (APOIOS ELASTICOS) JUNTAS INTERNAS
    IF(IAE)421,421,422
422 VR(1)=-DMF(3)*VE(1)
    VR(2)=-DMF(4)*VE(2)
    LD=LDF-1
    WRITE(4,LDR)LD,VR
    GOTO 421
417 IND=JC(LDF)
    READ(2,LDF)IAE,DMF
C-----REACOES E DESLOCAMENTOS JUNTA NJ
    DO 450 I=1,2
450 LR(I)=LRD(I)
    CALL AEVB(DMF,IND,V,K,VR,LR)
    WRITE(3,LDD)V(1),V(2)
    IF(IAE+K)424,424,426
426 LD=LDF-1
    WRITE(4,LDR)LD,VR
424 WRITE(W,445)
445 FORMAT(/,1X'DESLOCAMENTOS GENERALIZADOS'/)
    WRITE(W,455)
455 FORMAT(1X'JUNTA'4X'DESLOC Y'4X'DESLOC Z')
    DO 460 I=1,NJ
    READ(3,I)VR
    WRITE(W,465)I,VR
465 FORMAT(1X,I5,2F12.4)
460 CONTINUE
    LD=LDR-1
    IF(LD)427,427,428
428 WRITE(W,475)
475 FORMAT(/,1X'REACOES'/)
    WRITE(W,485)
485 FORMAT(1X'JUNTA'4X'REACAO Y'4X'REACAO Z')
    DO 470 I=1,LD
    READ(4,I)IND,VR
    WRITE(W,495)IND,VR
495 FORMAT(1X,I5,2F12.2)
470 CONTINUE
427 CALL LINK(VBET4)
491 LD=LDT-1
    IER=6
    WRITE(W,492)LD,IER
492 FORMAT(/,5X'BARRA='I5,3X'ERRO='I5)
    CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS

PAGE 1 PR-VBET4

// JOB.

PR-VBET4

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEVBET4

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION KOM(4),LCAB(4),OBRA(20),FULA(20),RESP(20),INP5(12)

DIMENSION LRE(2),LRD(2)

COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,LRE,LRD

DEFINE FILE 1(200,24,U,LDT),2(201,25,U,LDF),3(201,06,U,LDD)

DEFINE FILE 4(201,07,U,LDR)

DATA INP5/'/','*',' ',' ',' ',' ','/','E','S','T','/','F','I','M'/'

C-----LER CODIGO DE DECISAO

READ(RE,505)KOM

505 FORMAT(4A1)

I=0

DO 480 J=1,9,4

K=J+3

I=I+1

IF(I TEST(INP5,J,K,KOM,1))480,431,480

431 GOTO(432,433,434),I

480 CONTINUE

C-----FINALIZAR PROCESSAMENTO

WRITE(W,493)KOM

493 FORMAT(/5X'CODIGO ('4A1') NAO IDENTIFICADO')

GOTO 494

434 WRITE(W,436)KOM

436 FORMAT(/1X,4A1)

494 CALL CABV(2)

WRITE(W,437)

437 FORMAT(/1X'PROCESSAMENTO CONCLUIDO'//1X'HR. MAQUINA INICIAL='//1X

'HR. MAQUINA FINAL ='///1X'TEMPO PROCESSAMENTO= HRS. MIN.*

**'/)

WRITE(W,438)

438 FORMAT('1')

CALL EXIT

C-----PROCESSAR NOVO CARREGAMENTO

432 CALL LINK(VBET2)

C-----PROCESSAR NOVA ESTRUTURA

433 CALL LINK(VBET1)

END

FEATURES SUPPORTED

PAGE 1 SR-VBEMT

// JOB

SR-VBEMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR SR-VBLIB - LIBERACOES GENERALIZADAS NAS JUNTAS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE VBLIB(DD,IND,N)
DIMENSION DD(15),DMF(8)

DO 100 I=1,N

J=DD(I)

READ(2,J)IAE,DMF

DMF(IND)=1.E+15

WRITE(2,J)IAE,DMF

100 CONTINUE

RETURN

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR VBLIB

COMMON 0 VARIABLES 28 PROGRAM 66

RELATIVE ENTRY POINT ADDRESS IS 0021 (HEX)

END OF COMPILATION

// FOR SR-VBDJ - DADOS DAS JUNTAS - ARQUIVO

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE VBDJ(DT,DI,IND,INC)

DIMENSION DT(6),DI(2),DMF(8)

N1=DI(1)

N2=DI(2)

K=0

DO 100 I=N1,N2

K=1+K*INC

READ(2,I)IAE,DMF

DMF(IND)=DT(K)

IF(IND-3)101,102,102

102 IAE=1

101 WRITE(2,I)IAE,DMF

100 CONTINUE

INC=K

RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VBDJ
COMMON 0 VARIABLES 32 PROGRAM 108

RELATIVE ENTRY POINT ADDRESS IS 0024 (HEX)

END OF COMPILATION

// FOR SR=VBCB - CARGAS NAS BARRAS - ROTINA PRINCIPAL

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```

SUBROUTINE VBCB(DD,DI,DI,INDS,KOM,INC)
  INTEGER RE,W
  DIMENSION DD(6),DI(2),KOM(4),DMT(8),LCAB(4),OBRA(20),FULA(20)
  DIMENSION RESP(20),MC(200),LRE(2),LRD(2),LEN(2),KAC(2)
  COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP, LRE,LRD,LEN,KAC,MC
  N1=DI(1)
  N2=DI(2)
  K=0
  IF((N2-N1-2)*INC)101,101,102
102 IF(INDS-1)101,101,103
101 DO 100 I=N1,N2
  K=1+K*INC
  READ(1,I)DMT
  IF(MC(I))108,108,109
108 DO 110 J=5,8
110 DMT(J)=0.
  MC(I)=1
109 GOTO(115,125),INDS
115 IF(DMT(4))111,111,112
C-----CARGA UNIFORME - FLEXAO LIVRE
111 CALL VCUY(DMT,DD,K)
  GOTO 113
C-----CARGA UNIFORME - BARRA S/ BASE ELASTICA
112 CALL VBUY(DMT,DD,K)
  GOTO 113
125 IF(DMT(4))116,116,117
C-----CARGA LINEAR - FLEXAO LIVRE
116 CALL VCLY(DMT,DD,K)
  GOTO 113
C-----CARGA LINEAR - BARRA S/ BASE ELASTICA
117 CALL VBLY(DMT,DD,K)
113 WRITE(1,I)DMT
100 CONTINUE

```


PAGE 3 SR-VBEMT

```
121 WRITE(W,121)KOM,ID,DI,(DD(J),J=1,K)
    FORMAT(1X,4A1,A1,2F5.0,6F15.2)
    RETURN
103 IER=5
    WRITE(W,104)KOM,ID,DI,IER
104 FORMAT(//5X,4A1,A1,2F5.0//5X'ERRO='I5)
    CALL EXIT
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VBCB
COMMON 396 VARIABLES 32 PROGRAM 270

RELATIVE ENTRY POINT ADDRESS IS 0043 (HEX)

END OF COMPILATION

// FOR SR-VBUY - CARGA UNIFORME - BASE ELASTICA

*ONE WORD INTEGERS
*EXTENDED PRECISION

```
*LIST SOURCE PROGRAM
    SUBROUTINE VBUY (DMT,DD,K)
    DIMENSION DMT(8),DD(6)
    GAM2=SQRT(DMT(4)/DMT(3)/DMT(2)/4.)
    GAM=SQRT(GAM2)
    GL=GAM*DMT(1)
    SN=SIN(GL)
    CS=COS(GL)
    AUX=EXP(GL)
    GL=DD(K)/DMT(4)
    SH=(AUX-1./AUX)/2.
    CH=(AUX+1./AUX)/2.
    AUX=SH*CS-CH*SN
    TEP=SH*CS+CH*SN
    DMT(5)=DMT(5)+(1.-CH*CS)*GL
    DMT(6)=DMT(6)-AUX*GL*GAM
    GL=GL*GAM2*DMT(2)*DMT(3)
    DMT(7)=DMT(7)-2.*TEP*GL*GAM
    DMT(8)=DMT(8)+2.*SH*SN*GL
    RETURN
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VBUY
COMMON 0 VARIABLES 40 PROGRAM 212

PAGE 4 SR-VBEMT

RELATIVE ENTRY POINT ADDRESS IS 0031 (HEX)

END OF COMPILATION

// DUP

*DELETE VBUY
D 26 NAME NOT FOUND IN LET/FLET

*STORE WS UA VBUY
CART ID 0005 DB ADDR 2840 DB CNT 0011

// FOR SR-VBLY = CARGA LINEAR - BASE ELASTICA

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM
SUBROUTINE VBLY (DMT,DD,K)
DIMENSION DMT(8),DD(6)
GAM = SQRT(DMT(4)/DMT(3)/DMT(2)/4.)
GAM = SQRT(GAM)
GEIZ = GAM*DMT(2)*DMT(3)
AUX = GAM*DMT(1)
SN = SIN(AUX)
CS = COS(AUX)
AUX = EXP(AUX)
SH = (AUX - 1./AUX)/2.
CH = (AUX + 1./AUX)/2.
A = CH*CS
B = SH*SN
C = SH*CS - CH*SN
D = SH*CS + CH*SN
QA = DD(K)/DMT(4)
K = K + 1
QB = DD(K)/DMT(4)
DMT(5) = DMT(5) + (QA - QB)*(D/DMT(1)/GAM/2.) + QB - A*QA
DMT(6) = DMT(6) + (A - 1.)*(QA - QB)/DMT(1) - C*GAM*QA
DMT(7) = DMT(7) - ((QB - QA)*B/DMT(1) + D*GAM*QA)*GAM*GEIZ*2.
DMT(8) = DMT(8) + ((QA - QB)*C/DMT(1) + B*GAM*QA*2.)*GEIZ
RETURN
END

FEATURES SUPPORTED

ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR VBLY

COMMON 0 VARIABLES 54 PROGRAM 326

RELATIVE ENTRY POINT ADDRESS IS 0040 (HEX)

PAGE 5 SR-VBEMT

END OF COMPILE

// FOR SR-VBCJ - DESLOCAMENTOS IMPOSTOS E CARGAS NAS JUNTAS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE VBCJ(DD,DI,NV,INC)
      INTEGER RE,W
      DIMENSION DD(6),DI(2),DMF(8),LCAB(4),OBRA(20),FULA(20)
      DIMENSION RESP(20),MC(200),JC(201),LRE(2),LRD(2),LEN(2),KAC(2)
      COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC
      N1=DI(1)
      N2=DI(2)
      K=0
      DO 100 I=N1,N2
      K=1+K*INC
      READ(2,I)IAE,DMF
      IF(JC(I))109,109,111
109   DO 110 J=5,8
110   DMF(J)=0.
      JC(I)=1
111   DMF(NV)=DMF(NV)+DD(K)
      WRITE(2,I)IAE,DMF
100   CONTINUE
      INC=K
      RETURN
      END
```

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR VBCJ

COMMON 596 VARIABLES 32 PROGRAM 146

RELATIVE ENTRY POINT ADDRESS IS 0028 (HEX)

END OF COMPILE

// FOR SR-FMFVB - MATRIZ FRONTEIRA

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE FMFVB(DMF,F,IND)
      DIMENSION DMF(8),F(5,5)
      DO 100 I=1,5
      DO 110 J=1,5
110   F(I,J)=0.
100   F(1,1)=1.
      F(3,3)=-1.
      F(4,4)=-1.
```

PAGE 6 SR-VBEMT

```
      F(1,3)=DMF(1)
      F(2,4)=DMF(2)
      F(3,1)=-DMF(3)
      F(4,2)=-DMF(4)
      IF(IND) 101,101,102
102    DO 120 I=1,4
      K=I+4
120    F(1,5)=DMF(K)
101    RETURN
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR FMFVB
COMMON 0 VARIABLES 12 PROGRAM 166

RELATIVE ENTRY POINT ADDRESS IS 0015 (HEX)

END OF COMPILATION

// FOR SR-FMTVB - MATRIZ TRANSFERENCIA - BASE ELASTICA

*ONE WORD INTEGERS
*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE FMTVB(DMT,T,IND)
      DIMENSION DMT(8),T(5,5)
      DO 100 I=1,4
100    T(5,I)=0.
      T(I,5)=0.
      T(5,5)=1.
      GAM2=SQRT(DMT(4)/DMT(3)/DMT(2)/4.)
      GAM=SQRT(GAM2)
      GEI2=DMT(2)*DMT(3)*GAM
      GL=GAM*DMT(1)
      SN=SIN(GL)
      CS=COS(GL)
      AUX=EXP(GL)
      SH=(AUX-1./AUX)/2.
      CH=(AUX+1./AUX)/2.
      B=SH*SN
      C=SH*CS-CH*SN
      D=SH*CS+CH*SN
      T(1,1)=CH*CS
      T(2,2)=T(1,1)
      T(3,3)=-T(1,1)
      T(4,4)=T(3,3)
      T(1,2)=D/GAM/2.
      T(4,3)=T(1,2)
      T(1,3)=-C/GEI2/GAM2/4.
```

PAGE 7 SR=VBEMT

```
T(1,4)=-B/GEIZ/GAM/2.
T(2,3)=-T(1,4)
T(2,1)=C*GAM
T(3,4)=T(2,1)
T(2,4)=-D/GEIZ/2.
T(3,1)=2.*D*GEIZ*GAM2
T(3,2)=2.*B*GEIZ*GAM
T(4,1)=-T(3,2)
T(4,2)=C*GEIZ
IF(IND)108,108,109
C-----VETOR DE CARGA
109 DO 110 I=1,4
110 T(I,5)=DMT(I+4)
108 RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR FMTVB
COMMON 0 VARIABLES 56 PROGRAM 396

RELATIVE ENTRY POINT ADDRESS IS 0046 (HEX)

END OF COMPILATION

// FOR SR=CAEVB - CONDIC0ES DE CONTORNO

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE CAEVB(F,A,VE)
INTEGER RE,W
DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20),LRE(2),LRD(2)
DIMENSION F(5,5),A(5,5),VE(5),LEN(2),KAC(2)
COMMON RE,W,NBAR,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC
DO 100 I=1,2
LEN(I)=I
KAC(I)=I
IF(LRE(I))101,101,102
102 KAC(I)=I+2
101 IF(LRD(I))103,103,100
103 LEN(I)=I+2
100 CONTINUE
DO 110 I=1,2
LE=LEN(I)
VE(I)=F(LE,5)
DO 120 J=1,2
KA=KAC(J)
120 A(I,J)=F(LE,KA)
110 CONTINUE
```

PAGE 8 SR-VBEMT

RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR CAEBV
COMMON 196 VARIABLES 8 PROGRAM 160

RELATIVE ENTRY POINT ADDRESS IS 000A (HEX)

END OF COMPILATION

// FOR SR-AEBV = REACOES NOS APOIOS EXTREMOS

*ONE WORD INTEGERS
*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE AEBV(DMF,IND,V,IR,VR,LR)
      DIMENSION DMF(8),V(5),VR(2),LR(2)
      IR=0
      IF(IND)101,101,102
101    DO 110 I=5,8
110    DMF(I)=0.
102    DO 120 I=1,2
      J=I+2
      K=I+6
      IF(LR(I))103,103,104
104    VR(I)=V(J)-DMF(K)
      V(I)=DMF(I+4)
      IR=1
      GOTO 120
103    VR(I)=-DMF(I+2)*V(I)
      V(J)=VR(I)+DMF(K)
120    CONTINUE
      RETURN
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR AEBV
COMMON 0 VARIABLES 8 PROGRAM 186

RELATIVE ENTRY POINT ADDRESS IS 0011 (HEX)

END OF COMPILATION

PAGE 1 PR-PPMT

// JOR

PR-PPMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPPMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(TYPEWRITER)

INTEGER RE,W

COMMON RE,W

WRITE(1,11)

11 FORMAT(/1X'LIGAR CHAVE 1 P/1442 DESLIGAR P/2501READER'/1X'LIGAR
*CHAVE 2 P/1132, DESLIGAR P/1403PRINTER'/1X'POSICIONAR CHAVES E DAR

* START')

PAUSE 1111

RE=2

W=3

CALL DATSW(1,I)

CALL DATSW(2,J)

GOTO(12,13),I

13 RE=8

12 GOTO(16,17),J

17 W=5

16 CALL LINK(PPMT1)

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

IOCS

CORE REQUIREMENTS FOR PPMT

COMMON 2 VARIABLES

2 PROGRAM

146

END OF COMPILATION

PR-PPMT1

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPPMT1

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

* IOCS (CARD, 2501 READER, 1132 PRINTER, 1403 PRINTER, DISK)

INTEGER REW

```

DIMENSION OBRA(20),FULA(20),RESP(20),LCAB(4),DI(2),DD(15),DT(6)

```

DIMENSION KOM(4), LRE(3), LRD(3), INP1(24), INP2(36), INP3(52), MF(3)

```

DIMENSION DMT(10),MT(3)

```

COMMON RE, W, NBAR, NRAM, LCAB, OBRA, FULA, RESP, LRE, LRD

```
DEFINE FILE 1(200,42,U,LDT),2(201,45,U,LDF),3(201,9,U,LDD)
```

```
DEFINE FILE 4(201,10,U,LDR)
```

DATA INP1/'/', 'E', 'S', 'T', '/', '/', 'J', 'O', '/', '/', 'F', 'O', '/', '/',

'R', 'T', 'I', '*', '/', '*', '/', '*', '/', '*', '/',

DATA INP2/'/', '/', '/', 'N', '/', '/', 'N', 'R', '#', 'L', 'R', 'E', '*', 'L',

'R','D','','R','A','M','*','*','L','X','*','*','L','Y','*','L','Z

', 'Z', ' / ', ' ', ' ', ' ' /

DATA INP3/'*', '*', '*', 'X', '*', '*', '*', 'Y', '*', '*', 'K', 'X', '*', '*',

*'K','Y','I','K','Z','Z','/','/','A','X','/','/','I','Z','/','/','/'

* 'Ě', ' / ', 'Č', 'Ř', 'Ý', ' / ', 'Í', 'Ď', 'P', ' / ', 'F', 'I', 'M', ' / ', 'E', 'S',

* T / *

153 NBAR=0

$$NBAR=0$$

NRAM=0

$NJ=0$

DO 100 I=1,4

```
100 LCAB(1)=2
```

```
LCAB(1)=2
```

C-----INICIALIZAR LISTA RESTRICAO APOIOS EXTREMOS

C-----CODIGOS FIXO = 1 LIVRE = 0

DO_110 I=1,3

$$LRE(I) \approx 1$$
$$LRD(i) = i$$
$$DT(I) = 0.$$
$$DT(I+3) = 0.$$

110 MF(I)=0

CALL CABP(1)

```
LCAB(1)=1
```

C-----LÊR DADOS DE IDENTIFICACAO

```
101 READ(RE,102)KOM
```

WRITE(W,103)KOM

102 FÖRMÄT 74A17

```
103  FORMAT( /1X,4A1)
```

$$I = 0$$

```
DO 120 J=1,21,4
```



```

      K=J+3
      I=I+1
      IF(ITEST(INP1,J,K,KOM,1))120,104,120
104  GOTO(101,115,125,135,145,113),I
120  CONTINUE
105  WRITE(W,106)KOM
106  FORMAT(720X'NAO EXISTE O CODIGO ('4A1')'///)
      CALL EXIT
115  READ(RE,107)OBRA
      WRITE(W,108)OBRA
107  FORMAT(20A4)
108  FORMAT(1X,20A4)
      LCAB(2)=1
      GOTO 101
125  READ(RE,107)FULA
      WRITE(W,108)FULA
      LCAB(3)=1
      GOTO 101
135  READ(RE,107)RESP
      WRITE(W,108)RESP
      LCAB(4)=1
      GOTO 101
145  LCAB(1)=2
      GOTO 101
191  N=3
127  WRITE(W,128)KOM,ID,(DD(K),K=1,N)
128  FORMAT(1X,5A1,15F5.0)
C-----LER DADOS DE DEFINICAO
113  READ(RE,114)KOM,ID,DD
114  FORMAT(5A1,15F5.0)
      I=0
      DO 140 J=1,33,4
      K=J+3
      I=I+1
      IF(ITEST(INP2,J,K,KOM,1))140,116,140
116  GOTO(165,175,185,195,205,215,215,215,225),I
140  CONTINUE
      GOTO 105
225  WRITE(W,103)KOM
      GOTO 235
165  CALL CONCA(DD,1,200,IER,2,N)
      IF(IER)121,121,198
121  NBAR=DD(1)
      NB=NBAR
      NJ=NBAR+1
      WRITE(W,112)KOM,DD(1)
112  FORMAT(1X,4A1,1X,2F5.0)
      DO 130 I=1,NJ
130  WRITE(2'I)MF,DT
      DO 150 I=1,NBAR
150  WRITE(1'I)MF,DT

```

```

175 GOTO 113
CALL CONCA(DD,1,200,IER,2,N)
IF( IER)122,122,198
122 NRAM=DD(1)
NB=NB+NRAM
NJ=NJ+NRAM
WRITE(W,112)KOM,DD(1)
DO 160 I=NBAR,NB
160 WRITE(1,I)MF,DT
DO 170 I=NBAR,NJ
170 WRITE(2,I)MF,DT
GOTO 113
185 DO 180 I=1,3
180 LRE(I)=DD(I)
GOTO 191
195 DO 190 I=1,3
190 LRD(I)=DD(I)
GOTO 191
205 CALL CONCA(DD,15,NBAR,IER,4,N)
IF( IER)123,123,198
123 CALL PEJR(DD,ID,N)
GOTO 127
215 CALL CONCA(DD,15,NBAR,IER,4,N)
IF( IER)124,124,198
124 CALL PPJL(DD,I,N)
GOTO 127
129 WRITE(W,132)KOM,ID,DI,(DT(J),J=1,INC)
GOTO 235
131 WRITE(W,133)KOM,ID,DI,(DT(J),J=1,INC)
C-----LER DADOS DE DEFINICAO COMPLEMENTAR
235 READ(RE,117)KOM,ID,DI,DT
117 FORMAT(5A1,2F5.0,6F10.0)
132 FORMAT(1X,5A1,2F5.0,6F15.3)
133 FORMAT(1X,5A1,2F5.0,6E15.6)
I=0
DO 200 J=1,49,4
K=J+3
I=I+1
IF( ITEST( INP3,J,K,KOM,1) )200,118,200
118 IF( I-5)134,134,136
134 CALL CONCB(DI,ID,INC,IER,NJ,4)
IF( IER)137,137,199
137 CALL PPDJ(DT,DI,I,INC)
IF( I-2)129,129,131
136 IF( I-9)144,144,146
144 CALL CONCB(DI,ID,INC,IER,NB,3)
IF( IER)138,138,199
138 CALL PPOB(DT,DI,I,INC)
GOTO 131
146 N=I-9
WRITE(W,103)KOM

```

C-----CALCULAR COMPRIMENTOS E COS. DIRETORES

```

      IRAM=0
      NUMB=NBAR
      READ(2,1)MF,X,Y
      DO 210 I=1,NBAR
      K=I+1
      IF (IRAM) 141,142,141
141    IND=1
      NUMB=NUMB+1
      LD=NUMB+1
      READ(1,NUMB)MT,DMT
      READ(2,LD)MF,XJ,YJ
      LD=NUMB
      GOTO 143
142    IND=0
      XJ=X
      YJ=Y
      READ(1,I)MT,DMT
      READ(2,K)MF,X,Y
      LD=I
143    CX=X-XJ
      CY=Y-YJ
      IRAM=MF(1)
      XL=SQRT(CX*CX+CY*CY)
      CX=CX/XL
      CY=CY/XL
      WRITE(1,LD)MT,DMT,XL,CX,CY
      IF (IND) 210,210,142
210    CONTINUE
      GOTO(147,148,153,154),N
147    WRITE(W,156)
156    FORMAT(/1X'BARRA'2X'COMPR.'6X'CX'6X'CY'/)
C-----IMPRIMIR COMPRIMENTOS E COS. DIRETORES SE SOLICITADO /IMP
      DO 230 K=1,NB
      READ(1,K)MT,DMT,XL,CX,CY
230    WRITE(W,157)K,XL,CX,CY
157    FORMAT(1X,I5,3F8.3)
      GOTO 235
148    CALL EXIT
154    CALL LINK(PPMT2)
200    CONTINUE
      GOTO 105
198    WRITE(W,196)KOM,DD,IER
196    FORMAT(/5X,4A1,15F5.0//5X'ERRO='I5)
      CALL EXIT
199    WRITE(W,197)KOM,ID,DI,IER
197    FORMAT(/5X,5A1,2F5.0//5X'ERRO='I5)
      CALL EXIT
      END

```

PR-PPMT2

```
LOG DRIVE      CART SPEC      CART AVAIL      PHY DRIVE
  0000         0005         0005         0000
```

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPPMT2

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)
```

INTEGER REVIEW

```

DIMENSION OBRA(20),FULA(20),RESP(20),TITUL(20),LCAB(4),KOM(4),D(3)

```

DIMENSION DT(6),DI(2),LRE(3),LRD(3),LEN(3),KAC(3),INP4(68)

DIMENSION MC(200), JC(201)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC

COMMON TITUL

EQUIVALENCE (D(1),DI(1)),(DT(1),D(3))

```

DEFINE FILE 1(200,42,U,LDT),2(201,45,U,LDF),3(201,9,U,LDD)

```

DEFINITION FILE 4 (201, 10, U, LDR)

```

DATA INP42LXZS1UUCXDR//UYY//UY//LX//T/U
**DALY**/*/*/*CUX**C//ZDZ/Y*/CZ*Z*/*TCU
**X**/*/*/*DYX**C*ZDZ/Y*/CZ*Z*/*TCU
**//E*X'E//

```

$$NJ = NBAR + 1$$
$$NB = NBAR + NRAM$$
$$N_{JT} = N_J + N_{RAM}$$
 $K_M = 0$

KJ=00

C-----RERAR CONTROLES DE BARRAS E JUNTAS CARREGADAS

```
DO 300 I=1,NB
```

$$MC(I) = 0$$

300 JC(1)=0

$$JC(NJT) = 0$$

C-----IMPRIMIR CABECALHO DE IDENTIFICACAO

CALL CABP(1)

C-----LER E IMPRIMIR TITULO DO CARREGAMENTO

READ(RE,301)TITUL

WRITE (W, 302) TITUL

301 WRITE(W,502)
FORMAT(20A4)

```
301 FORMAT(20A4)  
302 FORMAT(7/1X'*LIST, DADOS* ',20A4/1X,80('=''))
```

FORMAT 17
GOTO 303

```

323      WRITE(W,324)KOM,ID,DI,(DT(J),J=1,INC)

```

323 WRITETW, 324) KUM, TD, DI, (DI
C-----LER DADOS DO CARREGAMENTO

303 READ (RF,304)KOM,LD,DI,DT

303 READ (TRE,304) RUM,10,DI,DI
304 FORMAT (5A1,2F5.0,6F10.0)

```

304  FORMAT(5A1,2F5.0,8F10.0)
324  FORMAT(1X,5A1,2F5.0,6F10.2)

```

FOR
112

PAGE 2 PR-PPMT2

```
DO 320 J=1,65,4
K=J+3
I=I+1
IF(1TEST(INP4,J,K,KOM,1))320,306,320
306 IF(I-14)307,307,309
307 CALL CONCB(DI,ID,INC,IER,NJT,3)
IF(1IER)308,308,309
308 IF(I-8)311,311,312
311 CALL PPBC(DT,ID,DI,I,KOM,INC)
GOTO 303
312 CALL PPJC(DT,ID,DI,I,INC)
GOTO 323
309 IF(I-16)313,313,314
313 CALL CONCA(D,3,NJT,IER,4,N)
IF(1IER)316,316,398
316 IF(N-2)317,318,319
317 IER=7
GOTO 398
C-----INICIALIZAR CONTROLES DE CARGAS CONFORME /CAR *CAR
318 N3=1
GOTO 321
319 N3=D(3)
321 N1=D(1)
N2=D(2)
IF(I-16)325,325,335
325 IF(KM)322,322,326
322 DO 340 J=1,NB
340 MC(J)=0
KM=1
326 DO 350 J=N1,N2,N3
350 MC(J)=1
GOTO 393
335 IF(KJ)327,327,328
327 DO 360 J=1,NJT
360 JC(J)=0
KJ=1
328 DO 370 J=N1,N2,N3
370 JC(J)=1
393 WRITE(W,329)KOM,ID,(D(K),K=1,N)
329 FORMAT(1X,5A1,3F5.0)
GOTO 303
314 WRITE(W,324)KOM
CALL LINK(PPMT3)
320 CONTINUE
WRITE(W,331)KOM
331 FORMAT(/1X'NAO EXISTE O CODIGO ('4A1')'///)
CALL EXIT
398 WRITE(W,399)KOM,ID,DI,IER
399 FORMAT(/5X,5A1,2F5.0//5X'ERRO='I5)
CALL EXIT
END
```

PAGE 1 PR-PPMT3

// JOB

PR-PPMT3

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPPMT3

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(DISK)

INTEGER RE,W

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),LRE(3),LRD(3),VE(7)

DIMENSION TITUL(20),LEN(3),KAC(3),MF(3),MT(3),V(7),LR(3),VR(3)

DIMENSION F(7,7),T(7,7),P(7,7),DMT(10),DMF(14),MC(200),JC(201)

DIMENSION SM(4),VAUX(7)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC

COMMON TITUL,LDR,VAUX

DEFINE FILE 1(200,42,U,LDT),2(201,45,U,LDF),3(201,9,U,LDD)

DEFINE FILE 4(201,10,U,LDR)

DATA VE/6*0.,1./,LDD,LDT,LDF/1,1,1/

LDR=1

VAUX(7)=1.

C-----PRIMEIRA PASSAGEM PELA ESTRUTURA

NUMB=NBAR

IND=JC(LDF)

READ(2,LDF)MF,DMF

CALL PPMF(DMF,MF,F,IND)

DO 400 I=1,3

K=I+3

F(K,K)=1.

IF(LRE(I))400,400,401

401 F(K,7)=0.

400 CONTINUE

405 IND=MC(LDT)

READ(1,LDT)MT,DMT,XL,CX,CY

CALL PPML(DMT,MT,T,XL,IND)

CALL PPMG(T,IND,CX,CY)

CALL PMMPP(T,F,P)

IND=JC(LDF)

READ(2,LDF)MF,DMF

CALL PPMF(DMF,MF,T,IND)

IF(MF(1))406,406,407

407 NUMB=NUMB+1

IND=MC(NUMB)

LD=LDT

READ(1,NUMB)MT,DMT,XL,CX,CY

LDT=LD

CALL PPBR(DMT,T,IND,XL,CX,CY,SM)

PAGE 2 PR-PPMT3

```
406 IF(LDT-NBAR)408,408,409
408 CALL PMMPP(T,P,F)
      GOTO 405
409 DO 410 I=1,3
      K=I+3
      T(K,K)=1.
      IF(LRD(I))402,402,411
402 T(K,7)=-T(K,7)
      T(K,I)=-T(K,I)
      GOTO 410
411 T(K,7)=0.
      T(I,7)=-T(I,7)
410 CONTINUE
      CALL PMMPP(T,P,F)
C-----CONDICOES DE CONTORNO
      CALL CAEPP(F,T,VE)
C-----RESOLUCAO DO SISTEMA RESULTANTE
      CALL PPINV(T)
      CALL PMVPP(T,VE,V)
      DO 420 I=1,6
420 VE(I)=0.
C-----REARRANJAR VETOR DE ESTADO DA JUNTA 1
      DO 430 I=1,3
      KA=KAC(I)
430 VE(KA)=-V(I)
C-----DESLOCAMENTOS E REACOES (APOIOS ELASTICOS ) JUNTA 1
      LDF=1
      IND=JC(LDF)
      READ(2,LDF)MF,DMF
      DO 440 I=1,3
440 LR(I)=LRE(I)
      CALL AEPP(DMF,IND,VE,K,VR,LR)
      DO 450 I=1,6
450 VAUX(I)=VE(I)
      IF(MF(2)+K)413,413,414
414 LD=LDF-1
      WRITE(4,LDR)LD,VR
413 WRITE(3,LDD)VAUX(1),VAUX(2),VAUX(3)
      CALL LINK(PPMT4)
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR PPMT3
COMMON 684 VARIABLES 634 PROGRAM 510

END OF COMPILATION

PAGE 1 PR-PPMT4

// JOB

PR-PPMT4

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPPMT4

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),LRE(3),LRD(3),VE(7)

DIMENSION TITUL(20),LEN(3),KAC(3),MF(3),MT(3),V(7),LR(3),VR(3)

DIMENSION F(7,7),T(7,7),DMT(10),DMF(14),AM(6),SM(4),MC(200)

DIMENSION JC(201),VD(3),VAUX(7)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC

COMMON TITUL,LDR,VAUX

EQUIVALENCE (T(1),F(1)),(VE(1),VD(1))

DEFINE FILE 1(200,42,U,LDT),2(201,45,U,LDF),3(201,9,U,LDD)

DEFINE FILE 4(201,10,U,LDR)

DATA LDF,LDD,LDT/2,2,1/

NUMB=NBAR

NJ=NBAR+1

DO 605 I=1,7

605 V(I)=VAUX(I)

C-----IMPRIMIR TITULO DO CARREGAMENTO

WRITE(W,505)TITUL

505 FORMAT(/1X'*RESULTADOS* ',20A4/1X,80('='))

WRITE(W,515)

515 FORMAT(/1X'ACOES NAS EXTREMIDADES DAS BARRAS'/)

WRITE(W,525)

FORMAT(1X'BARRA'13X'FX'10X'FY'9X'MZZ')

C-----SEGUNDA PASSAGEM PELA ESTRUTURA - RESULTADOS

501 IND=MC(LDT)

READ(1,LDT)MT,DMT,XL,CX,CY

CALL PPML(DMT,MT,T,XL,IND)

CALL PPMG(T,IND,CX,CY)

CALL PMVPP(T,V,VE)

C-----ACOES DE EXTREMO DE BARRAS PRINCIPAIS

AM(1)=V(4)*CX+V(5)*CY

AM(2)=V(5)*CX-V(4)*CY

AM(3)=V(6)

AM(4)=VE(4)*CX+VE(5)*CY

AM(5)=VE(5)*CX-VE(4)*CY

AM(6)=VE(6)

LD=LDT-1

WRITE(W,535)LD,AM

535 FORMAT(1X,I5,1X'JJ',3F12.2/7X'JK',3F12.2/)


```

      IND=JC(LDF)
      READ(2,LDF)MF,DMF
      IF(LDT-NBAR)502,502,503
502  CALL PPMF(DMF,MF,F,IND)
      IF(MF(1))506,506,504
C-----ACOES DE EXTREMO DE RAMIFICACOES
504  NUMB=NUMB+1
      LD=LDT
      IND=MC(NUMB)
      READ(1,NUMB)MT,DMT,XL,CX,CY
      CALL PPBR(DMT,F,IND,XL,CX,CY,SM)
      IF(IND)507,507,508
507  DO 520 I=5,10
520  DMT(I)=0.
508  AUX=VE(1)*CX+VE(2)*CY
      TEP=VE(1)*CY-VE(2)*CX
      AM(1)=DMT(5)-SM(1)*AUX
      AM(2)=DMT(6)+SM(4)*TEP+SM(3)*VE(3)
      AM(3)=DMT(7)+SM(3)*TEP+SM(2)*VE(3)/2.
      AM(4)=DMT(8)+SM(1)*AUX
      AM(5)=DMT(9)-SM(4)*TEP-SM(3)*VE(3)
      AM(6)=DMT(10)+SM(3)*TEP+SM(2)*VE(3)
      WRITE(W,535)NUMB,AM
      V(1)=AM(1)*CX-AM(2)*CY
      V(2)=AM(1)*CY+AM(2)*CX
      V(3)=AM(3)
      K=NUMB+1
      WRITE(4,LDR)K,V(1),V(2),V(3)
      LDT=LD
506  CALL PMVPP(F,VE,V)
      WRITE(3,LDD)VD
C-----REACOES (APOIOS ELASTICOS) JUNTAS INTERNAS
      IF(MF(2))501,501,509
509  DO 570 I=1,3
      J=I+2
570  VR(I)=-DMF(J)*V(I)
      WRITE(4,LDR)LDT,VR
      GOTO 501
C-----REACOES E DESLOCAMENTOS JUNTA NJ
503  DO 580 I=1,3
580  LR(I)=LRD(I)
      CALL AEPP(DMF,IND,VE,K,VR,LR)
      WRITE(3,LDD)VD
      IF(MF(2)+K)512,512,513
513  WRITE(4,LDR)LDT,VR
512  LD=LDR-1
      IF(LD)516,516,517
C-----IMPRESSAO DAS REACOES
517  WRITE(W,545)
545  FORMAT(/1X'REACOES'/)
      WRITE(W,555)

```

PAGE 3 PR-PPMT4

```
555  FORMAT(1X'JUNTA'10X'RX'10X'RY'9X'RZZ')
      DO 590 I=1,LD
      READ(4,I)K,VR
590  WRITE(W,565)K,VR
565  FORMAT(1X,I5,3F12.2)
516  WRITE(W,575)
575  FORMAT(/1X'DESLOCAMENTOS GENERALIZADOS'/)
      WRITE(W,585)
585  FORMAT(1X'JUNTA'10X'DX'10X'DY'9X'DZZ')
C-----IMPRESSAO DOS DESLOCAMENTOS
      DO 600 I=1,NJ
      READ(3,I)VR
600  WRITE(W,595)I,VR
595  FORMAT(1X,I5,3F12.5)
      CALL LINK(PPMT5)
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR PPMT4
COMMON 684 VARIABLES 376 PROGRAM 940

END OF COMPILATION

PAGE 1 PR-PPMT5

// JOB

PR-PPMT5

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPPMT5

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20),KOM(4),LRE(3),LRD(3)

DIMENSION INP5(12)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD

DEFINE FILE 1(200,42,U,LDT),2(201,45,U,LDF),3(201,9,U,LDD)

DEFINE FILE 4(201,10,U,LDR)

DATA INP5(1,1,1,1,1,1,1,1,1,1,1,1),/,'E','S','T','/','F','I','M'/'

C-----LER CODIGO DE DECISAO

READ(RE,601)KOM

601 FORMAT(4A1)

I=0

DO 610 J=1,9,4

K=J+3

I=I+1

IF(ITEST(INP5,J,K,KOM,1))610,602,610

602 GOTO(605,615,625),I

610 CONTINUE

WRITE(W,603)KOM

603 FORMAT(7X'CODIGO ('4A1') NAO IDENTIFICADO')

GOTO 604

625 WRITE(W,606)KOM

606 FORMAT(//1X,4A1)

C-----FINALIZAR PROCESSAMENTO

604 CALL CABP(1)

WRITE(W,607)

607 FORMAT(//1X'PROCESSAMENTO CONCLUIDO'//1X'HR. MAQUINA INICIAL='//1X

'HR. MAQUINA FINAL ='//1X'TEMPO PROCESSAMENTO= HRS.* MIN

.'//)

WRITE(W,608)

608 FORMAT('1')

CALL EXIT

C-----PROCESSAR NOVO CARREGAMENTO

605 CALL LINK(PPMT2)

C-----PROCESSAR NOVA ESTRUTURA

615 CALL LINK(PPMT1)

END

FEATURES SUPPORTED

PAGE 1 SR-PPMT

// JOB

SR-PPMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR SR-PPBC - CARGAS NAS BARRAS - ROTINA PRINCIPAL

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE PPBC(DT, ID, DI, I, KOM, INC)
  INTEGER RE, W
  DIMENSION DT(6), DI(2), DMT(10), KOM(4), LCAB(4), OBRA(20), FULA(20)
  DIMENSION RESP(20), LRE(3), LEN(3), KAC(3), MT(3), LRD(3), MC(200)
  COMMON RE, W, NBAR, NRAM, LCAB, OBRA, FULA, RESP, LRE, LRD, LEN, KAC, MC
  N1=DI(1)
  N2=DI(2)
  K=0
  IPR=1
  IF((N2-N1-2)*INC) 101, 101, 102
102 IF(I-2) 101, 101, 103
101 DO 100 J=N1, N2
  K=1+K*INC
  IF(J-NBAR) 104, 104, 106
106 IPR=2
104 READ(1, J) MT, DMT, XL
  DO 120 N=1, 3
  IF(DMT(N)) 121, 121, 120
120 CONTINUE
  IF(MC(J)) 107, 107, 108
107 DO 110 N=5, 10
110 DMT(N)=0.
  MC(J)=1
108 A=DT(K)
  B=A
  GOTO(1, 1, 2, 2, 3, 3, 3, 4), I
C-----CARGAS LINEARES
1 CALL PPLIN(DMT, A, B, IPR, I, XL, MT)
  GOTO 109
2 K=K+1
  B=DT(K)
  GOTO 1
C-----CARGAS CONCENTRADAS
3 CALL PPFCB(DMT, DT, K, IPR, I, XL, INC, N1, N2)
  GOTO 109
4 K=K+1
  B=DT(K)
C-----VARIACAO UNIFORME DE TEMPERATURA
CALL PPTUX(DMT, A, B, IPR, XL)
109 WRITE(1, J) MT, DMT
```

PAGE 2 SR-PPMT

```
100 CONTINUE
    IF(I-8)111,112,112
111 WRITE(W,118)KOM,ID,DI,(DT(J),J=1,K)
118 FORMAT(1X,5A1,2F5.0,6F10.2)
    RETURN
112 WRITE(W,119)KOM,ID,DI,(DT(J),J=1,K)
119 FORMAT(1X,5A1,2F5.0,3(F10.2,F10.6))
    RETURN
103 IER=5
    WRITE(W,11)KOM,ID,DI,IER
11 FORMAT(//5X,5A1,2F5.0//5X'ERRO='I5)
    CALL EXIT
121 IER=6
    WRITE(W,12)J,IER
12 FORMAT(/5X'BARRA='I5,3X'ERRO='I5)
    CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PPBC
COMMON 400 VARIABLES 52 PROGRAM 442

RELATIVE ENTRY POINT ADDRESS IS 0070 (HEX)

END OF COMPILEATION

```
// FOR SR-PPLIN - CARGAS DISTRIBUIDAS LINEARES
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
    SUBROUTINE PPLIN(DMT,P,Q,IPR,I,XL,MT)
        DIMENSION DMT(10),MT(3)
        AUX=XL*XL
        EIZ=DMT(2)*DMT(3)
        GOTO(11,12),IPR
C-----BARRAS PRINCIPAIS
11 IF(I-1)13,13,14
C-----SEGUNDO X
13 DMT(5)=DMT(5)-(2.*P+Q)*AUX/DMT(3)/DMT(1)/6.
    DMT(8)=DMT(8)-(P+Q)*XL/2.
    RETURN
C-----SEGUNDO Y
14 IF(MT(2))21,21,22
C-----BARRAS DE FLEXAO LIVRE
21 DMT(6)=DMT(6)+(4.*P+Q)*AUX*AUX/EIZ/120.
    DMT(7)=DMT(7)+(3.*P+Q)*AUX*XL/EIZ/24.
    DMT(9)=DMT(9)-(P+Q)*XL/2.
    DMT(10)=DMT(10)+(2.*P+Q)*AUX/6.
```

PAGE 3 SR-PPMT

```
      RETURN
C-----BARRAS S/ BASE ELASTICA
22      GAM2=SQRT(DMT(4)/EIZ/4.)
      GAM=SQRT(GAM2)
      AUX=GAM*XL
      SN=SIN(AUX)
      CS=COS(AUX)
      AUX=EXP(AUX)
      SH=(AUX-1./AUX)/2.
      CH=(AUX+1./AUX)/2.
      A=CH*CS
      B=SH*SN
      C=SH*CS-CH*SN
      D=SH*CS+CH*SN
      P=P/DMT(4)
      Q=Q/DMT(4)
      DMT(6)=DMT(6)+(P-Q)*(D/XL/GAM/2.)+Q-A*P
      DMT(7)=DMT(7)+(A-1.)*(P-Q)/XL-C*GAM*P
      DMT(9)=DMT(9)-((Q-P)*B/XL+D*GAM*P)*GAM2*EIZ*2.
      DMT(10)=DMT(10)+((P-Q)*C/XL+B*GAM*P*2.)*GAM*EIZ
      RETURN
C-----RAMIFICACOES
12      IF(I-1)16,16,17
C-----SEGUNDO X
16      DMT(5)=DMT(5)-(2.*P+Q)*XL/6.
      DMT(8)=DMT(8)-(P+2.*Q)*XL/6.
      RETURN
C-----SEGUNDO Y
17      DMT(6)=DMT(6)-(7.*P+3.*Q)*XL/20.
      DMT(7)=DMT(7)-(3.*P+2.*Q)*AUX/60.
      DMT(9)=DMT(9)-(3.*P+7.*Q)*XL/20.
      DMT(10)=DMT(10)+(2.*P+3.*Q)*AUX/60.
      RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PPLIN
COMMON 0 VARIABLES 48 PROGRAM 656

RELATIVE ENTRY POINT ADDRESS IS 004F (HEX)

END OF COMPILATION

// FOR SR-PPFCB = CARGAS CONCENTRADAS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE PPFCB(DMT,DT,K,IPR,I,XL,INC,N1,N2)

```

        DIMENSION DMT(10),DT(6)
        NV=1
        J=K-1
        IF((N2-N1)*INC)101,101,102
101     NV=3
102     GOTO(11,12),IPR
C-----BARRAS PRINCIPAIS
11     DO 100 N=1,NV
        J=J+1
        IF(DT(J))104,106,104
104     P=DT(J)
        J=J+1
        A=XL-DT(J)
        B=A*A
        IF(I-6)13,14,16
C-----SEGUNDO X
13     DMT(5)=DMT(5)-P*A/DMT(3)/DMT(1)
        DMT(8)=DMT(8)-P
        GOTO 111
C-----SEGUNDO Y
14     DMT(6)=DMT(6)+P*A*B/DMT(3)/DMT(2)/6.
        DMT(7)=DMT(7)+P*B/DMT(3)/DMT(2)/2.
        DMT(9)=DMT(9)-P
        DMT(10)=DMT(10)+P*A
        GOTO 111
C-----SEGUNDO ZZ
16     DMT(6)=DMT(6)-P*B/DMT(3)/DMT(2)/2.
        DMT(7)=DMT(7)-P*A/DMT(3)/DMT(2)
        DMT(10)=DMT(10)-P
111     K=J
100     CONTINUE
106     RETURN
C-----RAMIFICACOES
12     AUX=XL*XL
        DO 200 N=1,NV
        J=J+1
        IF(DT(J))107,108,107
107     P=DT(J)
        J=J+1
        A=DT(J)
        B=XL-A
        IF(I-6)17,18,19
C-----SEGUNDO X
17     DMT(5)=DMT(5)-P*B/XL
        DMT(8)=DMT(8)-P*A/XL
        GOTO 112
C-----SEGUNDO Y
18     DMT(6)=DMT(6)-P*B*B*(3.*A+B)/AUX/XL
        DMT(7)=DMT(7)-P*A*B*B/AUX
        DMT(9)=DMT(9)-P*A*A*(A+3.*B)/AUX/XL
        DMT(10)=DMT(10)+P*A*A*B/AUX

```

PAGE 5 SR-PPMT

```
      GOTO 112
C-----SEGUNDO ZZ
19      DMT(6)=DMT(6)+6.*P*A*B/AUX/XL
        DMT(7)=DMT(7)+P*B*(2.*A-B)/AUX
        DMT(9)=DMT(9)-6.*P*A*B/AUX/XL
        DMT(10)=DMT(10)+P*A*(2.*B-A)/AUX
112     K=J
200     CONTINUE
108     RETURN
        END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PPFCB
COMMON 0 VARIABLES 24 PROGRAM 608

RELATIVE ENTRY POINT ADDRESS IS 0024 (HEX)

END OF COMPILATION

```
// FOR SR-PPMF - MATRIZ FRONTEIRA
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
      SUBROUTINE PPMF(DMF,MF,T,IND)
      DIMENSION MF(3),DMF(14),T(7,7)
      DO 100 J=1,7
      DO 100 K=1,7
        T(J,K)=0.
100     T(J,J)=1.
      DO 110 J=4,6
110     T(J,J)=-1.
      IF(MF(2))101,101,102
102     DO 120 J=1,3
        K=J+3
120     T(K,J)=-DMF(J+2)
101     IF(MF(3))103,103,104
104     DO 130 J=1,3
        K=J+3
130     T(J,K)=DMF(J+5)
103     IF(IND)106,106,107
107     DO 140 J=1,6
        K=J+8
140     T(J,7)=DMF(K)
106     RETURN
        END
```

FEATURES SUPPORTED
ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR PPMF
COMMON 0 VARIABLES 4 PROGRAM 232

RELATIVE ENTRY POINT ADDRESS IS 0010 (HEX)

END OF COMPILATION

// FOR SR-PPML - MATRIZ TRANSFERENCIA SISTEMA LOCAL

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE PPML(DMT,MT,T,XL,IND)

DIMENSION DMT(10),MT(3),T(7,7)

DO 100 I=1,7

DO 100 J=1,7

100 T(I,J)=0.

T(1,1)=1.

T(4,4)=-1.

T(7,7)=1.

EIZ=DMT(2)*DMT(3)

IF(MT(2))101,101,102

C-----P/ BARRAS DE FLEXAO LIVRE

101 T(2,2)=1.

T(3,3)=1.

T(5,5)=-1.

T(6,6)=-1.

T(2,3)=XL

T(6,5)=XL

T(1,4)=-XL/DMT(3)/DMT(1)

T(3,6)=-XL/EIZ

T(2,6)=T(3,6)*XL/2.

T(3,5)=-T(2,6)

T(2,5)=T(3,5)*XL/3.

GOTO 111

C-----P/ BARRAS S/ BASE ELASTICA

102 GAM2=SQRT(DMT(4)/EIZ/4.)

GAM=SQRT(GAM2)

GL=GAM*XL

SN=SIN(GL)

CS=COS(GL)

GL=EXP(GL)

SH=(GL-1./GL)/2.

CH=(GL+1./GL)/2.

C=SH*CS-CH*SN

D=SH*CS+CH*SN

T(2,2)=CH*CS

T(3,3)=T(2,2)

T(5,5)=-T(2,2)

T(6,6)=T(5,5)

PAGE 7 SR-PPMT

```
T(2,3)=D/GAM/2.
T(6,5)=T(2,3)
T(2,5)=-C/GAM2/EIZ/GAM/4.
T(2,6)=-SH*SN/GAM2/EIZ/2.
T(3,5)=-T(2,6)
T(3,2)=C*GAM
T(5,6)=T(3,2)
T(3,6)=-D/EIZ/GAM/2.
T(5,2)=2.*D*GAM2*EIZ*GAM
T(5,3)=2.*SH*SN*GAM2*EIZ
T(6,2)=-T(5,3)
T(6,3)=T(3,2)*EIZ
111 IF(IND)103,103,104
C-----VETOR DE CARGAS
104 DO 110 I=1,6
110 T(I,7)=DMT(I+4)
103 RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PPML
COMMON 0 VARIABLES 50 PROGRAM 544

RELATIVE ENTRY POINT ADDRESS IS 0044 (HEX)

END OF COMPILATION

// FOR SR-PPMG - MATRIZ TRANSFERENCIA SISTEMA GLOBAL

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE PPMG(T,IND,CX,CY)
DIMENSION T(7,7),A(6,6)
DO 100 K=1,2
KC=3*K
KB=KC-1
KA=KC-2
DO 100 J=1,6
A(J,KA)=T(J,KA)*CX-T(J,KB)*CY
100 A(J,KB)=T(J,KA)*CY+T(J,KB)*CX
A(J,KC)=T(J,KC)
DO 110 K=1,2
KC=3*K
KB=KC-1
KA=KC-2
DO 110 J=1,6
T(KA,J)=A(KA,J)*CX-A(KB,J)*CY
T(KB,J)=A(KA,J)*CY+A(KB,J)*CX
```

PAGE 8 SR-PPMT

```
110 T(KC,J)=A(KC,J)
    IF(IND)101,101,102
102 A(1,1)=T(1,7)*CX-T(2,7)*CY
    T(2,7)=T(1,7)*CY+T(2,7)*CX
    T(1,7)=A(1,1)
    A(1,1)=T(4,7)*CX-T(5,7)*CY
    T(5,7)=T(4,7)*CY+T(5,7)*CX
    T(4,7)=A(1,1)
101 RETURN
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PPMG
COMMON 0 VARIABLES 122 PROGRAM 404

RELATIVE ENTRY POINT ADDRESS IS 007E (HEX)

END OF COMPILATION

// FOR SR-PPBR - MATRIZ DE RIGIDEZ DAS RAMIFICACOES

*ONE WORD INTEGERS
*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
    SUBROUTINE PPBR(DMT,F,IND,XL,CX,CY,SM)
    DIMENSION DMT(10),F(7,7),SM(4)
C-----SISTEMA LOCAL
    SM(1)=DMT(1)*DMT(3)/XL
    SM(2)=4.*DMT(2)*DMT(3)/XL
    SM(3)=1.5*SM(2)/XL
    SM(4)=2.*SM(3)/XL
C-----SISTEMA GLOBAL - SUB-MATRIZ DA MATRIZ FRONTEIRA
    F(4,1)=F(4,1)-SM(1)*CX*CX-SM(4)*CY*CY
    F(5,2)=F(5,2)-SM(1)*CY*CY-SM(4)*CX*CX
    F(4,2)=F(4,2)-(SM(1)-SM(4))*CX*CY
    F(5,1)=F(5,1)-(SM(1)-SM(4))*CX*CY
    F(4,3)=F(4,3)-SM(3)*CY
    F(6,1)=F(6,1)-SM(3)*CY
    F(5,3)=F(5,3)+SM(3)*CX
    F(6,2)=F(6,2)+SM(3)*CX
    F(6,3)=F(6,3)-SM(2)
    IF(IND)101,101,102
C-----ACOES EQUIVALENTES NAS JUNTAS
102 F(4,7)=F(4,7)-DMT(8)*CX+DMT(9)*CY
    F(5,7)=F(5,7)-DMT(8)*CY-DMT(9)*CX
    F(6,7)=F(6,7)-DMT(10)
101 RETURN
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PPBR
COMMON 0 VARIABLES 16 PROGRAM 364

RELATIVE ENTRY POINT ADDRESS IS 0019 (HEX)

END OF COMPILATION

// FOR SR-AEPP - REACOES NOS APOIOS EXTREMOS

*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM

```

SUBROUTINE AEPP(DMF,IND,V,IR,VR,LR)
  DIMENSION DMF(14),V(7),VR(3),LR(3)
  IR=0
  IF(IND)101,101,102
101 DO 110 I=9,14
110 DMF(I)=0.
102 DO 120 I=1,3
    J=I+3
    K=I+11
    IF(LR(I))103,103,104
104 VR(I)=V(J)-DMF(K)
    V(I)=DMF(I+8)
    IR=1
    GOTO 120
103 VR(I)=-DMF(I+2)*V(I)
    V(J)=VR(I)+DMF(K)
120 CONTINUE
  RETURN
  END

```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR AEPP
COMMON 0 VARIABLES 8 PROGRAM 186

RELATIVE ENTRY POINT ADDRESS IS 0011 (HEX)

END OF COMPILATION

PAGE 1 PR-PEMT

// JOB

PR-PEMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(TYPEWRITER)

INTEGER RE,W

COMMON RE,W

WRITE(1,11)

11 FORMAT(/1X'LIGAR CHAVE 1 P/1442, DESLIGAR P/2501READER'//1X'LIGAR
*CHAVE 2 P/1132, DESLIGAR P/1403PRINTER'/1X'POSICIONAR CHAVES E DAR
*START')

PAUSE 1111

RE=2

W=3

CALL DATSW(1,I)

CALL DATSW(2,J)

GOTO(12,13),I

13 RE=8

12 GOTO(16,17),J

17 W=5

16 CALL LINK(PEMT1)

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

IOCS

CORE REQUIREMENTS FOR PEMT

COMMON 2 VARIABLES 2 PROGRAM 146

END OF COMPILATION

PR-PEMT1

LOG DRIVE	CART SPEC	CART AVAIL	PHY DRIVE
0000	0005	0005	0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPENT1

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER REVIEW

```

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),DI(2),DD(15),DT(6)

```

DIMENSION KOM(4),LRF(6),LRD(6),INP1(24),INP2(48),INP3(80),INP4(16)

```

DIMENSION RUM(4),VERK(8),VERD(8),VINT(1,24),
DIMENSION MF(3),DMT(20),R(9),DM(3),MT(3)

```

COMMON RE, W, NBAR, NRAM, LCAB, OBRA, FULA, RESP, LRE, LRD

```

COMMON RE, W, NBAR, NRAM, ECAS, SUBRA, I, CLAY, RESF, ERE, ERD
DEFINE FILE 1(200,99,U,LDT),2(201,84,U,LDF),3(201,18,U,LDD)

```

[illegible]

```
DATA INP1/'/'/'E','S','T','/'/'/'J','O','/'/'/'F','O','/'/'/'
```

[illegible]

DATA INP2/'/'/'/'/'/'/'/'/'N'/'/'/'/'/'N'/'R','*', 'L','R','E','*', 'L',

DATA INPZ77070707N070707NRYR000LEIRYELXLE
*R'D' *R'A'M' * * * X * * * Y * * *

!R!D!!R!A!*M!*!*!L!*A!*!*!*L!*!*!*L!*
!Z!!*!Y!Y!*!*!Y!Y!*!*!Z!Z!Z!Z!*!

[illegible]

DATA INP3/'*','*','*','Y','*','*','*','Y','*','*','*','7','*','*'.

[illegible]

* * K * A * * * K * Y * * * K * Z * * * K * A * A * * K * Y
* * Y * * K * 7 * 7 * / / * X * / / * X * / / * I *

[illegible]

*! ! ! / ! ! ! C ! ! P ! ! 7 ! ! / ! ! ! / ! ! ! Y ! ! P ! ! / ! ! ! / ! ! ! 7 ! ! P ! /

DATA INP4//T/M/P//E/I/M//E/S/T//Z//#//

*'!'!'/

153 NBAR=0

```

NBAR=0
NRAM=0

```

$$NJ=0$$

DO 240

$$MF(1) = 0$$
$$R(I) = 0.$$

DO 100

LCAB(1)=2

C-----INICIALIZAR LISTA RESTRICAO APOIOS EXTREMOS

C-----CODIGOS FIXO = 1 LIVRE = 0

DO 110 $I=1.6$

$$\text{LRE}(i) = i$$
$$R(I+3) = 0.$$

```
110 LRD(1)=1
```

-IMPRIMIR CABECALHO DE IDENTIFICACAO

CALL CARP (2)

CALL CADP
LCAB(1)=1

C-----LER DADOS DE IDENTIFICACAO

PAGE 2 PR-PEMT1

```
101 READ(RE,102)KOM
102 FORMAT(4A1)
103 WRITE(W,103)KOM
103 FORMAT(/1X,4A1)
I=0
DO 120 J=1,21,4
K=J+3
I=I+1
IF(ITEST(INP1,J,K,KOM,1))120,104,120
104 GOTO(101,115,125,135,145,113),I
120 CONTINUE
105 WRITE(W,106)KOM
106 FORMAT(/20X'NAO EXISTE O CODIGO ('4A1')'////)
CALL EXIT
115 READ(RE,107)OBRA
107 WRITE(W,108)OBRA
107 FORMAT(20A4)
108 FORMAT(1X,20A4)
LCAB(2)=1
GOTO 101
125 READ(RE,107)FULA
107 WRITE(W,108)FULA
107 LCAB(3)=1
108 GOTO 101
135 READ(RE,107)RESP
107 WRITE(W,108)RESP
107 LCAB(4)=1
108 GOTO 101
145 LCAB(1)=2
107 GOTO 101
127 WRITE(W,128)KOM,ID,(DD(K),K=1,N)
128 FORMAT(1X,4A1,A1,15F5.0)
C-----LER DADOS DE DEFINICAO
113 READ(RE,114)KOM,ID,DD
114 FORMAT(4A1,A1,15F5.0)
I=0
DO 140 J=1,45,4
K=J+3
I=I+1
IF(ITEST(INP2,J,K,KOM,1))140,116,140
116 GOTO(165,175,185,195,205,215,215,215,215,215,225),I
140 CONTINUE
GOTO 105
225 WRITE(W,103)KOM
GOTO 235
165 CALL CONCA(DD,1,200,IER,2,N)
IF(IER)121,121,198
121 NBAR=DD(1)
NB=NBAR
NJ=NBAR+1
WRITE(W,112)KOM,DD(1)
```

```

PAGE      3      PR=PENT1
112  FORMAT(1X,4A1,1X,2F5.0)
      DO 130 I=1,NJ
130   WRITE(2,I)MF,R
      DO 150 I=1,NBAR
150   WRITE(1,I)MF,R
      GOTO 113
175   CALL CONCA(DD,1,200,IER,2,N)
      IF( IER)122,122,198
122   NRAM=DD(1)
      NB=NB+NRAM
      NJ=NJ+NRAM
      WRITE(W,112)KOM,DD(1)
      DO 160 I=NBAR,NB
160   WRITE(1,I)MF,R
      DO 170 I=NBAR,NJ
170   WRITE(2,I)MF,R
      GOTO 113
185   DO 180 I=1,6
180   LRE(I)=DD(I)
191   WRITE(W,128)KOM,(DD(I),I=1,6)
      GOTO 113
195   DO 190 I=1,6
190   LRD(I)=DD(I)
      GOTO 191
205   CALL CONCA(DD,15,NBAR,IER,4,N)
      IF( IER)123,123,198
123   CALL PEJR(DD,ID,N)
      GOTO 127
215   CALL CONCA(DD,15,NBAR,IER,4,N)
      IF( IER)124,124,198
124   K=I+4
      CALL PEJL(DD,K,N)
      GOTO 127
129   WRITE(W,132)KOM,ID,DI,(DT(J),J=1,INC)
132   FORMAT(1X,4A1,A1,2F5.0,6F15.3)
      GOTO 235
131   WRITE(W,133)KOM,ID,DI,(DT(J),J=1,INC)
133   FORMAT(1X,4A1,A1,2F5.0,6E15.6)
C-----LER DADOS DE DEFINICAO COMPLEMENTAR
235   READ(RE,117)KOM,ID,DI,DT
117   FORMAT(4A1,A1,2F5.0,6F10.0)
      I=0
      DO 200 J=1,77,4
      K=J+3
      I=I+1
      IF( ITEST(INP3,J,K,KOM,1))200,118,200
118   IF( I-9)134,134,136
134   CALL CONCB(DI,ID,INC,IER,NJ,4)
      IF( IER)137,137,199
137   CALL PEDJ(DT,DI,I,INC)
      IF( I-3)129,129,131

```


PAGE 4 PR-PEMT1

```
136 CALL CONCB(DI,ID,INC,IER,NB,3)
    IF(IER)138,138,199
138 CALL PEDB(DT,DI,I,INC)
    IF(I-17)131,131,129
200 CONTINUE
    IRAM=0
    NUMB=NBAR
    READ(2,1)MF,X,Y,Z
    DO 210 I=1,NBAR
        K=I+1
        IF(IRAM)141,142,141
141 IND=1
        NUMB=NUMB+1
        LD=NUMB+1
        READ(1,NUMB)MT,DMT,R,DM
        READ(2,LD)MF,XJ,YJ,ZJ
        LD=NUMB
        GOTO 143
142 IND=0
        XJ=X
        YJ=Y
        ZJ=Z
        READ(1,I)MT,DMT,R,DM
        READ(2,K)MF,X,Y,Z
        LD=I
143 DX=X-XJ
        DY=Y-YJ
        DZ=Z-ZJ
        IRAM=MF(1)
        IF(MT(3))144,146,144
144 ICP=1
        DXP=DMT(9)-XJ
        DYP=DMT(10)-YJ
        DZP=DMT(11)-ZJ
        GOTO 147
146 ICP=0
C-----FORMAR VETOR ROTACAO R(9)
147 CALL PEVR(ICP,DX,DY,DZ,DXP,DYP,DZP,R,DM)
    WRITE(1,LD)MT,DMT,R,DM
    IF(IND)210,210,142
210 CONTINUE
152 I=0
    WRITE(W,103)KOM
    DO 220 J=1,13,4
        K=J+3
        I=I+1
        IF(ITEST(INP4,J,K,KOM,1))220,148,220
148 GOTO(253,245,153,265),I
220 CONTINUE
    GOTO 105
C-----IMPRIMIR COMPRIMENTOS E R(9) SE ESPECIFICADO /IMP
```

PAGE 5 PR-PEMT1

```
253 WRITE(W,149)
149 FORMAT(/1X'BARRA'2X'COMPR.'4X'BETA'32X'R(1 A 9)'/)
DO 230 K=1,NB
  READ(1,K)MT,DMT,R,DM
  WRITE(W,151)K,DM(1),DM(2),R
151 FORMAT(1X,I5,11F8.3)
230 CONTINUE
  READ(RE,102)KOM
  GOTO 152
245 CALL EXIT
265 CALL LINK(PEMT2)
198 WRITE(W,196)KOM,DD,IER
196 FORMAT(/5X,4A1,15F5.0//5X'ERRO='I5)
  CALL EXIT
199 WRITE(W,197)KOM,ID,DI,IER
197 FORMAT(/5X,5A1,2F5.0//5X'ERRO='I5)
  CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR PEMT1
COMMON 200 VARIABLES 428 PROGRAM 1412

END OF COMPILATION

PR-PEMT2

```
LOG DRIVE      CART SPEC      CART AVAIL      PHY DRIVE
  0000          0005          0005          0000
```

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPEMT2

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)
```

INTEGERS REW

```

DIMENSION OBRA(20),RESP(20),FULA(20),TITUL(20),LCAB(4),KOM(4),D(3)

```

DIMENSION DT(6),DI(2),LRF(6),LRD(6),LEN(6),KAC(6),INP5(80)

DIMENSION INP6(36),MC(200),JC(201)

COMMON RE, W, NBAR, NRAM, LCAB, OBRA, FULA, RESP, LRE, LRD, LEN, KAC, MC, JC

COMMON TITUL

EQUIVALENCE (D(1),DI(1)),(DT(1),D(3))

```

DEFINE FILE 1(200,99,U,LDT),2(201,84,U,LDF),3(201,18,U,LDD)

```

DEFINE FILE 4 (201,19,U,LDR)

[illegible]
$$NJ = NBAR + 1$$
$$NB = NBAR + NRAM$$
$$NJT = NJ + NRAM$$

C-----ZERAR CONTROLES DE BARRAS E JUNTAS CARREGADAS

$$KM=0$$
$$KJ=0$$

```
DO 300 I=1,NB
```

$$MC(1) = 0$$
$$JC(1) = 0$$

JC(NJT)=0

CALL CABP(2)

C-----LER E IMPRIMIR TITULO DO CARREGAMENTO

READ (RE, 301) TITUL

WRITE (W, 302) TITUL

301 FORMAT(20A4)

```
302 FORMAT(7/1X'*LIST. DADOS* ',20A4/1X,80('=''))
```

GOTO 303

```
323 WRITE(W,324)KOM,ID,DI,(DT(J),J=1,INC)
```

LER DADOS DO CARREGAMENTO

303 READ(RE,304)KOM,ID,DI,DT

```

303 READ (REV, 304) RCH, ID, B1, B2,
304 FORMAT(4A1, A1, 2F5, 0, 6F10, 0)

```

```

PAGE      2      PR-PEMT2
324  FORMAT(1X,4A1,A1,2F5.0,6F10.2)
      I=0
      DO 320 J=1,77,4
      K=J+3
      I=I+1
      IF(ITEST(INP5,J,K,KOM,1))320,306,320
320  CONTINUE
      GOTO 311
306  CALL CONCB(DI,ID,INC,IER,NJT,3)
      IF(IER)307,307,398
307  IF(I-14)308,308,309
308  CALL PEBC(DT,ID,DI,I,KOM,INC)
      GOTO 303
309  K=I+1
      CALL PEJC(DT,DI,K,INC)
      GOTO 323
311  I=0
      DO 330 J=1,33,4
      K=J+3
      I=I+1
      IF(ITEST(INP6,J,K,KOM,1))330,312,330
330  CONTINUE
      WRITE(W,313)KOM
313  FORMAT(/1X'NAO EXISTE O CODIGO ('4A1')'///)
      CALL EXIT
312  IF(I-6)331,331,332
331  CALL CONCB(DI,ID,INC,IER,NJT,4)
      IF(IER)333,333,398
333  K=I+21
      CALL PEJC(DT,DI,K,INC)
      GOTO 323
332  IF(I-8)305,305,315
305  CALL CONCA(D,3,NJT,IER,4,N)
      IF(IER)314,314,398
314  IF(N-2)316,317,318
316  IER=7
      GOTO 398
C-----INICIALIZAR CONTROLES DE CARGAS CONFORME /CAR *CAR
317  N3=1
      GOTO 319
318  N3=D(3)
319  N1=D(1)
      N2=D(2)
      IF(I-7)325,325,335
325  IF(KM)321,321,322
321  DO 340 J=1,NB
340  MC(J)=0
      KM=1
322  DO 350 J=N1,N2,N3
350  MC(J)=1
      GOTO 393

```

PAGE 3 PR-PEMT2

```
335 IF(KJ)342,342,326
342 DO 360 J=1,NJT
360 JC(J)=0
    KJ=1
326 DO 370 J=N1,N2,N3
370 JC(J)=1
393 WRITE(W,327)KOM,ID,(D(K),K=1,N)
327 FORMAT(1X,4A1,A1,3F5.0)
    GOTO 303
315 WRITE(W,324)KOM
    CALL LINK(PEMT3)
398 WRITE(W,399)KOM,ID,DI,IER
399 FORMAT(/5X,4A1,A1,2F5.0//5X'ERRO='I5)
    CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR PEMT2
COMMON 674 VARIABLES 194 PROGRAM 642

END OF COMPILATION

PAGE 1 PR-PEMT3

// JOB

PR-PEMT3

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPEMT3

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(DISK)

INTEGER RE,W

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),LRE(6),LRD(6),VE(13)

DIMENSION TITUL(20),LEN(6),KAC(6),MF(3),MT(3),V(13),R(9),LR(6)

DIMENSION VR(6),F(13,13),T(13,13),P(13,13),DMT(20),DMF(27)

DIMENSION SMR(6,6),MC(200),JC(201),VAUX(13)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC

COMMON TITUL,LDR,VAUX

EQUIVALENCE (F(1),SMR(1))

DEFINE FILE 1(200,99,U,LDT),2(201,84,U,LDF),3(201,18,U,LDD)

DEFINE FILE 4(201,19,U,LDR)

DATA VE/12*0.,1.7,LDD,LDT,LDF/1,1,1/

LDR=1

NUMB=NBAR

VAUX(13)=1.

C-----PRIMEIRA PASSAGEM PELA ESTRUTURA

IND=JC(LDF)

READ(2,LDF)MF,DMF

CALL PEMF(DMF,MF,F,IND)

DO 400 I=1,6

K=I+6

F(K,K)=1.

IF(LRE(I))400,400,401

401

F(K,13)=0.

400

CONTINUE

405

IND=MC(LDT)

READ(1,LDT)MT,DMT,R,XL

CALL PEMG(XL,DMT,T,R,IND)

CALL PMPE(T,F,P)

IND=JC(LDF)

READ(2,LDF)MF,DMF

CALL PEMF(DMF,MF,T,IND)

IF(MF(1))406,406,407

407

NUMB=NUMB+1

IND=MC(NUMB)

LD=LDT

READ(1,NUMB)MT,DMT,R,XL

LDT=LD

CALL PEBR(XL,DMT,T,R,IND,SMR)

PAGE 2 PR-PEMT3

```
406 IF(LDT-NBAR)408,408,409
408 CALL PMMPE(T,P,F)
      GOTO 405
409 DO 410 I=1,6
      K=I+6
      T(K,K)=1.
      IF(LRD(I))402,402,411
402 T(K,13)=-T(K,13)
      T(K,I)=-T(K,I)
      GOTO 410
411 T(K,13)=0.
      T(I,13)=-T(I,13)
410 CONTINUE
      CALL PMMPE(T,P,F)
C-----CONDICOES DE CONTORNO
      CALL CAEPE(F,T,VE)
C-----RESOLUCAO DO SISTEMA RESULTANTE
      CALL PEINV(T)
      CALL PMVPE(T,VE,V)
C-----REARRANJAR VETOR DE ESTADO DA JUNTA 1
      DO 420 I=1,12
420 VE(I)=0.
      DO 430 I=1,6
      KA=KAC(I)
430 VE(KA)=-V(I)
      LDF=1
      IND=JC(LDF)
      READ(2,LDF)MF,DMF
      DO 440 I=1,6
440 LR(I)=LRE(I)
      CALL AEPE(DMF,IND,VE,K,VR,LR)
C-----DESLOCAMENTOS E REACOES (APOIOS ELASTICOS ) JUNTA 1
      DO 450 I=1,12
450 VAUX(I)=VE(I)
      IF( MF(2)+K )413,413,414
414 LD=LDF-1
      WRITE(4,LDR)LD,VR
413 WRITE(3,LDD)(VAUX(I),I=1,6)
      CALL LINK(PEMT4)
      END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR PEMT3
COMMON 714 VARIABLES 1840 PROGRAM 508

END OF COMPILATION

PAGE 1 PR-PEMT4

// JOB

PR-PEMT4

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPEMT4

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION OBRA(20),RESP(20),FULA(20),LCAB(4),LRE(6),LRD(6),VE(13)

DIMENSION TITUL(20),LEN(6),KAC(6),MF(3),MT(3),V(13),R(9),LR(6)

DIMENSION VR(6),F(13,13),T(13,13),DMT(20),DMF(27),AM(12),SMR(6,6)

DIMENSION MC(200),JC(201),VAUX(13),VD(6)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC

COMMON TITUL,LDR,VAUX

EQUIVALENCE (VE(1),VD(1))

DEFINE FILE 1(200,99,U,LDT),2(201,84,U,LDF),3(201,18,U,LDD)

DEFINE FILE 4(201,19,U,LDR)

DATA LDF,LDD,LDT/2,2,1/

C-----SEGUNDA PASSAGEM PELA ESTRUTURA - RESULTADOS

NUMB=NBAR

NJ=NBAR+1

DO 605 I=1,13

605 V(I)=VAUX(I)

C-----IMPRIMIR TITULO DO CARREGAMENTO

WRITE(W,505)TITUL

505 FORMAT(/1X'*RESULTADOS* ',20A4/1X,80('='))

WRITE(W,515)

515 FORMAT(/1X'ACOES NAS EXTREMIDADES DAS BARRAS'/)

WRITE(W,525)

525 FORMAT(1X'BARRA'13X'FX'10X'FY'10X'FZ'9X'MXX'9X'MYY'9X'MZZ')

501 IND=MC(LDT)

READ(1,LDT)MT,DMT,R,XL

CALL PEMG(XL,DMT,T,R,IND)

CALL PMVPE(T,V,VE)

DO 500 I=1,12

500 AM(I)=0.

K=0

DO 510 I=1,3

DO 510 J=7,9

K=K+1

AM(I)=AM(I)+V(J)*R(K)

AM(I+3)=AM(I+3)+V(J+3)*R(K)

AM(I+6)=AM(I+6)+VE(J)*R(K)

510 AM(I+9)=AM(I+9)+VE(J+3)*R(K)

LD=LDT-1

C-----ACOES DE EXTREMO DE BARRAS PRINCIPAIS

```

WRITE(W,535)LD,AM
535  FORMAT(1X,I5,1X'JJ',6F12.2/7X'JK',6F12.2/)
      IND=JC(LDF)
      READ(2'LDF)MF,DMF
      IF(LDT-NBAR)502,502,503
502  CALL PEMF(DMF,MF,F,IND)
      IF(MF(1))506,506,504
504  NUMB=NUMB+1
      LD=LDT
      IND=MC(NUMB)
      READ(1'NUMB)MT,DMT,R,XL
      CALL PEBR(XL,DMT,F,R,IND,SMR)
      IF(IND)507,507,508
507  DO 520 I=9,20
520  DMT(I)=0.

```

C-----ACOES DE EXTREMO DE RAMIFICACOES

```

508  DO 540 I=1,4
      K=I+8
540  AM(I)=DMT(K)-SMR(I,1)*VE(1)-SMR(I,2)*VE(2)-SMR(I,3)*VE(3)-SMR(I,4)
      *VE(4)-SMR(I,5)*VE(5)-SMR(I,6)*VE(6)
      DO 550 I=5,6
      K=I+8
550  AM(I)=DMT(K)+SMR(I,1)*VE(1)+SMR(I,2)*VE(2)+SMR(I,3)*VE(3)+SMR(I,4)
      *VE(4)/2.+SMR(I,5)*VE(5)/2.+SMR(I,6)*VE(6)/2.
      DO 530 I=1,6
      J=I+6
      K=I+14
530  AM(J)=DMT(K)+SMR(I,1)*VE(1)+SMR(I,2)*VE(2)+SMR(I,3)*VE(3)+SMR(I,4)
      *VE(4)+SMR(I,5)*VE(5)+SMR(I,6)*VE(6)
      WRITE(W,535)NUMB,AM
      DO 560 I=1,3
      J=I+3
      K=I+6
      V(I)=R(I)*AM(1)+R(J)*AM(2)+R(K)*AM(3)
560  V(J)=R(I)*AM(4)+R(J)*AM(5)+R(K)*AM(6)
      K=NUMB+1
      WRITE(4'LDR)K,(V(I),I=1,6)
      LDT=LD
506  CALL PMVPE(F,VE,V)
      WRITE(3'LDD)VD
      IF(MF(2))501,501,509
509  DO 570 I=1,6
      J=I+3
570  VR(I)=-DMF(J)*V(I)
      WRITE(4'LDR)LDT,VR
      GOTO 501
503  DO 580 I=1,6
580  LR(I)=LRD(I)
      CALL AEPE(DMF,IND,VE,K,VR,LR)
      WRITE(3'LDD)VD

```

PAGE 3 PR-PEMT4

```
IF(MF(2)+K)512,512,513
513 WRITE(4,LDR)LDT,VR
512 LD=LDR-1
IF(LD)516,516,517
517 WRITE(W,545)
545 FORMAT(/,1X'REACOE'/)
WRITE(W,555)
555 FORMAT(1X'JUNTA'10X'RX'10X'RY'10X'RZ'9X'RX'9X'RY'9X'RZ')
DO 590 I=1,LD
READ(4,I)K,VR
590 WRITE(W,565)K,VR
565 FORMAT(1X,I5,6F12.2)
516 WRITE(W,575)
575 FORMAT(/,1X'DESLOCAMENTOS GENERALIZADOS'/)
WRITE(W,585)
585 FORMAT(1X'JUNTA'10X'DX'10X'DY'10X'DZ'9X'DX'9X'DY'9X'DZ')
DO 600 I=1,NJ
READ(3,I)VR
600 WRITE(W,595)I,VR
595 FORMAT(1X,I5,6F12.5)
CALL LINK(PEMT5)
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION
IOCS

CORE REQUIREMENTS FOR PEMT4
COMMON 714 VARIABLES 1504 PROGRAM 1304

END OF COMPILATION

PAGE 1 PR-PEMT5

// JOB

PR-PEMT5

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR ELDON LONDE MELLO - COPPE-UFRJ - PROG. ENG. CIVIL

*NAMEPEMT5

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

*IOCS(CARD,2501READER,1132PRINTER,1403PRINTER,DISK)

INTEGER RE,W

DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20),KOM(4),LRE(6),LRD(6)

DIMENSION INP7(12)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD

DEFINE FILE 1(200,99,U,LDT),2(201,84,U,LDF),3(201,18,U,LDD)

DEFINE FILE 4(201,19,U,LDR)

DATA INP7('/', '*', ',', '/', 'E', 'S', 'T', '/', 'F', 'I', 'M', '/

C-----LER CODIGO DE DECISAO

READ(RE,601)KOM

601 FORMAT(4A1)

I=0

DO 610 J=1,9,4

K=J+3

I=I+1

IF(ITEST(INP7,J,K,KOM,1))610,602,610

602 GOTO(605,615,625),I

610 CONTINUE

C-----FINALIZAR PROCESSAMENTO

WRITE(W,603)KOM

603 FORMAT(/5X'CODIGO ('4A1') NAO IDENTIFICADO')

GOTO 604

625 WRITE(W,606)KOM

606 FORMAT(/1X,4A1)

604 CALL CABP(2)

WRITE(W,607)

607 FORMAT(/1X'PROCESSAMENTO CONCLUIDO'//1X'HR. MAQUINA INICIAL='//1X

'HR. MAQUINA FINAL ='//1X'TEMPO PROCESSAMENTO= HRS. MIN

*.'//)

WRITE(W,608)

608 FORMAT('1')

CALL EXIT

C-----PROCESSAR NOVO CARREGAMENTO

605 CALL LINK(PEMT2)

C-----PROCESSAR NOVA ESTRUTURA

615 CALL LINK(PEMT1)

END

FEATURES SUPPORTED

PAGE 1 SR-PEMT

// JOB

SR-PEMT

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR SR-CABP - CABECALHO - PPMT E PEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE CABP(NE)

INTEGER RE,W

DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20)

COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP

GOTO(11,12),NE

11 WRITE(W,14)

14 FORMAT('1'//5X'ANALISE PORTICO PLANO M. MATRIZES TRANSFERENCIA'//)

GOTO 15

12 WRITE(W,16)

16 FORMAT('1'//5X'ANALISE PORTICO NO ESPACO M. MATRIZES TRANSFERENCIA

*'//)

15 DO 100 I=1,4

IF(LCAB(I)-1)102,102,100

102 GOTO(105,115,125,135),I

105 WRITE(W,105)

106 FORMAT(5X'ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO'//)

GOTO 100

115 WRITE(W,107)OBRA

107 FORMAT(5X,20A4)

GOTO 100

125 WRITE(W,107)FULA

GOTO 100

135 WRITE(W,107)RESP

100 CONTINUE

WRITE(W,111)

111 FORMAT(1X,40('='))

RETURN

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR CABP

COMMON 188 VARIABLES 2 PROGRAM 198

RELATIVE ENTRY POINT ADDRESS IS 006E (HEX)

END OF COMPILATION

PAGE 2 SR-PEMT

// FOR SR-PEJL - LIBERACOES GENERALIZADAS - JUNTAS INTERNAS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE PEJL(DD,NV,N)
      DIMENSION DD(15),DMF(27),MF(3)
      DO 100 I=1,N
      J=DD(I)
      READ(2,J)MF,DMF
      IF(MF(3))11,11,12
11      DO 110 K=10,15
110     DMF(K)=0.
      MF(3)=1
12      DMF(NV)=1.E+15
100     WRITE(2,J)MF,DMF
      RETURN
      END
```

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR PEJL

COMMON 0 VARIABLES 88 PROGRAM 106

RELATIVE ENTRY POINT ADDRESS IS 0062 (HEX)

END OF COMPILATION

// FOR SR-PEJR - JUNTAS COM RAMIFICACOES - PPMT E PEMT

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
      SUBROUTINE PEJR(DD,ID,N)
      DIMENSION DD(15),MF(3)
      IDD=ID/8
      IF(IDD-2056)101,102,101
101     ID=32320
      N1=DD(1)
      N2=DD(2)
      N3=DD(3)
      IF(N-2)102,103,104
103     N3=1
104     DO 110 I=N1,N2,N3
      READ(2,I)MF
      MF(1)=1
110     WRITE(2,I)MF
      RETURN
102     DO 100 I=1,N
      J=DD(I)
      READ(2,J)MF
```

PAGE 3 SR-PEMT

```
MF(1)=1
100 WRITE(2,'J')MF
RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEJR
COMMON 0 VARIABLES 12 PROGRAM 148

RELATIVE ENTRY POINT ADDRESS IS 0011 (HEX)

END OF COMPILATION

// FOR SR-PEDJ - DADOS DAS JUNTAS - COORD. E RIGIDEZES DE MOLAS

```
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE PEDJ(DT,DI,NV,INC)
DIMENSION DT(6),DI(2),MF(3),DMF(27)
N1=DI(1)
N2=DI(2)
K=0
DO 100 I=N1,N2
K=1+K*INC
READ(2,'I')MF,DMF
IF(NV-3)11,11,12
12 MF(2)=1
11 DMF(NV)=DT(K)
100 WRITE(2,'I')MF,DMF
INC=K
RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEDJ
COMMON 0 VARIABLES 90 PROGRAM 112

RELATIVE ENTRY POINT ADDRESS IS 005E (HEX)

END OF COMPILATION

// FOR SR-PEDB - DADOS DAS BARRAS - PROP. PRISMATICAS E CONST.

```
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
```

```

SUBROUTINE PEDB(DT,DI,I,INC)
DIMENSION DT(6),DI(2),MT(3),DMT(20)
N1=DI(1)
N2=DI(2)
K=0
NV=1-9
DO 100 J=N1,N2
K=1+K*INC
READ(1,J)MT,DMT
IF(NV=6)11,11,12
11 MT(1)=1
GOTO 13
12 IF(NV=8)14,14,16
14 MT(2)=1
GOTO 13
16 MT(3)=1
13 DMT(NV)=DT(K)
WRITE(1,J)MT,DMT
100 CONTINUE
INC=K
RETURN
END

```

FEATURES SUPPORTED
 ONE WORD INTEGERS
 EXTENDED PRECISION

CORE REQUIREMENTS FOR PEDB
 COMMON 0 VARIABLES 70 PROGRAM 140

RELATIVE ENTRY POINT ADDRESS IS 004B (HEX)

END OF COMPILATION

// FOR SR-PEVR - VETOR ROTACAO

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```

SUBROUTINE PEVR(IND,CX,CY,CZ,DXP,DYP,DZP,R,DM)
DIMENSION R(9),DM(3)
DM(2)=0.
DO 100 I=1,9
R(I)=0.
DM(1)=SQRT(CX*CX+CY*CY+CZ*CZ)
CX=CX/DM(1)
CY=CY/DM(1)
CZ=CZ/DM(1)
Q=SQRT(CX*CX+CZ*CZ)
IF(Q=0.001)101,101,102
101 R(2)=CY
R(4)=-CY

```

PAGE 5 SR-PEMT

```
103 R(9)=1
    IF(IND)105,105,103
    SQ=SQRT(DXP*DXP+DZP*DZP)
    COSA=-DXP*CY/SQ
    SINA=DZP/SQ
    R(4)=-CY*COSA
    R(6)=SINA
    R(7)=CY*SINA
    R(9)=COSA
102 GOTO 106
    R(1)=CX
    R(2)=CY
    R(3)=CZ
    R(4)=-CX*CY/Q
    R(5)=Q
    R(6)=-CY*CZ/Q
    R(7)=-CZ/Q
    R(9)=CX/Q
104 IF(IND)105,105,104
    YPG=R(4)*DXP+R(5)*DYP+R(6)*DZP
    ZPG=R(7)*DXP+R(8)*DYP+R(9)*DZP
    SQ=SQRT(YPG*YPG+ZPG*ZPG)
    COSA=YPG/SQ
    SINA=ZPG/SQ
    R(4)=(-CX*CY*COSA-CZ*SINA)/Q
    R(5)=Q*COSA
    R(6)=(-CY*CZ*COSA+CX*SINA)/Q
    R(7)=(CX*CY*SINA-CZ*COSA)/Q
    R(8)=-Q*SINA
    R(9)=(CY*CZ*SINA+CX*COSA)/Q
106 DM(2)=ATAN(SINA/COSA)*57.2957795
105 RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEVR
COMMON 0 VARIABLES 36 PROGRAM 582

RELATIVE ENTRY POINT ADDRESS IS 0032 (HEX)

END OF COMPILE

// FOR SR-PEBC - CARGAS NAS BARRAS - ROTINA PRINCIPAL
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE PEBC(DT,ID,DI,I,KOM,INC)
INTEGER RE,W


```

      DIMENSION DT(6),DI(2),DMT(20),KOM(4),LCAB(4),OBRA(20),FULA(20)
      DIMENSION RESP(20),LRE(6),LRD(6),LEN(6),KAC(6),MC(200),MT(3),R(9)
      COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC
      N1=DI(1)
      N2=DI(2)
      K=0
      IPR=1
      IF((N2-N1-2)*INC)101,101,102
102  IF(I-4)101,101,103
101  DO 100 J=N1,N2
      IF(J-NBAR)104,104,106
106  IPR=2
104  K=1+K*INC
      READ(1,J)MT,DMT,R,XL
      DO 110 N=1,6
      IF(DMT(N))121,121,110
110  CONTINUE
      IF(MC(J))107,107,108
107  DO 120 N=9,20
120  DMT(N)=0.
      MC(J)=1
108  A=DT(K)
      B=A
C-----CARGAS DISTRIBUIDAS LINEARES
      GOTO(1,1,1,2,3,3,3,4,4,4,5,5,5,6),I
1    CALL PELIN(DMT,A,B,IPR,I,XL)
      GOTO 109
C-----MOMENTO UNIFORME SEGUNDO X
2    CALL PEUXX(DMT,A,IPR,XL)
      GOTO 109
3    K=K+1
      B=DT(K)
      I=I-4
      GOTO 1
C-----CARGAS CONCENTRADAS
4    CALL PEFCB(DMT,DT,K,IPR,I,XL,INC,N1,N2)
      GOTO 109
C-----MOMENTOS APLICADOS NAS BARRAS
5    CALL PEMCB(DMT,DT,K,IPR,I,XL,INC,N1,N2)
      GOTO 109
6    K=K+1
      B=DT(K)
C-----VARIACAO UNIFORME DE TEMPERATURA
      CALL PETUX(DMT,A,B,IPR,XL)
109  WRITE(1,J)MT,DMT
100  CONTINUE
      IF(I-14)111,112,112
111  WRITE(W,113)KOM,ID,DI,(DT(J),J=1,K)
113  FORMAT(1X,5A1,2F5.0,6F10.2)
      GOTO 114
112  WRITE(W,116)KOM,ID,DI,(DT(J),J=1,K)

```

PAGE 7 SR-PEMT

```
116 FORMAT(1X,5A1,2F5.0,3(F10.2,F10.6))
114 RETURN
103 IER=5
WRITE(W,131)KOM,ID,DI,IER
131 FORMAT(/75X,5A1,2F5.0//75X'ERRO='I5)
CALL EXIT
121 IER=6
WRITE(W,122)J,IER
122 FORMAT(/5X'BARRA='I5,3X'ERRO='I5)
CALL EXIT
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEBC
COMMON 412 VARIABLES 108 PROGRAM 482

RELATIVE ENTRY POINT ADDRESS IS 00A9 (HEX)

END OF COMPILATION

// FOR SR-PELIN - CARGAS DISTRIBUIDAS LINEARES

*ONE WORD INTEGERS
*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE PELIN(DMT,A,B,IPR,I,XL)

DIMENSION DMT(20)

AUX=XL*XL

GOTO(11,12),IPR

C-----BARRAS PRINCIPAIS

11 IF(I-2)13,14,16

C-----SEGUNDO X

13 DMT(9)=DMT(9)-(2.*A+B)*AUX/DMT(5)/DMT(1)/6.

DMT(15)=DMT(15)-(A+B)*XL/2.

RETURN

C-----SEGUNDO Y

14 DMT(10)=DMT(10)+(4.*A+B)*AUX*AUX/DMT(5)/DMT(4)/120.

DMT(14)=DMT(14)+(3.*A+B)*AUX*XL/DMT(5)/DMT(4)/24.

DMT(16)=DMT(16)-(A+B)*XL/2.

DMT(20)=DMT(20)+(2.*A+B)*AUX/6.

RETURN

C-----SEGUNDO Z

16 DMT(11)=DMT(11)+(4.*A+B)*AUX*AUX/DMT(5)/DMT(3)/120.

DMT(13)=DMT(13)-(3.*A+B)*AUX*XL/DMT(5)/DMT(3)/24.

DMT(17)=DMT(17)-(A+B)*XL/2.

DMT(19)=DMT(19)-(2.*A+B)*AUX/6.

RETURN

C-----RAMIFICACOES

12 IF(I-2)17,18,19

PAGE 8 SR-PEMT

```
C-----SEGUNDO X
17 DMT(9)=DMT(9)-(2.*A+B)*XL/6.
   DMT(15)=DMT(15)-(A+2.*B)*XL/6.
   RETURN
C-----SEGUNDO Y
18 DMT(10)=DMT(10)-(7.*A+3.*B)*XL/20.
   DMT(14)=DMT(14)-(3.*A+2.*B)*AUX/60.
   DMT(16)=DMT(16)-(3.*A+7.*B)*XL/20.
   DMT(20)=DMT(20)+(2.*A+3.*B)*AUX/60.
   RETURN
C-----SEGUNDO Z
19 DMT(11)=DMT(11)-(7.*A+3.*B)*XL/20.
   DMT(13)=DMT(13)+(3.*A+2.*B)*AUX/60.
   DMT(17)=DMT(17)-(3.*A+7.*B)*XL/20.
   DMT(19)=DMT(19)-(2.*A+3.*B)*AUX/60.
   RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PELIN
COMMON 0 VARIABLES 12 PROGRAM 594

RELATIVE ENTRY POINT ADDRESS IS 0028 (HEX)

END OF COMPILE

// FOR SR-PEUXX - MOMENTO UNIFORME SEGUNDO X

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE PEUXX(DMT,A,IPR,XL)

DIMENSION DMT(20)

A=XL*A

GOTO(11,12),IPR

C-----BARRAS PRINCIPAIS

11 DMT(18)=DMT(18)-A

DMT(12)=DMT(12)-XL*A/DMT(6)/DMT(2)/2.

RETURN

C-----RAMIFICACOES

12 A=A/2.

DMT(12)=DMT(12)-A

DMT(18)=DMT(18)-A

RETURN

END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEUXX
COMMON 0 VARIABLES 4 PROGRAM 98

RELATIVE ENTRY POINT ADDRESS IS 0007 (HEX)

END OF COMPILATION

// FOR SR-PEFCB - CARGAS CONCENTRADAS NAS BARRAS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE PEFCB(DMT,DT,K,IPR,I,XL,INC,N1,N2)

DIMENSION DMT(20),DT(6)

NV=1

J=K-1

IF((N2-N1)*INC)101,101,102

101 NV=3

102 GOTO(11,12),IPR

C-----BARRAS PRINCIPAIS

11 DO 100 N=1,NV

J=J+1

IF(DT(J))104,106,104

104 P=DT(J)

J=J+1

A=XL-DT(J)

B=A*A

IF(I-9)13,14,16

C-----SEGUNDO X

13 DMT(9)=DMT(9)-P*A/DMT(5)/DMT(1)

DMT(15)=DMT(15)-P

GOTO 111

C-----SEGUNDO Y

14 DMT(10)=DMT(10)+P*A*B/DMT(5)/DMT(4)/6.

DMT(14)=DMT(14)+P*B/DMT(5)/DMT(4)/2.

DMT(16)=DMT(16)-P

DMT(20)=DMT(20)+P*A

GOTO 111

C-----SEGUNDO Z

16 DMT(11)=DMT(11)+P*A*B/DMT(5)/DMT(3)/6.

DMT(13)=DMT(13)-P*B/DMT(5)/DMT(3)/2.

DMT(17)=DMT(17)-P

DMT(19)=DMT(19)-P*A

111 K=J

100 CONTINUE

106 RETURN

C-----RAMIFICACOES

12 AUX=XL*XL

DO 200 N=1,NV

J=J+1

IF(DT(J))107,108,107

PAGE 10 SR-PEMT

```
107 P=DT(J)
    J=J+1
    A=DT(J)
    B=XL-A
    IF(I-9)17,18,19
C-----SEGUNDO X
17 DMT(9)=DMT(9)-P*B/XL
   DMT(15)=DMT(15)-P*A/XL
   GOTO 112
C-----SEGUNDO Y
18 DMT(10)=DMT(10)-P*B*B*(3.*A+B)/AUX/XL
   DMT(14)=DMT(14)-P*A*B*B/AUX
   DMT(16)=DMT(16)-P*A*A*(A+3.*B)/AUX/XL
   DMT(20)=DMT(20)+P*A*A*B/AUX
   GOTO 112
C-----SEGUNDO Z
19 DMT(11)=DMT(11)-P*B*B*(3.*A+B)/AUX/XL
   DMT(13)=DMT(13)+P*A*B*B/AUX
   DMT(17)=DMT(17)-P*A*A*(A+3.*B)/AUX/XL
   DMT(19)=DMT(19)-P*A*A*B/AUX

112 K=J
200 CONTINUE
108 RETURN
    END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEFCB
COMMON 0 VARIABLES 24 PROGRAM 628

RELATIVE ENTRY POINT ADDRESS IS 0024 (HEX)

END OF COMPILATION

```
// FOR SR-PEMCB - MOMENTOS APLICADOS NAS BARRAS
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE PEMCB(DMT,DT,K,IPR,I,XL,INC,N1,N2)
  DIMENSION DMT(20),DT(6)
  NV=1
  J=K-1
  IF((N2-N1)*INC)101,101,102
101 NV=3
102 GOTO(11,12),IPR
C-----BARRAS PRINCIPAIS
11 DO 100 N=1,NV
   J=J+1
   IF(DT(J))104,106,104
```

```

104  XM=DT(J)
      J=J+1
      A=XL-DT(J)
      B=A*A
      IF(I-12)13,14,16
C-----SEGUNDO  X
13    DMT(18)=DMT(18)-XM
      DMT(12)=DMT(12)-XM*A/DMT(6)/DMT(2)
      GOTO 111
C-----SEGUNDO  Y
14    DMT(11)=DMT(11)-XM*B/DMT(5)/DMT(3)/2.
      DMT(13)=DMT(13)-XM*A/DMT(5)/DMT(3)
      DMT(19)=DMT(19)-XM
      GOTO 111
C-----SEGUNDO  Z
16    DMT(10)=DMT(10)-XM*B/DMT(5)/DMT(4)/2.
      DMT(14)=DMT(14)-XM*A/DMT(5)/DMT(4)
      DMT(20)=DMT(20)-XM
111   K=J
100   CONTINUE
106   RETURN
C-----RAMIFICACOES
12    AUX=XL*XL
      DO 200 N=1,NV
      J=J+1
      IF(DT(J))107,108,107
107   XM=DT(J)
      J=J+1
      A=DT(J)
      B=XL-A
      IF(I-12)17,18,19
C-----SEGUNDO  X
17    DMT(12)=DMT(12)-XM*B/XL
      DMT(18)=DMT(18)-XM*A/XL
      GOTO 112
C-----SEGUNDO  Y
18    DMT(11)=DMT(11)-6.*XM*A*B/AUX/XL
      DMT(13)=DMT(13)+XM*B*(2.*A-B)/AUX
      DMT(17)=DMT(17)+6.*XM*A*B/AUX/XL
      DMT(19)=DMT(19)+XM*A*(2.*B-A)/AUX
      GOTO 112
C-----SEGUNDO  Z
19    DMT(10)=DMT(10)+6.*XM*A*B/AUX/XL
      DMT(14)=DMT(14)+XM*B*(2.*A-B)/AUX
      DMT(16)=DMT(16)-6.*XM*A*B/AUX/XL
      DMT(20)=DMT(20)+XM*A*(2.*B-A)/AUX
112   K=J
200   CONTINUE
108   RETURN
      END

```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEMCB
COMMON 0 VARIABLES 24 PROGRAM 586

RELATIVE ENTRY POINT ADDRESS IS 0021 (HEX)

END OF COMPILATION

// FOR SR-PETUX - VARIACAO UNIFORME DE TEMPERATURA

*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM

SUBROUTINE PETUX(DMT,A,B,IPR,XL)
DIMENSION DMT(20)
GOTO(11,12),IPR

C-----BARRAS PRINCIPAIS

11 DMT(9)=DMT(9)+A*B*XL
RETURN

C-----RAMIFICACOES

12 DMT(9)=DMT(9)+A*B*DMT(5)*DMT(1)
DMT(15)=DMT(15)-A*B*DMT(5)*DMT(1)
RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PETUX
COMMON 0 VARIABLES 4 PROGRAM 84

RELATIVE ENTRY POINT ADDRESS IS 0004 (HEX)

END OF COMPILATION

// FOR SR-PEJC - DESLOCAMENTOS IMPOSTOS E CARGAS NAS JUNTAS

*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM

SUBROUTINE PEJC(DT,DI,NV,INC)
INTEGER RE,W

DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20),LRE(6),LRD(6),LEN(6)
DIMENSION KAC(6),DT(6),DI(2),MF(3),DMF(27),MC(200),JC(201)
COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC,MC,JC
N1=DI(1)
N2=DI(2)
K=0

PAGE 13 SR-PEMT

```
DO 100 J=N1,N2
K=1+K*INC
READ(2,J)MF,DMF
IF(JC(J))11,11,12
11 DO 110 I=16,27
110 DMF(I)=0.
JC(J)=1
12 DMF(NV)=DT(K)+DMF(NV)
WRITE(2,J)MF,DMF
100 CONTINUE
INC=K
RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEJC
COMMON 614 VARIABLES 92 PROGRAM 144

RELATIVE ENTRY POINT ADDRESS IS 0064 (HEX)

END OF COMPILATION

```
// FOR SR-PEMF - MATRIZ FRONTEIRA
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE PEMF(DMF,MF,T,IND)
DIMENSION MF(3),DMF(27),T(13,13)
DO 10 J=1,13
DO 10 K=1,13
T(J,K)=0.
10 T(J,J)=1.
DO 20 J=7,12
20 T(J,J)=-1.
IF(MF(2))11,11,12
12 DO 30 J=1,6
K=J+6
30 T(K,J)=-DMF(J+3)
11 IF(MF(3))13,13,14
14 DO 40 J=1,6
K=J+6
40 T(J,K)=DMF(J+9)
13 IF(IND)16,16,17
17 DO 50 J=1,12
K=J+15
50 T(J,13)=DMF(K)
16 RETURN
END
```


FEATURES SUPPORTED
 ONE WORD INTEGERS
 EXTENDED PRECISION

CORE REQUIREMENTS FOR PEMF
 COMMON 0 VARIABLES 4 PROGRAM 232

RELATIVE ENTRY POINT ADDRESS IS 0010 (HEX)

END OF COMPILATION

// FOR SR-PEMG - MATRIZ TRANSF. - S. LOCAL E GLOBAL

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE PEMG(XL,DMT,T,R,IND)
 DIMENSION DMT(20),R(9),T(13,13),A(10)
 C-----MATRIZ TRANSFERENCIA SISTEMA LOCAL

DO 10 J=1,13
 DO 10 K=1,13
 10 T(J,K)=0.
 T(13,13)=1.
 A(1)=-XL/DMT(1)/DMT(5)
 A(3)=-XL/DMT(3)/DMT(5)
 A(4)=-XL/DMT(4)/DMT(5)
 A(2)=-XL/DMT(2)/DMT(6)
 A(5)=XL*A(4)/2.
 A(6)=-XL*A(5)/3.
 A(7)=XL*A(3)/2.
 A(9)=-XL*A(7)/3.
 A(8)=-A(7)
 A(10)=-A(5)

C-----SISTEMA GLOBAL
 T(1,5)=(R(4)*R(8)-R(5)*R(7))*XL
 T(1,6)=(R(4)*R(9)-R(6)*R(7))*XL
 T(2,6)=(R(5)*R(9)-R(6)*R(8))*XL
 T(2,4)=-T(1,5)
 T(3,4)=-T(1,6)
 T(3,5)=-T(2,6)
 T(10,8)=-T(1,5)
 T(10,9)=-T(1,6)
 T(11,7)=-T(10,8)
 T(11,9)=-T(2,6)
 T(12,7)=-T(10,9)
 T(12,8)=-T(11,9)
 DO 20 J=1,3
 JA=J+3
 JB=J+6
 JC=J+9
 DO 20 K=1,3

PAGE 15 SR-PEMT

```
KA=K+3
KB=K+6
KC=K+9
T(J,K)=R(J)*R(K)+R(JA)*R(KA)+R(JB)*R(KB)
T(J,KB)=R(J)*R(K)*A(1)+R(KA)*R(JA)*A(6)+R(JB)*R(KB)*A(9)
T(J,KC)=R(JA)*R(KB)*A(5)+R(KA)*R(JB)*A(8)
T(JA,KB)=R(JA)*R(KB)*A(7)+R(KA)*R(JB)*A(10)
T(JA,KC)=R(J)*R(K)*A(2)+R(JA)*R(KA)*A(3)+R(JB)*R(KB)*A(4)
T(JA,KA)=T(J,K)
T(JB,KB)=-T(J,K)
20 T(JC,KC)=-T(J,K)
IF(IND)11,11,12
C-----VETOR DE CARGAS P/ SISTEMA GLOBAL
12 DO 30 K=1,3
KA=K+3
KB=K+6
KC=K+9
T(K,13)=R(K)*DMT(9)+R(KA)*DMT(10)+R(KB)*DMT(11)
T(KA,13)=R(K)*DMT(12)+R(KA)*DMT(13)+R(KB)*DMT(14)
T(KB,13)=R(K)*DMT(15)+R(KA)*DMT(16)+R(KB)*DMT(17)
30 T(KC,13)=R(K)*DMT(18)+R(KA)*DMT(19)+R(KB)*DMT(20)
11 RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEMG
COMMON 0 VARIABLES 60 PROGRAM 1084

RELATIVE ENTRY POINT ADDRESS IS 004D (HEX)

END OF COMPILATION

// FOR SR-PEBR - MATRIZ RIGIDEZ DA RAMIFICACAO

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE PEBR(XL,DMT,T,R,IND,A)
DIMENSION DMT(20),T(13,13),R(9),A(6,6),B(6,6)
C-----SISTEMA LOCAL
DO 10 J=1,6
DO 10 K=1,6
10 B(J,K)=0.
B(1,1)=DMT(5)*DMT(1)/XL
B(4,4)=DMT(6)*DMT(2)/XL
R(5,5)=4.*DMT(5)*DMT(3)/XL
B(5,3)=1.5*B(5,5)/XL
B(3,3)=2.*B(5,3)/XL
B(6,6)=4.*DMT(5)*DMT(4)/XL
```

PAGE 16 SR-PEMT

```
B{2,6}=-1.5*B(6,6)/XL
B{6,2}=B{2,6}
B(2,2)=-2.*B(2,6)/XL
B(3,5)=B(5,3)
C-----SISTEMA GLOBAL
DO 20 J=1,2
  JC=3*J
  JB=JC-1
  JA=JC-2
  DO 20 K=1,6
    A(K,JA)=B(K,JA)*R(1)+B(K,JB)*R(4)+B(K,JC)*R(7)
    A(K,JB)=B(K,JA)*R(2)+B(K,JB)*R(5)+B(K,JC)*R(8)
20    A(K,JC)=B(K,JA)*R(3)+B(K,JB)*R(6)+B(K,JC)*R(9)
  DO 30 J=1,2
    JC=3*J
    JB=JC-1
    JA=JC-2
    IC=JC+6
    IB=JB+6
    IA=JA+6
    DO 30 K=1,6
      T(IA,K)=T(IA,K)-A(JA,K)*R(1)-A(JB,K)*R(4)-A(JC,K)*R(7)
      T(IB,K)=T(IB,K)-A(JA,K)*R(2)-A(JB,K)*R(5)-A(JC,K)*R(8)
30    T(IC,K)=T(IC,K)-A(JA,K)*R(3)-A(JB,K)*R(6)-A(JC,K)*R(9)
  IF(IND)11,11,12
C-----ACOES EQUIVALENTES NAS JUNTAS
12 DO 40 J=1,3
  JA=J+3
  JB=J+6
  JC=J+9
  T(JB,13)=T(JB,13)-R(J)*DMT(15)-R(JA)*DMT(16)-R(JB)*DMT(17)
40  T(JC,13)=T(JC,13)-R(J)*DMT(18)-R(JA)*DMT(19)-R(JB)*DMT(20)
11 RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR PEBR
COMMON 0 VARIABLES 138 PROGRAM 780

RELATIVE ENTRY POINT ADDRESS IS 009B (HEX)

END OF COMPILATION

// FOR SR-CAEPE - CONDICAOES DE CONTORNO
*ONE WORD INTEGERS
*EXTENDED PRECISION
*LIST SOURCE PROGRAM
SUBROUTINE CAEPE(F,A,VE)

PAGE 17 SR-PEMT

```
INTEGER RE,W
DIMENSION LCAB(4),OBRA(20),FULA(20),RESP(20),LRE(6),LRD(6)
DIMENSION F(13,13),A(13,13),VE(13),LEN(6),KAC(6)
COMMON RE,W,NBAR,NRAM,LCAB,OBRA,FULA,RESP,LRE,LRD,LEN,KAC
DO 100 I=1,6
  LEN(I)=I
  KAC(I)=I
  IF(LRE(I))101,101,102
102  KAC(I)=I+6
101  IF(LRD(I))103,103,100
103  LEN(I)=I+6
100  CONTINUE
  DO 110 I=1,6
    LE=LEN(I)
    VE(I)=F(LE,13)
    DO 120 J=1,6
      KA=KAC(J)
120  A(I,J)=F(LE,KA)
110  CONTINUE
  RETURN
END
```

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR CAEPE
COMMON 212 VARIABLES 8 PROGRAM 160

RELATIVE ENTRY POINT ADDRESS IS 000A (HEX)

END OF COMPILE

// FOR SR-AEPE - REACOES NOS APOIOS EXTREMOS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

```
SUBROUTINE AEPE(DMF,IND,V,IR,VR,LR)
DIMENSION DMF(27),V(13),VR(6),LR(6)
IR=0
IF(IND)101,101,102
101 DO 110 I=16,27
110 DMF(I)=0.
102 DO 120 I=1,6
  J=I+6
  K=I+21
  IF(LR(I))103,103,104
104 VR(I)=V(J)-DMF(K)
  V(I)=DMF(I+15)
  IR=1
GOTO 120
```

PAGE 18 SR-PEMT

103 VR(I)=-DMF(I+3)*V(I)
120 CONTINUE
RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR AEPE
COMMON 0 VARIABLES 8 PROGRAM 160

RELATIVE ENTRY POINT ADDRESS IS 0011 (HEX)

END OF COMPILATION

PAGE 1 SR-GERAL

// JOB

SR-GERAL

LOG DRIVE CART SPEC CART AVAIL PHY DRIVE
0000 0005 0005 0000

V2 M09 ACTUAL 8K CONFIG 8K

// FOR FT-ITEST - IDENTIFICACAO DOS CODIGOS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

FUNCTION ITEST(JCARD,J,JLAST,KCARD,K)

DIMENSION JCARD(80), KCARD(4)

DO 2 JNOW=J,JLAST

KNOW=K+JNOW-J

ITEST=KCARD(KNOW)/8

ITEST=JCARD(JNOW)/8-ITEST

IF(ITEST)1,2,1

2 CONTINUE

1 RETURN

END

FEATURES SUPPORTED

ONE WORD INTEGERS

EXTENDED PRECISION

CORE REQUIREMENTS FOR ITEST

COMMON 0 VARIABLES 6 PROGRAM 70

RELATIVE ENTRY POINT ADDRESS IS 0007 (HEX)

END OF COMPILATION

// FOR SR-CONCA - CONSISTENCIA DE DADOS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE CONCA(DD,NCAMP,MAX,IER,ICOD,N)

DIMENSION DD(15)

IER=0

N=0

DO 100 K=1,NCAMP

IF(ABS(DD(K))-10000.)101,102,102

101 J=DD(K)

IF(J)103,104,106

106 IF(J-MAX)107,107,103

107 N=N+1

100 CONTINUE

104 IF(N)103,103,108

103 IER=ICOD

RETURN

PAGE 2 SR-GERAL

102 IER=1
108 RETURN
END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR CONCA
COMMON 0 VARIABLES 4 PROGRAM 110

RELATIVE ENTRY POINT ADDRESS IS 0009 (HEX)

END OF COMPILATION

// FOR SR-CONCB - CONSISTENCIA DE DADOS

*ONE WORD INTEGERS

*EXTENDED PRECISION

*LIST SOURCE PROGRAM

SUBROUTINE CONCB(DI,ID,INC,IER,MAX,ICOD)

DIMENSION DI(2)

IER=0

INC=0

IDD=ID/8

DO 100 K=1,2

IF (ABS(DI(K))-10000.) 101,102,102

101 J=DI(K)

IF(J) 103,103,104

104 IF(J-MAX) 100,100,103

103 IER=ICOD

RETURN

102 IER=1

RETURN

100 CONTINUE

J=DI(2)-DI(1)

IF (IDD-2056) 106,107,106

107 INC=1

IF(J-5) 108,108,109

109 IER=5

RETURN

106 ID=32320

108 RETURN

END

FEATURES SUPPORTED
ONE WORD INTEGERS
EXTENDED PRECISION

CORE REQUIREMENTS FOR CONCB
COMMON 0 VARIABLES 6 PROGRAM 152

8.5 - Código de Erros

Os códigos de erros previstos nos programas são os abaixo indicados:

Se qualquer deles for detectado, o programa encerra o processamento e devolve o controle ao supervisor.

As mensagens de erro são impressas no próprio formulário.

ERRO	MOTIVO
1	Falta o ponto decimal num campo destinado a <u>números</u> inteiros
2	Número de barras omisso, negativo ou nulo
3	Barra designada por um número maior que o <u>especificado</u>
4	Junta designada por um número maior que o <u>especificado</u>
5	O intervalo N1 - N2 é muito amplo, não havendo espaço para a transmissão dos dados num só cartão. Reduzir o intervalo N1 - N2
6	Ou AX, IX, IY, IZ, E, G são negativos ou nulos, ou o comprimento L das barras (fornecido ou calculado) é negativo ou nulo
7	N1 e/ou N2 de um código /CAR ou *CAR são negativos ou nulos.

A P Ê N D I C E

A - Notações utilizadas no desenvolvimento teórico.

$ T _i$	- matriz de transferência da barra i
$ F _i$	- matriz fronteira da junta i
$ t _i$	- matriz de transferência reduzida do vão i
$ f _i$	- matriz fronteira reduzida da junta i
$\{V\}_{j,i}$	- vetor de estado na extremidade j da barra i
$\{v\}_{j,i}$	- vetor de estado reduzido na extremidade j da barra i
$\{D\}$	- vetor deslocamento do vetor de estado
$\{A\}$	- vetor ações do vetor de estado
$\{VF\}_i$	- vetor fronteira da junta i
$ S $	- matriz de rigidez
$ R $	- matriz rotação
x, y, z	- eixos do sistema local
X, Y, Z	- eixos do sistema global
x_j, y_j, z_j	- translações na junta j
ϕ_j	- rotação de torção na extremidade j
θ_j	- rotação de flexão na extremidade j
N_j	- força axial na extremidade j
Q_j	- força cortante na extremidade j
M_j	- momento fletor na extremidade j
M_{tj}	- momento de torção na extremidade j
E	- módulo de elasticidade à flexão
G	- módulo de elasticidade ao cisalhamento
A_x	- área da seção normal ao eixo x
I_x	- momento de inércia segundo x

I_y	- momento de inércia segundo y
I_z	- momento de inércia segundo z
K, CR	- coeficiente de recalque da base elástica
K	- rigidezes dos apoios elásticos
Δ	- deformações impostas
R	- reações
ℓ_i	- comprimento da barra i
q_j, q_k	- taxas de carga distribuídas linearmente, nas <u>ex</u> tremidades j e k
N, P, F, M, M_t	- cargas concentradas

B - APLICAÇÃO DOS PROGRAMAS

Embora simples, as estruturas apresentadas permitem mostrar a versatilidade e a facilidade de entrada dos dados.

No cabeçalho dos formulários são impressos a identificação da estrutura analisada, as cargas que constituem um dado carregamento, as unidades adotadas e referências para o confronto dos resultados.

A seguir é reproduzida a imagem dos cartões de dados na ordem da leitura. Cada carregamento tem o cabeçalho de identificação da estrutura, a linha de identificação do título do carregamento, a imagem dos cartões de dados desse carregamento e os resultados da análise. Após a análise completa da estrutura, repete-se o cabeçalho de identificação e se anota o tempo de processamento.

APLICAÇÃO - I

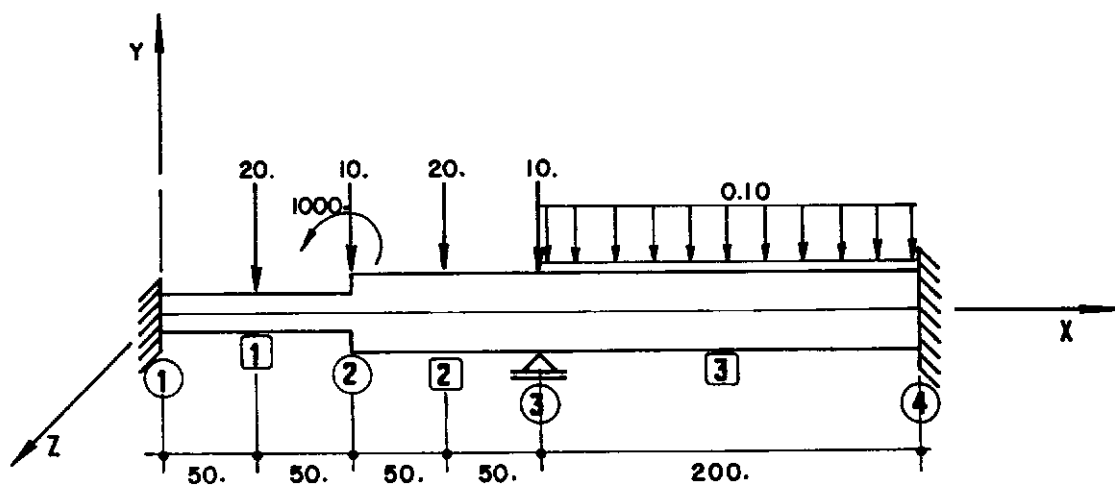


FIG. 8 - 1 VIGA CONTÍNUA

APLICAÇÃO - II

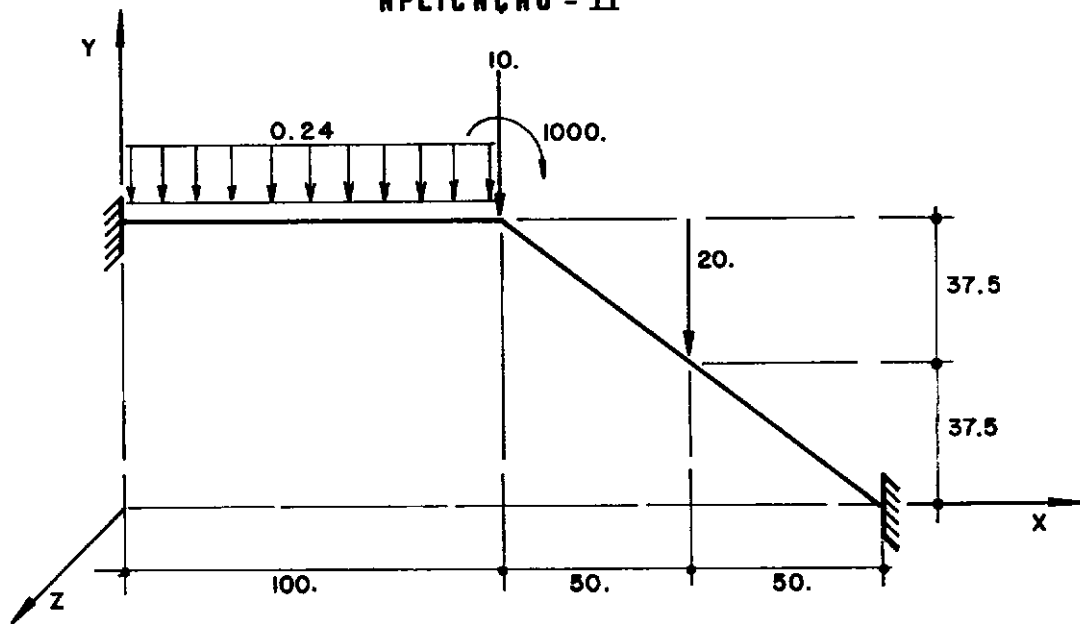


FIG. 8 - 2 PÓRTICO PLANO

APLICAÇÃO - III

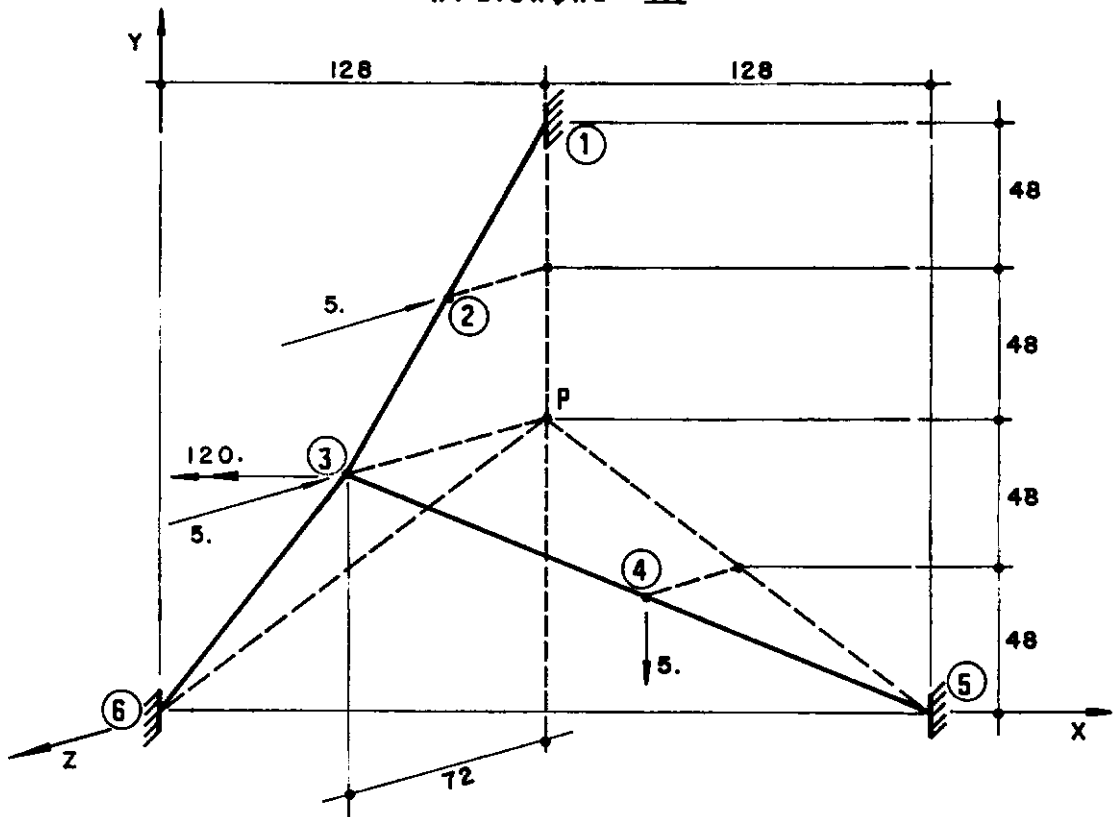


FIG. 8 - 3 PÓRTICO NO ESPAÇO

APLICAÇÃO - IV

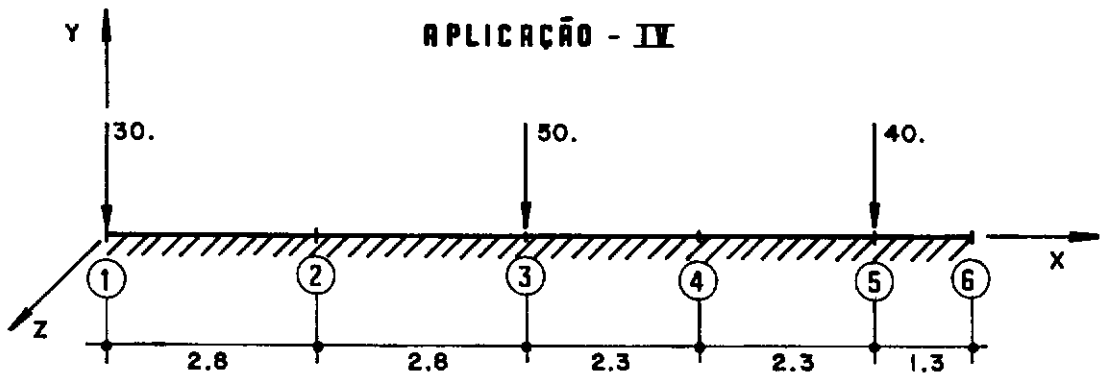


FIG. 8 - 4 VIGA S/BASE ELÁSTICA

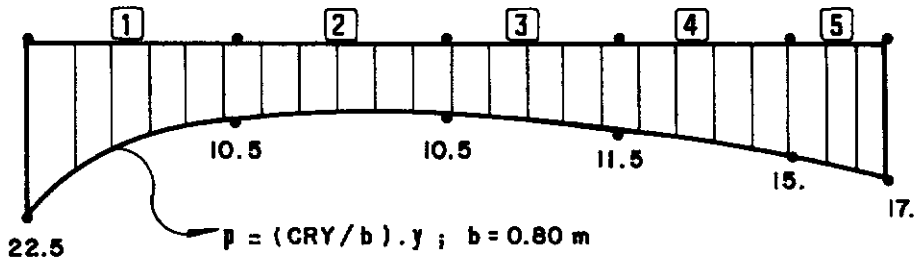


FIG. 8-4b LINHA DE PRESSÃO

APLICAÇÃO - V

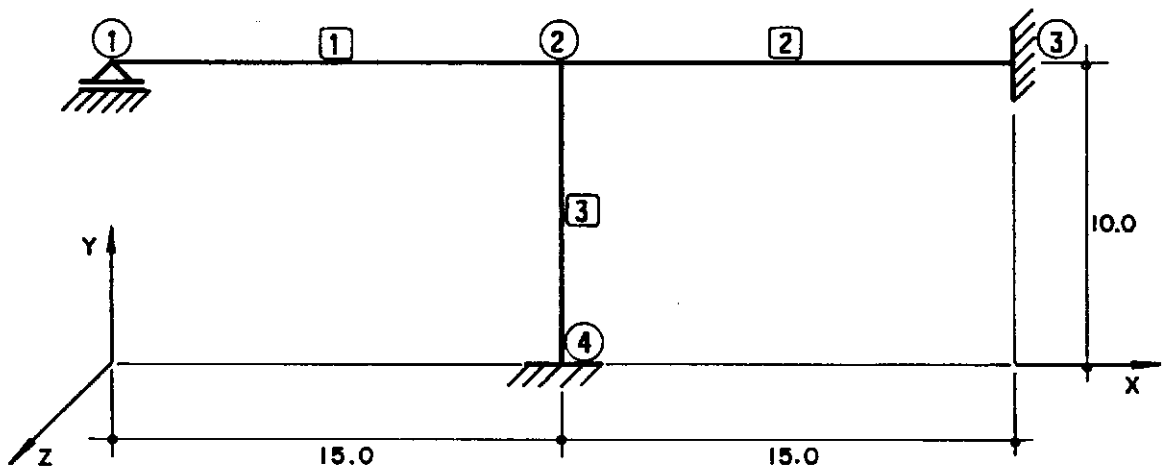


FIG. 8-5 PÓRICO PLANO COM RAMIFICAÇÃO

APLICAÇÃO - VI

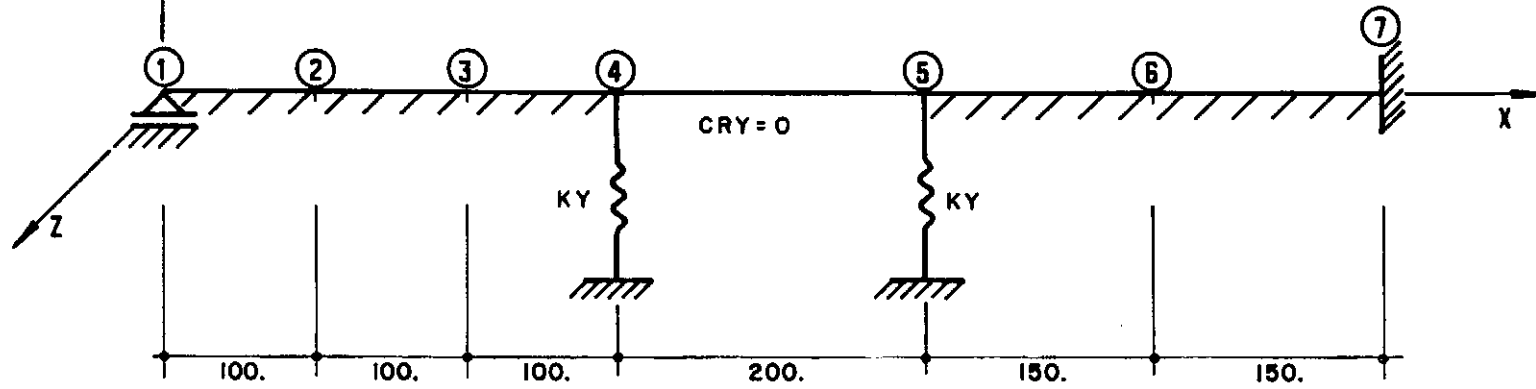


FIG. 8 - 6 a VIGA S/BASE ELÁSTICA COM APOIOS ELÁSTICOS

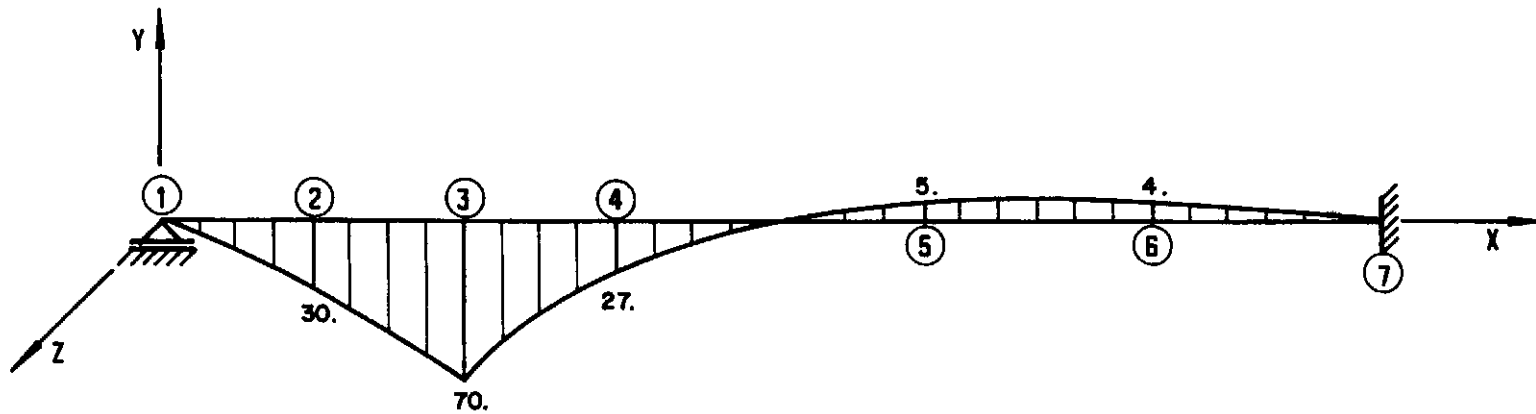


FIG. 8 - 6 b LINHA INFLUÊNCIA MOMENTO FLETOR SECÃO ③

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA

//JO
APLICACAO 1 FIG. B-1 UNID. KIPS-FOOT

//FO
CONFRONTO WEAVER(13), EXEMPLOS 3.7 FIG. 3.1 PAG. 127

/*
//N 3.
**SY 1. 3. 4.
*SZZ 1. 4.

/*
//L 1. 3. 100.00 100.00 200.00
//IZ 1. 3. 0.100000E 04 0.200000E 04 0.200000E 04
//E= 1. 3. 0.100000E 05

/*

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA
 ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO I WEAVING FIG. B-1), UNID. KIPS-FOOT
 CONFRONTO I WEAVING FIG. B-1), EXEMPLOS 3.7 FIG. 3.1 PAG. 127

LIST. DADOS CARREGAMENTO UNICO
 =====
 //CY= 1. 2. -20.00 50.00
 //UY= 3. 3. -0.10
 **CY= 2. 3. -10.00
 *CZZ 2. 2. 1000.00

/EXE

RESULTADOS CARREGAMENTO UNICO
 =====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA	CORTANTE	M.FLETOR	CORTANTE	M.FLETOR
1	33.05	1281.74	-13.05	1023.80
2	3.05	-23.80	16.94	-670.63
3	12.52	670.63	7.47	-164.68

REACOES

JUNTA	REACAO Y	REACAO Z
1	33.05	1281.74
3	39.47	0.00
4	7.47	-164.68

DESLOCAMENTOS GENERALIZADOS

JUNTA	DESLOC Y	DESLOC Z
1	0.0000	0.0000
2	-0.1316	0.0012
3	0.0000	0.0008
4	-0.0000	-0.0000

/FIM

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA
ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO I FIG. B-1 UNID. KIPS-FOOT
CONFRONTO WEAVER(13), EXEMPLOS 3.7 FIG. 3.1 PAG. 127

PROCESSAMENTO CONCLUIDO

HR. MAQUINA INICIAL=1908.17

HR. MAQUINA FINAL =1908.19

TEMPO PROCESSAMENTO=0. HRS.* 1.2MIN.* *

ANALISE PORTICO PLANO M. MATRIZES TRANSFERENCIA

//JO
APLICACAO II FIG. B-2 UNID. KIPS-FOOT

//FO
CONFRONTO WEAVER(13), EXEMPLOS 3.7 FIG. 3.3 PAG. 129

//N 3.

```

/*
***X      2.      4.      100.000      150.000      200.000
***Y      1.      3.      75.000      75.000      37.500
//AX=     1.      3.      0.100000E 02
//E=      1.      3.      0.100000E 05
//IZ=     1.      3.      0.100000E 04
    
```

/IMP

BARRA	COMPR.	CX	CY
1	100.000	1.000	0.000
2	62.500	0.800	-0.600
3	62.500	0.800	-0.600

/*

ANALISE PORTICO PLANO M. MATRIZES TRANSFERENCIA
 ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO II WEAVING FIG. 13-2, UNID. KIPS-FOOT
 CONFRONTO 3.7 FIG. 3.3 PAG. 129

LIST. DADOS CARREGAMENTO UNICO
 =====
 //UY 1. 1. -0.24
 **CY 2. 3. -10.00 -20.00
 *CZZ 2. 2. -1000.00
 /EXE

RESULTADOS CARREGAMENTO UNICO
 =====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA		FX	FY	MZZ
1	JJ	20.26	13.13	436.54
	JK	-20.26	10.86	-322.86
2	JJ	28.72	-4.53	-677.13
	JK	-28.72	4.53	393.80
3	JJ	40.72	-20.53	-393.80
	JK	-40.72	20.53	-889.52

REACOES

JUNTA		RX	RY	RZZ
1		20.26	13.13	436.54
4		-20.26	40.86	-889.52

DESLOCAMENTOS GENERALIZADOS

JUNTA		DX	DY	DZZ
1		0.00000	0.00000	0.00000
2		-0.02026	-0.09936	-0.00179
3		-0.03374	-0.08742	0.00154
4		0.00000	0.00000	0.00000

ANALISE PORTICO NO ESPACO M. MATRIZES TRANSFERENCIA

APLICACAO III FIG. B-3 UNID. KIPS-FOOT

CONFRONTO WEAVER(13), EXEMPLOS 3.7 FIG. 3.9 PAG. 133

/*
//N 4.
//NR 1.
*RAM 3.

/*									
***X	1.	5.	128.000		128.000		128.000		192.000
***Y	1.	4.	192.000		144.000		96.000		48.000
***Z	2.	4.	36.000		72.000		36.000		
//XP=	3.	5.	128.000						
//YP=	3.	5.	96.000						
//ZP=	3.	5.	0.000						
//AX=	1.	5.	0.900000E 01						
//IZ=	1.	5.	0.800000E 02						
//IX=	1.	5.	0.640000E 02						
//IY=	1.	5.	0.280000E 02						
//IG=	1.	5.	0.400000E 04						
//E=	1.	5.	0.100000E 05						

/IMP
BARRA COMPR. BETA R(1 A 9)

1	60.000	0.000	0.000	-0.800	0.600	0.000	0.600	0.800	-1.000	0.000	0.000
2	60.000	0.000	0.000	-0.800	0.600	0.000	0.600	0.800	-1.000	0.000	0.000
3	87.726	-72.893	0.729	-0.547	-0.410	-0.328	0.246	-0.911	0.600	0.800	0.000
4	87.726	-72.893	0.729	-0.547	-0.410	-0.328	0.246	-0.911	0.600	0.800	0.000
5	175.453	-72.893	0.729	0.547	0.410	0.328	0.246	-0.911	-0.600	0.800	0.000

/*

ANALISE PORTICO NO ESPACO M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO III FIG. B-3 UNID. KIPS-FOOT
CONFRONTO WEAVER(13); EXEMPLOS 3.7 FIG. 3.9 PAG. 133

LIST. DADOS CARREGAMENTO NO. 1 - CARGAS NA JUNTA 3

```

=====
**CZ      3.      3.      -5.00
*CXX      3.      3.     -120.00
/EXE
    
```

RESULTADOS CARREGAMENTO NO. 1 - CARGAS NA JUNTA 3

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA		FX	FY	FZ	MXX	MYY	MZZ
1	JJ	4.37	0.97	0.00	-0.00	-0.00	40.08
	JK	-4.37	-0.97	-0.00	0.00	0.00	18.52
2	JJ	4.37	0.97	0.00	-0.00	-0.00	-18.52
	JK	-4.37	-0.97	-0.00	0.00	-0.00	77.13
5	JJ	2.48	0.24	0.05	2.95	-2.94	13.70
	JK	-2.48	-0.24	-0.05	-2.95	-5.85	28.91
3	JJ	2.48	-0.24	-0.05	-2.95	5.85	-28.91
	JK	-2.48	0.24	0.05	2.95	-1.45	7.60
4	JJ	2.48	-0.24	-0.05	-2.95	1.45	-7.60
	JK	-2.48	0.24	0.05	2.95	2.94	-13.70

REACOES

JUNTA	RX	RY	RZ	RXX	RYY	RZZ
1	-0.00	-2.91	3.40	-40.08	0.00	-0.00
6	1.85	1.45	0.79	-7.03	11.85	3.90
5	-1.85	1.45	0.79	-7.03	-11.85	-3.90

DESLOCAMENTOS GENERALIZADOS

JUNTA	DX	DY	DZ	DXX	DYY	DZZ
-------	----	----	----	-----	-----	-----

1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	0.00000	-0.02540	-0.03872	0.00080	-0.00000	0.00000
3	0.00000	-0.00077	-0.01075	-0.00277	-0.00000	0.00000
4	0.00000	-0.02540	0.02797	-0.00046	0.00076	0.00062
5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

ANALISE PORTICO NO ESPACO M. MATRIZES TRANSFERENCIA
ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

CONFRONTO III WEAVER FIG. 8-7; EXEMPLOS 3.7 FIG. 3.9 PAG. 133

LIST. DADOS CARREGAMENTO NO. 2 - CARGAS NAS JUNTAS 2 E 4

```

=====
**CZ      2.      2.      -5.00
**CY      4.      4.      -5.00
/EXE
  
```

RESULTADOS CARREGAMENTO NO. 2 - CARGAS NAS JUNTAS 2 E 4

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA		FX	FY	FZ	MXX	MYX	MZY
1	JJ	2.21	2.84	0.40	6.07	-16.23	94.55
	JK	-2.21	-2.84	-0.40	-6.07	-8.12	76.15
2	JJ	-0.78	-1.15	0.40	6.07	8.12	-76.15
	JK	0.78	1.15	-0.40	-6.07	-32.49	6.86
5	JJ	2.26	0.24	-0.18	5.12	10.55	13.90
	JK	-2.26	-0.24	0.18	-5.12	21.24	28.52
3	JJ	-0.73	0.42	-1.72	-0.03	-56.03	3.95
	JK	-0.73	-0.42	-1.72	0.03	-95.60	32.94
4	JJ	-3.46	-0.81	-2.27	-0.03	95.60	-32.94
	JK	-3.46	0.81	2.27	0.03	103.65	-38.15

REACOES

JUNTA	RX	RY	RZ	RXX	RYY	RZZ
1	-0.40	-0.06	3.60	-94.55	-14.59	-9.34
6	1.83	1.15	0.70	-1.14	16.52	-7.52
5	-1.43	3.91	0.68	-56.89	-5.01	-94.54

DESLOCAMENTOS GENERALIZADOS

JUNTA	DX	DY	DZ	DXX	DYY	DZZ
1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

3	-0:05215	=0:04365	=0:06866	-0:00042	0:00165	=0:00015
4	-0:28169	-0:42828	0:06201	0:00057	-0:00007	0:00115
5	0:00000	0:00000	0:00000	0:00000	0:00000	0:00000

/FIM

ANALISE PORTICO NO ESPACO M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO III FIG. B-3 UNID. KIPS-FOOT
CONFRONTO WEAVER(13), EXEMPLOS 3.7 FIG. 3.9 PAG. 133

PROCESSAMENTO CONCLUIDO

HR. MAQUINA INICIAL= 1907.41

HR. MAQUINA FINAL = 1907.49

TEMPO PROCESSAMENTO= 0. HRS.* 4.8 MIN.*

ANALISE VIGA BASE ELASTICA M. MATRIZES TRANSFERENCIA

//JO
APLICACAO IV FIG. B-4 UNID. T-M

//FO
CONFRONTO WOLFER(15), EXEMPLO 3 PAG. 19

```

/*
///N      5.
*LRE      0.  0.
*LRD      0.  0.

/*
///L      1.  5.          2.80          2.80          2.30          2.30          1.30
///IZ=    1.  5.          0.340000E-01
///E=     1.  5.          0.210000E 07
/CRY=    1.  5.          0.400000E 04

/*

```

ANALISE VIGA BASE ELASTICA M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO IV FIG. B-4 UNID. T-M
CONFRONTO WOLFER(15), EXEMPLO 3 PAG. 19

LIST. DADOS CARGAS NAS JUNTAS 1-3-5

=====

**CY	1.	5.	-30.00	0.00	-50.00	0.00	-40.00

/EXE

RESULTADOS CARGAS NAS JUNTAS 1-3-5

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA	CORTANTE	M.FLETOR	CORTANTE	M.FLETOR
1	-30.00	0.00	-4.78	-28.79
2	4.78	28.79	-27.56	16.04
3	-22.43	-16.04	1.67	-11.89
4	-1.67	11.89	-22.98	11.26
5	-17.01	-11.26	0.00	-0.00

DESLOCAMENTOS GENERALIZADOS

JUNTA	DESLOC Y	DESLOC Z
1	-0.0045	0.0012
2	-0.0021	0.0003
3	-0.0021	-0.0001
4	-0.0023	-0.0001
5	-0.0030	-0.0003
6	-0.0034	-0.0002

/FIM

ANALISE PORTICO PLANO M. MATRIZES TRANSFERENCIA

//JO
APLICACAO V FIG. B-5 UNID. T-M

//FO
CONFRONTO FONSECA(14), EXEMPLO 380 FIG. 380.1 PAG. 132

/*
//N 2.
//NR 1.
*RAM 2.
*LRE 0. 1. 0.

/*
***X 2. 4. 15.000 30.000 15.000
***Y= 1. 3. 10.000
//I2 1. 3. 0.500000E-01 0.500000E-01 0.167000E-01
//E= 1. 3. 0.210000E 07
//AX= 1. 3. 0.100000E 00

/IMP

BARRA	COMPR.	CX	CY
1	15.000	1.000	0.000
2	15.000	1.000	0.000
3	10.000	0.000	1.000

/*

ANALISE PORTICO PLANO M. MATRIZES TRANSFERENCIA
 ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO V FIG. B-5 UNID. T-M
 CONFRONTO FONSECA(14), EXEMPLO 380 FIG. 380.1 PAG. 132

LIST. DADOS RACALQUE DA JUNTA 1, DIRECAO Y, DE 1 CM

=====

**DY	1.	1.	-0.01
------	----	----	-------

/EXE

RESULTADOS RACALQUE DA JUNTA 1, DIRECAO Y, DE 1 CM

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA		FX	FY	MZZ
1	JJ	0.00	-0.61	0.00
	JK	0.00	0.61	-9.20
3	JJ	1.22	0.45	1.50
	JK	-1.22	-0.45	3.08
2	JJ	-0.45	0.60	6.12
	JK	0.45	-0.60	2.98

REACOES

JUNTA	RX	RY	RZZ
1	0.00	-0.61	0.00
4	-0.45	1.22	1.50
3	0.45	-0.60	2.98

DESLOCAMENTOS GENERALIZADOS

JUNTA	DX	DY	DZZ
1	-0.00003	-0.01000	0.00088
2	-0.00003	-0.00005	0.00022
3	0.00000	0.00000	0.00000

ANALISE PORTICO PLANO M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO V FIG. B-5 UNID. T=M
CONFRONTO FONSECA(14), EXEMPLO 380 FIG. 380.1 PAG. 132

LIST. DADOS AUMENTO DE TEMPERATURA DE 20 C

=====

/TUX=	1.	3.	20.00	0.000010
/EXE				

RESULTADOS AUMENTO DE TEMPERATURA DE 20 C

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA		FX	FY	MZZ
1	JJ	0.00	-0.10	0.00
	JK	0.00	0.10	-1.59
3	JJ	-0.98	-1.11	-5.76
	JK	-0.98	1.11	-5.38
2	JJ	1.11	0.88	6.98
	JK	-1.11	-0.88	6.22

REACOES

JUNTA		RX	RY	RZZ
1		0.00	-0.10	0.00
4		1.11	-0.98	-5.76
3		-1.11	-0.88	6.22

DESLOCAMENTOS GENERALIZADOS

JUNTA		DX	DY	DZZ
1		-0.00592	0.00000	0.00016
2		-0.00292	0.00195	0.00005
3		0.00000	0.00000	0.00000

/FIM

ANALISE VIGA BASE ELASTICA M. MATRIZES TRANSFERENCIA

APLICACAO VI FIG. B-6 UNID. T-CM

CONFRONTO -----

```

/*
///N      6.      0.
*LRE      1.      0.

/*
///L=     1.      3.      100.00
///L      4.      6.      200.00      150.00      150.00
///IZ=     1.      6.      0.340000E 07
///E=      1.      6.      0.210000E 03
**KY=      4.      5.      0.280000E 03
/CRY=      1.      5.      0.400000E 00
/CRY      4.      4.      0.000000E 00

/*

```

ANALISE VIGA BASE ELASTICA M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

APLICACAO VI FIG. B-6 UNID. T-CM

CONFRONTO -----

LIST. DADOS LINHA DE INFL. DE MOMENTO FLETOR NA JUNTA 3

=====

*DZZ 3. 3. 1.00

/EXE

RESULTADOS LINHA DE INFL. DE MOMENTO FLETOR NA JUNTA 3

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA	CORTANTE	M.FLETOR	CORTANTE	M.FLETOR
1	-6955.03	0.00	6363.63	-676004.84
2	-6363.63	676004.84	4400.13	-1227382.63
3	-4400.13	1227382.64	2531.18	-1559457.79
4	4937.50	1559457.79	-4937.50	-571956.10
5	3494.10	571956.10	-3176.32	-72909.05
6	3176.32	72909.05	-3088.71	393958.27

DESLOCAMENTOS GENERALIZADOS

JUNTA	DESLOC Y	DESLOC Z
1	0.0000	-0.2876
2	-30.3725	-0.3356
3	-70.0696	0.5287
4	-26.6738	0.3314
5	5.1550	0.0329
6	3.7699	-0.0339
7	0.0000	0.0000

REACOES

JUNTA	REACAO Y	REACAO Z
1	-6955.03	0.00
4	7468.69	0.00
5	-1443.40	0.00
7	-3088.71	393958.27

ANALISE VIGA BASE ELASTICA M. MATRIZES TRANSFERENCIA
ENGENHARIA CIVIL - COPPE-UF RJ - ELDON LONDE MELLO

APLICACAO VI FIG. B-6 UNID. T-CM
CONFRONTO -----

PROCESSAMENTO CONCLUIDO

HR. MAQUINA INICIAL= 1906.85

HR. MAQUINA FINAL = 1906.87

TEMPO PROCESSAMENTO= 0.HRS.*1.2 MIN.* *

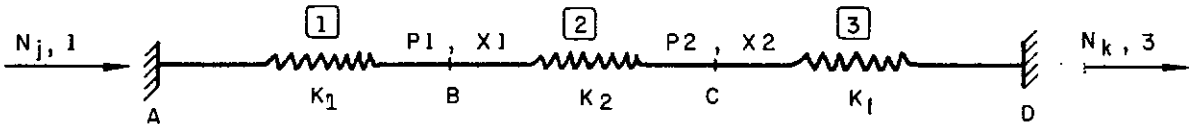
C - CONCLUSÃO SOBRE A EFICIÊNCIA DOS PROGRAMAS

1 - Condicionamento da Matriz de Transferência.

De um modo geral, pode-se dizer que as matrizes de rigidez e de transferência têm comportamentos opostos quanto ao seu condicionamento, isto é, quando uma é bem-condicionada a outra não o é.

Um simples exemplo apresentado por LIVESLEY (6), pág. 204 a 211, esclarece a questão.

Seja o problema uni-dimensional:



A análise pelo Método de Rigidez conduz às duas equações de equilíbrio

$$\begin{vmatrix} P_1 \\ P_2 \end{vmatrix} = \begin{vmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_1 + k_2 \end{vmatrix} \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} \quad C - 1$$

cujas soluções numéricas pelo processo da eliminação de GAUSS é, passo a passo:

a) multiplicar a primeira equação por $k_2 / (k_1 + k_2)$:

$$P_1 k_2 / (k_1 + k_2) = k_2 x_1 - k_2^2 / (k_1 + k_2)$$

b) adicionar a segunda equação para obter

$$p_2 + p_1 k_2 / (k_1 + k_2) = \left[k_1 + k_2 - k_2^2 / (k_1 + k_2) \right] x_2$$

$$\text{ou } x_2 = \frac{p_2 + p_1 k_2 / (k_1 + k_2)}{\frac{(k_1 + k_2)^2 - k_2^2}{k_1 + k_2}} \quad C - 2$$

A expressão $(k_1 + k_2)^2 - k_2^2$ de C - 2 é o determinante da matriz de rigidez de C - 1.

Se k_1 for pequeno em presença de k_2 , o determinante tende a zero e a matriz tende a singular: má-condicionada.

Mas, se $k_2 = 0$, o determinante será k_1^2 e a solução numérica não será afetada como se depreende dos passos a) e b): a matriz é bem-condicionada.

Façamos a análise numérica pelo Método das Matrizes de Transferência.

As matrizes básicas são:

$$|T|_1 = |T|_3 = \begin{vmatrix} 1 & 1/k_1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix} ; \quad |T|_2 = \begin{vmatrix} 1 & 1/k_2 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$|F|_B = \begin{vmatrix} 1 & 0 & 0 \\ 0 & -1 & p_1 \\ 0 & 0 & 1 \end{vmatrix} ; \quad |F|_C = \begin{vmatrix} 1 & 0 & 0 \\ 0 & -1 & p_2 \\ 0 & 0 & 1 \end{vmatrix}$$

Devemos ter:

$$\{V\}_{k,3} = |T|_3 |F|_C |T|_2 |F|_B |T|_1 \{V\}_{j,1}$$

cujo desenvolvimento é

$$\begin{vmatrix} 0 & 1 & 1/k_1 + \left(\frac{1}{k_1} + \frac{1}{k_2}\right) p_1 \left(\frac{1}{k_1} + \frac{1}{k_2}\right) p_2/k_1 \\ N & 0 & -1 & -p_1 - p_2 \\ 1_{k,3} & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} 0 \\ N \\ 1_{j,1} \end{vmatrix}$$

A primeira equação fornece

$$N_{j,1} = - \frac{p_1 \left(\frac{1}{k_1} + \frac{1}{k_2}\right) + p_2/k_1}{\frac{1}{k_1} + \left(\frac{1}{k_1} + \frac{1}{k_2}\right)} \quad C - 3$$

e, a segunda

$$N_{k,3} = -N_{j,1} - p_1 - p_2$$

Se $k_2 = 0$, a expressão C - 3 será, numericamente, indeterminada: mau-condicionamento da matriz.

Se k_1 for pequeno em presença de k_2 , não haverá mau-condicionamento porque a expressão C - 3 resulta numericamente bem determinada.

Pode ocorrer, no entanto, uma margem de erro no cálculo de C - 3 devida ao truncamento dos dígitos nas operações aritméticas.

2 - Erros de Truncamento

Os produtos sucessivos das matrizes acarretam erros nas operações aritméticas, devido aos truncamentos dos dígitos menos significativos.

No caso da matriz de transferência reduzida do programa de Viga Contínua, a ordem de grandeza de seus elementos se torna bastante elevada à medida que cresce o número de vãos, a ponto de ocorrer truncamento de dígitos significativos, ou seja, as operações de soma já não terão mais significado. Tal fato pode conduzir ao mau-condicionamento da matriz de transferência reduzida total da viga.

Os exemplos anexos mostram a evolução dos erros à medida que se aumenta o número de vãos de uma mesma viga com o mesmo número de barras.

A viga de 8 vãos está correta, mas a de 16 vãos já apresenta uma instabilidade numérica a partir da barra número 110.

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA

//JO
VIGA DE 160 BARRAS E NO. VARIABEL DE VAOS - UNID. T-M

//EO
NUMERO DE VAOS 8

/*
//N 160.
**SY= 1. 161. 20.

/*
//L= 1. 160. 0.20
//IZ= 1. 160. 0.450000E-03
//E= 1. 160. 0.200000E 07

/*

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

VIGA DE 160 BARRAS E NO. VARIÁVEL DE VAOS - UNID. T-M

NUMERO DE VAOS 8

LIST. DADOS CARGA UNIFORME EM TODAS AS BARRAS 1T/M

=====

//UY= 1. 160. -1.00

/EXE

RESULTADOS CARGA UNIFORME EM TODAS AS BARRAS 1T/M

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA	CORTANTE	M.FLETOR	CORTANTE	M.FLETOR
1	1.57	0.00	-1.37	0.29
2	1.37	-0.29	-1.17	0.55
3	1.17	-0.55	-0.97	0.76
4	0.97	-0.76	-0.77	0.94
5	0.77	-0.94	-0.57	1.07
6	0.57	-1.07	-0.37	1.17
7	0.37	-1.17	-0.17	1.22
8	0.17	-1.22	0.02	1.24
9	-0.02	-1.24	0.22	1.21
10	-0.22	-1.21	0.42	1.15
11	-0.42	-1.15	0.62	1.05
12	-0.62	-1.05	0.82	0.90
13	-0.82	-0.90	1.02	0.72
14	-1.02	-0.72	1.22	0.49
15	-1.22	-0.49	1.42	0.23
16	-1.42	-0.23	1.62	0.07
17	-1.62	0.07	1.82	-0.41
18	-1.82	0.41	2.02	-0.80
19	-2.02	0.80	2.22	-1.22
20	-2.22	1.22	2.42	-1.69
21	-2.42	1.69	-1.91	-1.28
22	-1.91	1.28	-1.71	-0.92
23	-1.71	0.92	-1.51	-0.60
24	-1.51	0.60	-1.31	-0.32
25	-1.31	0.32	-1.11	-0.07
26	-1.11	0.07	-0.91	0.12
27	-0.91	-0.12	-0.71	0.28
28	-0.71	-0.28	-0.51	0.41

[illegible][illegible][illegible]

0.49
0.53
0.53
0.50
0.42
0.30
0.14
0.04
0.28
0.56
0.87
1.23
0.86
0.52
0.23
0.01
0.23
0.40
0.53
0.63
0.68
0.70
0.67
0.60
0.50
0.35
0.17
0.05
0.32
0.62
0.97
1.36
0.97
0.63
0.33
0.07
0.14
0.33
0.47
0.57
0.63
0.65
0.64
0.58
0.48
0.34
0.17
0.04
0.30
0.60
0.94
1.31
0.94

82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

[illegible][illegible][illegible][illegible]

136	-0.91	-0.12	1.11	-0.97
137	-1.31	0.31	1.31	-0.91
138	-1.51	0.60	1.51	-0.60
139	-1.71	0.92	1.71	-0.92
140	-1.91	1.28	1.91	-1.28
141	-2.42	1.69	2.22	-1.69
142	2.22	1.22	-2.02	1.22
143	2.02	0.80	-1.82	0.80
144	1.82	0.41	-1.62	0.41
145	1.62	0.07	-1.42	0.07
146	1.42	-0.23	-1.22	0.49
147	1.22	-0.49	-1.02	0.72
148	1.02	-0.72	-0.82	0.90
149	0.82	-0.90	-0.62	1.05
150	0.62	-1.05	-0.42	1.15
151	0.42	-1.15	-0.22	1.21
152	0.22	-1.21	-0.02	1.24
153	0.02	-1.24	0.17	1.22
154	-0.17	-1.22	0.37	1.17
155	-0.37	-1.17	0.57	1.07
156	-0.57	-1.07	0.77	0.94
157	-0.77	-0.94	0.97	0.76
158	-0.97	-0.76	1.17	0.55
159	-1.17	-0.55	1.37	0.29
160	-1.37	-0.29	1.57	0.00

REACOES

JUNTA	REACAO Y	REACAO Z
1	1.57	0.00
21	4.53	0.00
41	3.85	0.00
61	4.04	0.00
81	3.97	0.00
101	4.04	0.00
121	3.85	0.00
141	4.53	0.00
161	1.57	0.00

DESLOCAMENTOS GENERALIZADOS

JUNTA	DESLOC Y	DESLOC Z
1	0.0000	-0.0017
2	-0.0003	-0.0016
3	-0.0006	-0.0015
4	-0.0009	-0.0014
5	-0.0012	-0.0012
6	-0.0014	-0.0010
7	-0.0016	-0.0007
8	-0.0017	-0.0005

9		
10	-0.00018	-0.00002
11	-0.00018	0.00000
12	-0.00018	0.00003
13	-0.00017	0.00005
14	-0.00016	0.00007
15	-0.00014	0.00009
16	-0.00012	0.00010
17	-0.00009	0.00011
18	-0.00007	0.00011
19	-0.00005	0.00011
20	-0.00003	0.00010
21	-0.00001	0.00007
22	0.00000	0.00004
23	0.00000	0.00001
24	0.00000	-0.00001
25	-0.00000	-0.00002
26	-0.00001	-0.00003
27	-0.00002	-0.00004
28	-0.00003	-0.00004
29	-0.00003	-0.00003
30	-0.00004	-0.00003
31	-0.00004	-0.00001
32	-0.00004	-0.00000
33	-0.00004	0.00000
34	-0.00003	0.00001
35	-0.00003	0.00002
36	-0.00002	0.00003
37	-0.00001	0.00003
38	-0.00001	0.00004
39	-0.00000	0.00003
40	-0.00000	0.00002
41	0.00000	0.00001
42	-0.00000	-0.00001
43	-0.00001	-0.00003
44	-0.00002	-0.00005
45	-0.00003	-0.00005
46	-0.00004	-0.00006
47	-0.00006	-0.00005
48	-0.00006	-0.00005
49	-0.00007	-0.00004
50	-0.00008	-0.00002
51	-0.00008	-0.00001
52	-0.00007	0.00000
53	-0.00007	0.00001
54	-0.00006	0.00003
55	-0.00005	0.00004
56	-0.00004	0.00005
57	-0.00003	0.00005
58	-0.00002	0.00006
59	-0.00001	0.00005
60	0.00000	0.00004
61	0.00000	0.00002

62	-0.00000	-0.00002
63	-0.00000	-0.00004
64	-0.00001	-0.00005
65	-0.00002	-0.00005
66	-0.00003	-0.00005
67	-0.00003	-0.00004
68	-0.00005	-0.00004
69	-0.00006	-0.00002
70	-0.00007	-0.00001
71	-0.00007	-0.00000
72	-0.00007	0.00001
73	-0.00006	0.00002
74	-0.00006	0.00003
75	-0.00005	0.00004
76	-0.00004	0.00005
77	-0.00002	0.00005
78	-0.00001	0.00005
79	-0.00000	0.00004
80	-0.00000	0.00002
81	0.00000	-0.00000
82	-0.00000	-0.00002
83	-0.00000	-0.00004
84	-0.00001	-0.00005
85	-0.00002	-0.00005
86	-0.00004	-0.00005
87	-0.00005	-0.00004
88	-0.00006	-0.00003
89	-0.00006	-0.00002
90	-0.00007	-0.00001
91	-0.00007	0.00000
92	-0.00007	0.00001
93	-0.00006	0.00002
94	-0.00005	0.00004
95	-0.00005	0.00004
96	-0.00003	0.00005
97	-0.00002	0.00005
98	-0.00001	0.00005
99	-0.00000	0.00004
100	-0.00000	0.00002
101	-0.00000	-0.00000
102	-0.00000	-0.00002
103	-0.00001	-0.00004
104	-0.00002	-0.00005
105	-0.00003	-0.00006
106	-0.00004	-0.00005
107	-0.00005	-0.00005
108	-0.00006	-0.00004
109	-0.00007	-0.00003
110	-0.00007	-0.00001
111	-0.00008	-0.00000
112	-0.00008	0.00001
113	-0.00007	0.00002
114	-0.00006	0.00004

115	-0.00006	0.00005
116	-0.00004	0.00005
117	-0.00003	0.00006
118	-0.00002	0.00005
119	-0.00001	0.00005
120	-0.00000	0.00003
121	0.00000	0.00001
122	-0.00000	-0.00001
123	-0.00000	-0.00002
124	-0.00001	-0.00003
125	-0.00001	-0.00004
126	-0.00002	-0.00003
127	-0.00003	-0.00003
128	-0.00003	-0.00002
129	-0.00004	-0.00001
130	-0.00004	-0.00000
131	-0.00004	0.00000
132	-0.00004	0.00001
133	-0.00003	0.00003
134	-0.00003	0.00003
135	-0.00002	0.00004
136	-0.00001	0.00004
137	-0.00000	0.00003
138	0.00000	0.00002
139	0.00000	0.00001
140	0.00000	-0.00001
141	0.00000	-0.00004
142	-0.00001	-0.00007
143	-0.00003	-0.00010
144	-0.00005	-0.00011
145	-0.00007	-0.00011
146	-0.00009	-0.00011
147	-0.00012	-0.00010
148	-0.00014	-0.00009
149	-0.00016	-0.00007
150	-0.00017	-0.00005
151	-0.00018	-0.00003
152	-0.00018	-0.00000
153	-0.00018	0.00002
154	-0.00017	0.00003
155	-0.00016	0.00007
156	-0.00014	0.00010
157	-0.00012	0.00012
158	-0.00009	0.00014
159	-0.00006	0.00015
160	-0.00003	0.00016
161	0.00000	0.00017

/FIM

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA
ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

VIGA DE 160 BARRAS E NO. VARIABEL DE VAOS - UNID. T-M
NUMERO DE VAOS 8

PROCESSAMENTO CONCLUIDO

HR. MAQUINA INICIAL= 1807.84.

HR. MAQUINA FINAL = 1807.99

TEMPO PROCESSAMENTO= 0.HRS.* 9.0 MIN.* *

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA

//JO
VIGA DE 160 BARRAS E NO. VARIABEL DE VAOS - UNID. T-M

//EO
NUMERO DE VAOS 16

/*
//N 160.
**SY= 1. 161. 10.

/*
//L= 1. 160. 0.20
//IZ= 1. 160. 0.450000E-03
//E= 1. 160. 0.200000E 07

/*

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

VIGA DE 160 BARRAS E NO. VARIÁVEL DE VAOS - UNID. T-M

NUMERO DE VAOS 16

LIST. DADOS CARGA UNIFORME EM TODAS AS BARRAS 1T/M

=====

//UY= 1. 160. -1.00

/EXE

RESULTADOS CARGA UNIFORME EM TODAS AS BARRAS 1T/M

=====

ACOES NAS EXTREMIDADES DAS BARRAS

BARRA	CORTANTE	M.FLETOR	CORTANTE	M.FLETOR
1	0.78	0.00	-0.58	0.13
2	0.58	-0.13	-0.38	0.23
3	0.38	-0.23	-0.18	0.29
4	0.18	-0.29	0.01	0.31
5	-0.01	-0.31	0.21	0.28
6	-0.21	-0.28	0.41	0.22
7	-0.41	-0.22	0.61	0.12
8	-0.61	-0.12	0.81	-0.01
9	-0.81	0.01	1.01	-0.20
10	-1.01	0.20	1.21	-0.42
11	1.05	0.42	-0.85	-0.23
12	0.85	0.23	-0.65	-0.08
13	0.65	0.08	-0.45	0.03
14	0.45	-0.03	-0.25	0.10
15	0.25	-0.10	-0.05	0.13
16	0.05	-0.13	0.14	0.12
17	-0.14	-0.12	0.34	0.07
18	-0.34	-0.07	0.54	-0.01
19	-0.54	0.01	0.74	-0.14
20	-0.74	0.14	0.94	-0.30
21	0.98	0.30	-0.78	-0.13
22	0.78	0.13	-0.58	0.00
23	0.58	-0.00	-0.38	0.10
24	0.38	-0.10	-0.18	0.15
25	0.18	-0.15	0.01	0.17
26	-0.01	-0.17	0.21	0.15
27	-0.21	-0.15	0.41	0.08
28	-0.41	-0.08	0.61	-0.01

29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81

-0.81
1.00
0.80
0.60
0.40
0.20
0.00
-0.19
-0.39
-0.59
-0.79
0.99
0.79
0.59
0.39
0.19
-0.00
-0.20
-0.40
-0.60
-0.80
1.00
0.80
0.60
0.40
0.20
0.00
-0.19
-0.39
-0.59
-0.79
0.99
0.79
0.59
0.39
0.19
-0.00
-0.20
-0.40
-0.60
-0.80
1.00

0.01
0.15
0.33
0.15
0.01
-0.08
-0.14
-0.16
-0.14
-0.08
0.01
0.15
0.33
0.01
-0.08
-0.14
-0.16
-0.14
-0.08
0.01
0.15
0.33
0.01
-0.08
-0.14
-0.16
-0.14
-0.08
0.01
0.15
0.33
0.01
-0.08
-0.14
-0.16
-0.14
-0.08
0.01
0.15
0.33
0.01
-0.08
-0.14
-0.16
-0.14
-0.08
0.01
0.15
0.33

0.81
1.00
0.80
0.60
0.40
0.20
0.19
0.39
0.59
0.79
0.99
0.79
0.59
0.39
0.19
0.00
0.20
0.40
0.60
0.80
1.00
0.80
0.60
0.40
0.20
0.19
0.39
0.59
0.79
0.99
0.79
0.59
0.39
0.19
0.00
0.20
0.40
0.60
0.80
1.00
0.80

-0.15
-0.33
-0.15
-0.01
0.08
0.14
0.16
0.14
0.08
-0.01
-0.15
-0.33
-0.01
0.08
0.14
0.16
0.14
0.08
-0.01
-0.15
-0.33
-0.01
0.08
0.14
0.16
0.14
0.08
-0.01
-0.15
-0.33
-0.01
0.08
0.14
0.16
0.14
0.08
-0.01
-0.15
-0.33
-0.15

-	0.6
-	0.4
-	0.2
-	0.0
-	0.1
-	0.3
-	0.5
-	0.7
-	0.9
-	0.7
-	0.5
-	0.3
-	0.1
-	0.0
-	0.2
-	0.4
-	0.6
-	0.8
-	1.0
-	0.8
-	0.6
-	0.4
-	0.2
-	0.0
-	0.1
-	0.3
-	0.5
-	0.7
-	0.9
-	0.7
-	0.5
-	0.3
-	0.1
-	0.0
-	0.2
-	0.4
-	0.6
-	0.8
-	1.0
-	0.8
-	0.6
-	0.4
-	0.2
-	0.0
-	0.1
-	0.3
-	0.5
-	0.7
-	0.9
-	0.6
-	0.4
-	0.2
-	0.0

[illegible][illegible]

82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

135	0.00	-0.00	0.00	0.00
136	0.00	-0.00	0.00	0.00
137	-1.00	-0.00	0.00	-1.00
138	0.00	0.00	0.00	0.00
139	-1.00	0.00	0.00	-1.00
140	0.00	0.00	0.00	0.00
141	-1.00	0.00	0.00	-1.00
142	1.00	0.00	0.00	1.00
143	1.00	0.00	0.00	1.00
144	0.00	-0.00	0.00	0.00
145	0.00	-0.00	0.00	0.00
146	0.00	-0.00	0.00	0.00
147	0.00	-0.00	0.00	0.00
148	-0.00	-0.00	0.00	-0.00
149	-0.00	-0.00	0.00	-0.00
150	-0.00	-0.00	0.00	-0.00
151	-0.00	-0.00	0.00	-0.00
152	-0.00	-0.00	0.00	-0.00
153	-1.00	-0.00	0.00	-1.00
154	-1.00	-0.00	0.00	-1.00
155	-1.00	-0.00	0.00	-1.00
156	-1.00	-0.00	0.00	-1.00
157	-1.00	-0.00	0.00	-1.00
158	-2.00	-0.00	0.00	-2.00
159	-2.00	-0.00	0.00	-2.00
160	-2.00	-0.00	0.00	-2.00

REACOES

JUNTA	REACAO Y	REACAO Z
1	0.78	0.00
11	2.26	0.00
21	1.92	0.00
31	2.01	0.00
41	1.99	0.00
51	2.00	0.00
61	1.99	0.00
71	2.00	0.00
81	2.00	0.00
91	1.99	0.00
101	2.00	0.00
111	1.98	0.00
121	2.04	0.00
131	1.83	0.00
141	2.60	0.00
151	-0.26	0.00
161	2.78	0.00

DESLOCAMENTOS GENERALIZADOS

JUNTA	DESLOC Y	DESLOC Z
-------	----------	----------

1	-0.0000	-0.0002
2	-0.0000	-0.0001
3	-0.0000	-0.0001
4	-0.0001	-0.0000
5	-0.0001	-0.0000
6	-0.0001	0.0000
7	-0.0001	0.0000
8	-0.0000	0.0001
9	-0.0000	0.0001
10	-0.0000	0.0001
11	0.0000	0.0000
12	0.0000	-0.0000
13	-0.0000	-0.0000
14	-0.0000	-0.0000
15	-0.0000	-0.0000
16	-0.0000	-0.0000
17	-0.0000	0.0000
18	-0.0000	0.0000
19	-0.0000	0.0000
20	-0.0000	0.0000
21	0.0000	-0.0000
22	-0.0000	-0.0000
23	-0.0000	-0.0000
24	-0.0000	-0.0000
25	-0.0000	-0.0000
26	-0.0000	0.0000
27	-0.0000	0.0000
28	-0.0000	0.0000
29	-0.0000	0.0000
30	-0.0000	0.0000
31	0.0000	0.0000
32	-0.0000	-0.0000
33	-0.0000	-0.0000
34	-0.0000	-0.0000
35	-0.0000	-0.0000
36	-0.0000	0.0000
37	-0.0000	0.0000
38	-0.0000	0.0000
39	-0.0000	0.0000
40	-0.0000	0.0000
41	0.0000	-0.0000
42	-0.0000	-0.0000
43	-0.0000	-0.0000
44	-0.0000	-0.0000
45	-0.0000	-0.0000
46	-0.0000	0.0000
47	-0.0000	0.0000
48	-0.0000	0.0000
49	-0.0000	0.0000
50	-0.0000	0.0000
51	0.0000	0.0000
52	-0.0000	-0.0000
53	-0.0000	-0.0000

54	-0.00000	-0.00000
55	-0.00000	-0.00000
56	-0.00000	-0.00000
57	-0.00000	0.00000
58	-0.00000	0.00000
59	-0.00000	0.00000
60	-0.00000	0.00000
61	0.00000	-0.00000
62	-0.00000	-0.00000
63	-0.00000	-0.00000
64	-0.00000	-0.00000
65	-0.00000	-0.00000
66	-0.00000	0.00000
67	-0.00000	0.00000
68	-0.00000	0.00000
69	-0.00000	0.00000
70	-0.00000	0.00000
71	0.00000	0.00000
72	-0.00000	-0.00000
73	-0.00000	-0.00000
74	-0.00000	-0.00000
75	-0.00000	-0.00000
76	-0.00000	0.00000
77	-0.00000	0.00000
78	-0.00000	0.00000
79	-0.00000	0.00000
80	-0.00000	0.00000
81	0.00000	-0.00000
82	-0.00000	-0.00000
83	-0.00000	-0.00000
84	-0.00000	-0.00000
85	-0.00000	-0.00000
86	-0.00000	-0.00000
87	-0.00000	0.00000
88	-0.00000	0.00000
89	-0.00000	0.00000
90	-0.00000	0.00000
91	0.00000	0.00000
92	-0.00000	-0.00000
93	-0.00000	-0.00000
94	-0.00000	-0.00000
95	-0.00000	-0.00000
96	-0.00000	0.00000
97	-0.00000	0.00000
98	-0.00000	0.00000
99	-0.00000	0.00000
100	-0.00000	0.00000
101	0.00000	-0.00000
102	-0.00000	-0.00000
103	-0.00000	-0.00000
104	-0.00000	-0.00000
105	-0.00000	-0.00000
106	-0.00000	0.00000

107	-0.00000	0.00000
108	-0.00000	0.00000
109	-0.00000	0.00000
110	-0.00000	0.00000
111	-0.00000	0.00000
112	-0.00000	-0.00000
113	-0.00000	-0.00000
114	-0.00000	-0.00000
115	-0.00000	-0.00000
116	-0.00000	0.00000
117	-0.00000	0.00000
118	-0.00000	0.00000
119	-0.00000	0.00000
120	-0.00000	0.00000
121	-0.00000	-0.00000
122	-0.00000	-0.00000
123	-0.00000	-0.00000
124	-0.00000	-0.00000
125	-0.00000	-0.00000
126	-0.00000	-0.00000
127	-0.00000	0.00000
128	-0.00000	0.00000
129	-0.00000	0.00000
130	-0.00000	0.00000
131	0.00000	0.00000
132	0.00000	-0.00000
133	-0.00000	-0.00000
134	-0.00000	-0.00000
135	-0.00000	0.00000
136	-0.00000	0.00000
137	0.00000	0.00000
138	0.00000	0.00000
139	0.00000	0.00000
140	0.00000	-0.00000
141	0.00000	-0.00001
142	-0.00000	-0.00002
143	-0.00000	-0.00002
144	-0.00001	-0.00002
145	-0.00001	-0.00001
146	-0.00001	-0.00000
147	-0.00002	0.00000
148	-0.00001	0.00001
149	-0.00001	0.00002
150	-0.00000	0.00003
151	0.00000	0.00004
152	0.00001	0.00005
153	0.00002	0.00005
154	0.00003	0.00005
155	0.00004	0.00004
156	0.00005	0.00003
157	0.00005	0.00000
158	0.00005	-0.00002
159	0.00004	-0.00006

169 8:8886 =8:8811

/FIM

ANALISE VIGA CONTINUA M. MATRIZES TRANSFERENCIA

ENGENHARIA CIVIL - COPPE-UFRJ - ELDON LONDE MELLO

VIGA DE 160 BARRAS E NO. VARIABEL DE VAOS - UNID. T-M

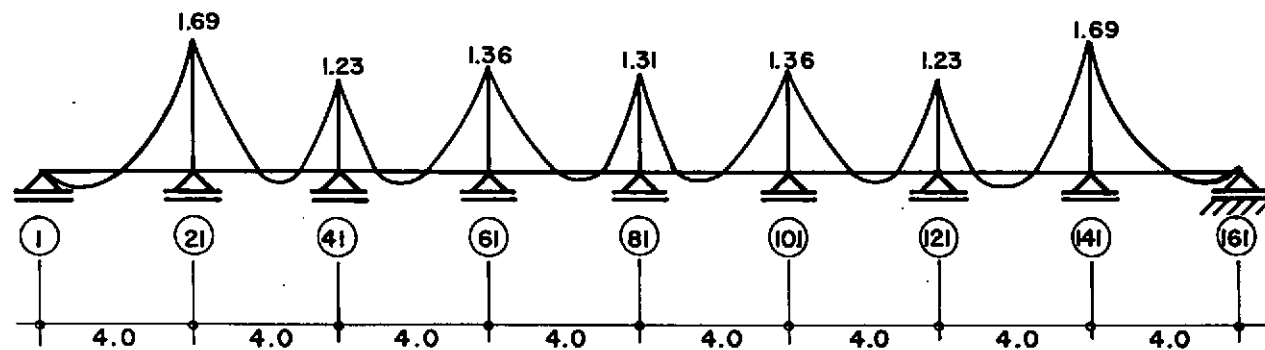
NUMERO DE VAOS 16

PROCESSAMENTO CONCLUIDO

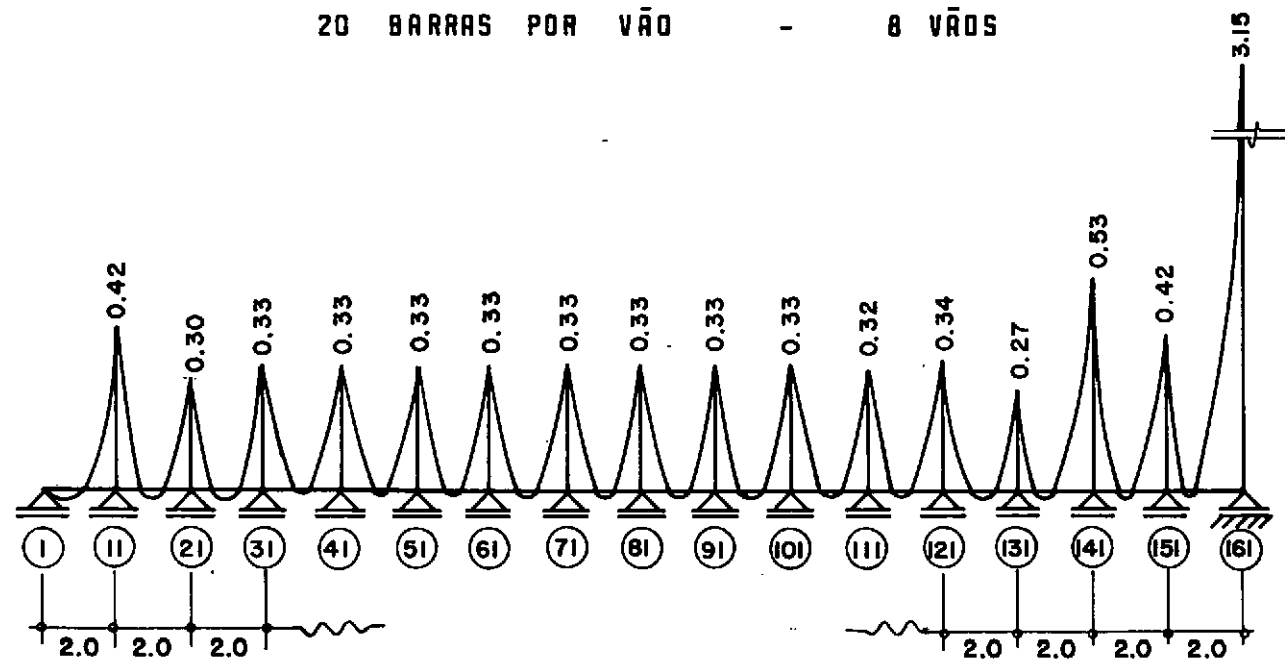
HR. MAQUINA INICIAL= 1807.99

HR. MAQUINA FINAL = 1808.15

TEMPO PROCESSAMENTO= 0.HRS.*9.6 MIN.* *



20 BARRAS POR VÃO - 8 VÃOS



10 BARRAS POR VÃO - 16 VÃOS

REFERÊNCIAS BIBLIOGRÁFICAS

1. Pastel, E.C., and Leckie, F. A., - "Matrix Methods in Elastomechanics", McGraw-Hill Book Co., Inc., New York, 1963
2. Gere, J.M., and Weaver, W., Jr., - "Analysis of Framed Structures", D. Van Nostrand Co., Inc., Princeton, New York, 1965.
3. A. Cornélis, G. Deprez, G. Founder, Ch. Massonnet, R. Maquoi, R. Muller - "Application des ordinateurs ao calcul des structures", Eyrolles Masson et Cie. Editeurs à Paris.
4. R. Lacroix - "La Méthode des Matrices Transfert", Annales I. T. B. T. P., Mars-avril 1967, n° 231-232 p. 374.
5. M. Biguenet - "Matrices de Transfert - application à l'étude des systèmes croisés à noeuds non déplaçables", Annales I. T. B. T. P., September 1968, n° 249, p. 1397.
6. Livesley, R. K. - "Matrix Methods of Structural Analysis", Pergamon Press, London, 1969.
7. Cheng, A. P., and Furr, H. L., - "Beams on discrete, non linear elastic supports", Journal of the Structural Division, ASCE, vol. 95, n° ST-11, Proc. Paper 6873, November, 1969, pp. 2335-2351.
8. Rubinstein, M. F. - "Matrix Computer Analysis of Structures", Prentice-Hall, Inc., Englewood Cliffs, New York, 1966.
9. Wang, P. C., - "Numerical and Matrix Methods in Structural Mechanics", John Wiley & Sons, Inc., New York, 1966.

McMinn, S.J., - "Matrices for Structural Analysis", John Wiley & Sons, Inc., New York, 1962.

Massachusetts Institute of Technology, Department of Civil Engineering, STRESS: A User's Manual, and STRESS: A Reference Manual, The M.I.T. Press, Cambridge, Mass., 1965.

Pacitti, Tércio - "FORTRAN - MONITOR, Princípios", Ao Livro Técnico S/A, Rio de Janeiro, 1968.

Weaver, W.Jr., - "Computer Programs for Structural Analysis", D. Van Nostrand Company, Inc., Princeton, 1967.

Fonseca, A. e Moreira, D.F. - "Estática das Construções", vol. 2, Ao Livro Técnico S/A, Rio de Janeiro, 1960.

Wölfer, K.H. - "Vigas Flotantes en medio elastico", Editorial Gustavo Gili, S.A., Barcelona, 1969.