In wide journal bearings and Winkler foundations, we use stiffness per unit length, s, of the supporting medium (Fig. 8.8b). Over the length of the support, this adds the following term to the total potential energy:

$$\frac{1}{2}\int_0^{\ell} sv^2\,dx \tag{8.41}$$

In Galerkin's approach, this term is $\int_0^{\ell} sv\phi\,dx$. When we substitute for $v = \mathbf{Hq}$ for the discretized model, the previous term becomes

$$\frac{1}{2}\sum_e \mathbf{q}^{\mathrm{T}} s \int_e \mathbf{H}^{\mathrm{T}}\mathbf{H}\,dx\,\mathbf{q} \tag{8.42}$$

We recognize the stiffness term in this summation, namely,

$$\mathbf{k}_s^e = s\int_e \mathbf{H}^{\mathrm{T}}\mathbf{H}\,dx = \frac{s\ell_e}{2}\int_{-1}^{+1} \mathbf{H}^{\mathrm{T}}\mathbf{H}\,d\xi \tag{8.43}$$

On integration, we have

$$\mathbf{k}_s^e = \frac{s\ell_e}{420}\begin{bmatrix} 156 & 22\ell_e & 54 & -13\ell_e \\ 22\ell_e & 4\ell_e^2 & 13\ell_e & -3\ell_e^2 \\ 54 & 13\ell_e & 156 & -22\ell_e \\ -13\ell_e & -3\ell_e^2 & -22\ell_e & 4\ell_e^2 \end{bmatrix} \tag{8.44}$$

For elements supported on an elastic foundation, this stiffness has to be added to the element stiffness given by Eq. 8.29. Matrix $\mathbf{k}_s^e$ is the consistent stiffness matrix for the elastic foundation.

8.7 PLANE FRAMES

Here, we consider plane structures with rigidly connected members. These members will be similar to the beams except that axial loads and axial deformations are present. The elements also have different orientations. Figure 8.9 shows a frame element. We have two displacements and a rotational deformation for each node. The nodal displacement vector is given by

$$\mathbf{q} = [q_1, q_2, q_3, q_4, q_5, q_6]^{\mathrm{T}} \tag{8.45}$$

We also define the local or body coordinate system x', y', such that x' is oriented along 1–2, with direction cosines ℓ, m (where $\ell = \cos\theta$, $m = \sin\theta$). These are evaluated using relationships given for the truss element, shown in Fig. 4.4. The nodal displacement vector in the local system is

$$\mathbf{q}' = [q'_1, q'_2, q'_3, q'_4, q'_5, q'_6]^{\mathrm{T}} \tag{8.46}$$

Recognizing that $q'_3 = q_3$ and $q'_6 = q_6$, which are rotations with respect to the body, we obtain the local–global transformation

$$\mathbf{q}' = \mathbf{Lq} \tag{8.47}$$

where

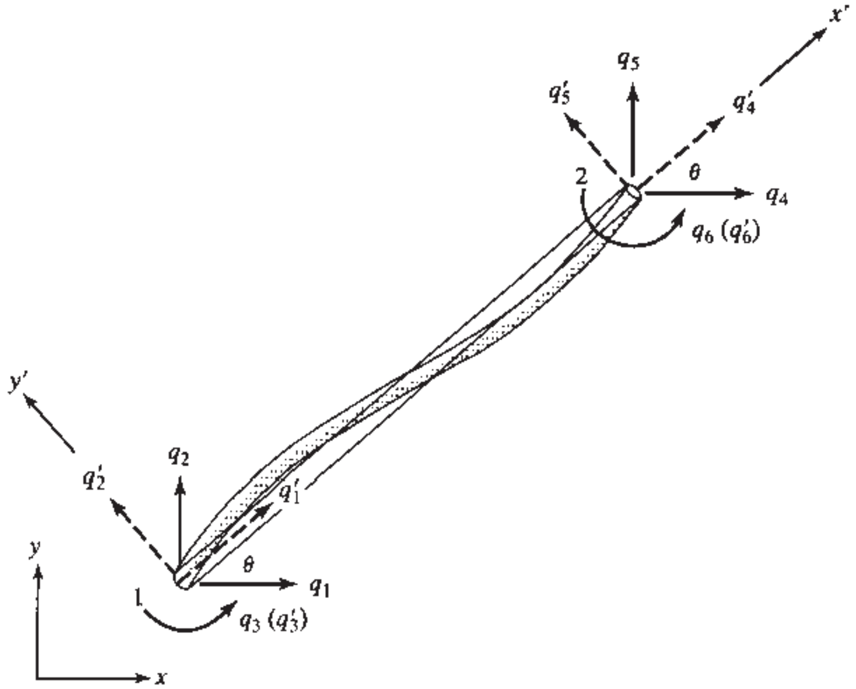


FIGURE 8.9 Frame element.

$$\mathbf{L} = \begin{bmatrix} \ell & m & 0 & 0 & 0 & 0 \\ -m & \ell & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ell & m & 0 \\ 0 & 0 & 0 & -m & \ell & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.48}$$

It is now observed that q'_2, q'_3, q'_5, and q'_6, are like the beam degrees of freedom, while q'_1 and q'_4 are similar to the displacements of a rod element, as discussed in Chapter 3. Combining the two stiffnesses and arranging in proper locations, we get the element stiffness for a frame element as

$$\mathbf{k}'^e = \begin{bmatrix} \frac{EA}{\ell_e} & 0 & 0 & \frac{-EA}{\ell_e} & 0 & 0 \\ 0 & \frac{12EI}{\ell_e^3} & \frac{6EI}{\ell_e^2} & 0 & \frac{-12EI}{\ell_e^3} & \frac{6EI}{\ell_e^2} \\ 0 & \frac{6EI}{\ell_e^2} & \frac{4EI}{\ell_e} & 0 & \frac{-6EI}{\ell_e^2} & \frac{2EI}{\ell_e} \\ \frac{-EA}{\ell_e} & 0 & 0 & \frac{EA}{\ell_e} & 0 & 0 \\ 0 & \frac{-12EI}{\ell_e^3} & \frac{-6EI}{\ell_e^2} & 0 & \frac{12EI}{\ell_e^3} & \frac{-6EI}{\ell_e^2} \\ 0 & \frac{6EI}{\ell_e^2} & \frac{2EI}{\ell_e} & 0 & \frac{-6EI}{\ell_e^2} & \frac{4EI}{\ell_e} \end{bmatrix} \tag{8.49}$$

As discussed in the development of a truss element in Chapter 4, we recognize that the element strain energy is given by

$$U_e = \tfrac{1}{2}\mathbf{q}'^{\mathrm{T}}\mathbf{k}'^{e}\mathbf{q}' = \tfrac{1}{2}\mathbf{q}^{\mathrm{T}}\mathbf{L}^{\mathrm{T}}\mathbf{k}'^{e}\mathbf{L}\mathbf{q} \tag{8.50}$$

or in Galerkin's approach, the internal virtual work of an element is

$$W_e = \boldsymbol{\psi}'^{\mathrm{T}}\mathbf{k}'^{e}\mathbf{q}' = \boldsymbol{\psi}^{\mathrm{T}}\mathbf{L}^{\mathrm{T}}\mathbf{k}'^{e}\mathbf{L}\mathbf{q} \tag{8.51}$$

where $\boldsymbol{\psi}'$ and $\boldsymbol{\psi}$ are virtual nodal displacements in local and global coordinate systems, respectively. From Eq. 8.50 or 8.51, we recognize the element stiffness matrix in global coordinates to be

$$\boxed{\mathbf{k}^{e} = \mathbf{L}^{\mathrm{T}}\mathbf{k}'^{e}\mathbf{L}} \tag{8.52}$$

In the finite element program implementation, $\mathbf{k}'^{e}$ can first be defined, and then this matrix multiplication can be carried out.

If there is distributed load on a member, as shown in Fig. 8.10, we have

$$\mathbf{q}'^{\mathrm{T}}\mathbf{f}' = \mathbf{q}^{\mathrm{T}}\mathbf{L}^{\mathrm{T}}\mathbf{f}' \tag{8.53}$$

where

$$\mathbf{f}' = \left[0,\ \frac{p\ell_e}{2},\ \frac{p\ell_e^2}{12},\ 0,\ \frac{p\ell_e}{2},\ -\frac{p\ell_e^2}{12}\right]^{\mathrm{T}} \tag{8.54}$$

The nodal loads due to the distributed load p are given by

$$\mathbf{f} = \mathbf{L}^{\mathrm{T}}\mathbf{f}' \tag{8.55}$$

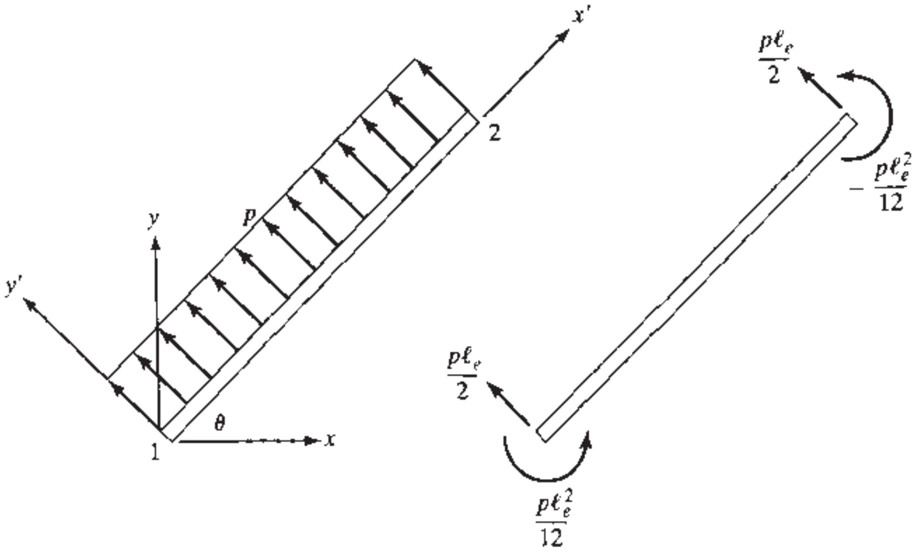


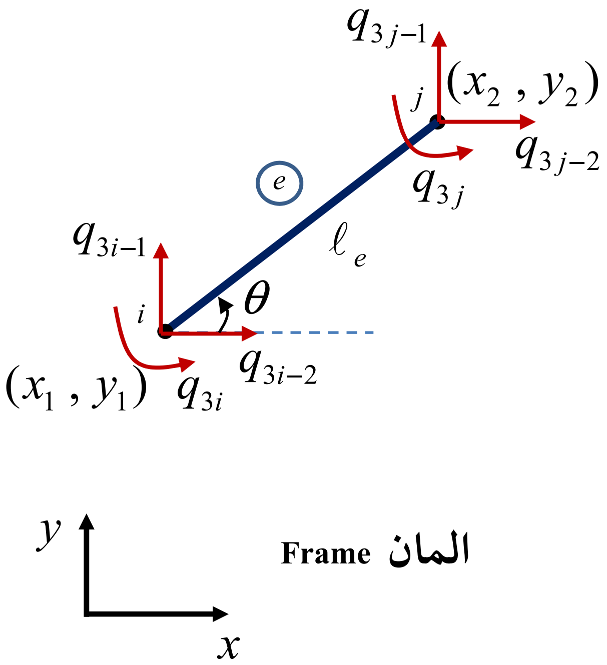
FIGURE 8.10 Distributed load on a frame element.

MDOF: Equations of Motion

II. محاسبه ماتریس‌های مشخصه‌های سازه‌ای - ماتریس سختی

۳) روش المان محدود برای تعیین ماتریس سختی.

حالت دوم : المان Frame



ماتریس تبدیل مختصات محلی به کلی به صورت زیر است:

$$\left[L^{(e)}\right]=\begin{bmatrix} \ell & m & 0 & 0 & 0 & 0 \\ -m & \ell & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ell & m & 0 \\ 0 & 0 & 0 & -m & \ell & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad (14)$$

MDOF: Equations of Motion

II. محاسبه ماتریس‌های مشخصه‌های سازه‌ای - ماتریس سختی

۳) روش المان محدود برای تعیین ماتریس سختی.

حالت دوم : المان Frame

ماتریس سختی المان در دستگاه مختصات محلی برابر است با:

$$\left[k^{(e)\prime}\right]=\begin{bmatrix} \frac{EA}{\ell_e} & 0 & 0 & -\frac{EA}{\ell_e} & 0 & 0 \\ 0 & \frac{12EI}{\ell_e^{\,3}} & \frac{6EI}{\ell_e^{\,2}} & 0 & -\frac{12EI}{\ell_e^{\,3}} & \frac{6EI}{\ell_e^{\,2}} \\ 0 & \frac{6EI}{\ell_e^{\,2}} & \frac{4EI}{\ell_e} & 0 & -\frac{6EI}{\ell_e^{\,2}} & \frac{2EI}{\ell_e} \\ -\frac{EA}{\ell_e} & 0 & 0 & \frac{EA}{\ell_e} & 0 & 0 \\ 0 & -\frac{12EI}{\ell_e^{\,3}} & -\frac{6EI}{\ell_e^{\,2}} & 0 & \frac{12EI}{\ell_e^{\,3}} & -\frac{6EI}{\ell_e^{\,2}} \\ 0 & \frac{6EI}{\ell_e^{\,2}} & \frac{2EI}{\ell_e} & 0 & -\frac{6EI}{\ell_e^{\,2}} & \frac{4EI}{\ell_e} \end{bmatrix} \qquad (15)$$

E : مدول الاستیسیته المان

A : سطح مقطع المان

I : ممان اینرسی المان

MDOF: Equations of Motion

II. محاسبه ماتریس‌های مشخصه‌های سازه‌ای – ماتریس سختی

۳) روش المان محدود برای تعیین ماتریس سختی.

حالت دوم : المان Frame

ماتریس سختی المان در دستگاه مختصات کلی به صورت زیر به دست می‌آید:

$$\left[k^{(e)}\right]=\left[L^{(e)}\right]^{T}\left[k^{(e)'}\right]\left[L^{(e)}\right]\quad\overset{(14)\,\&\,(15)}{\Rightarrow}$$

$$\left[k^{(e)}\right]=\frac{\mathrm{E}}{\ell_e}\left[\begin{array}{cccccc|c}
3i-2 & 3i-1 & 3i & 3j-2 & 3j-1 & 3j & \\
\left(\mathrm{A}\ell^2+\frac{12\mathrm{I}}{\ell_e{}^2}m^2\right) & \left(A-\frac{12\mathrm{I}}{\ell_e{}^2}\right)\ell m & -\frac{6\mathrm{I}}{\ell_e}m & -\left(A\ell^2+\frac{12\mathrm{I}}{\ell_e{}^2}m^2\right) & \left(\frac{12\mathrm{I}}{\ell_e{}^2}-A\right)\ell m & -\frac{6\mathrm{I}}{\ell_e}m & 3i-2 \\
\left(A-\frac{12\mathrm{I}}{\ell_e{}^2}\right)\ell m & \left(Am^2+\frac{12\mathrm{I}}{\ell_e{}^2}\ell^2\right) & \frac{6\mathrm{I}}{\ell_e}\ell & \left(\frac{12\mathrm{I}}{\ell_e{}^2}-A\right)\ell m & -\left(Am^2+\frac{12\mathrm{I}}{\ell_e{}^2}\ell^2\right) & \frac{6\mathrm{I}}{\ell_e}\ell & 3i-1 \\
-\frac{6\mathrm{I}}{\ell_e}m & \frac{6\mathrm{I}}{\ell_e}\ell & 4\mathrm{I} & \frac{6\mathrm{I}}{\ell_e}m & -\frac{6\mathrm{I}}{\ell_e}\ell & 2\mathrm{I} & 3i \\
-\left(A\ell^2+\frac{12\mathrm{I}}{\ell_e{}^2}m^2\right) & \left(\frac{12\mathrm{I}}{\ell_e{}^2}-A\right)\ell m & \frac{6\mathrm{I}}{\ell_e}m & \left(A\ell+\frac{12\mathrm{I}}{\ell_e{}^2}m^2\right) & \left(A-\frac{12\mathrm{I}}{\ell_e{}^2}\right)\ell m & \frac{6\mathrm{I}}{\ell_e}m & 3j-2 \\
\left(\frac{12\mathrm{I}}{\ell_e{}^2}-A\right)\ell m & -\left(Am^2+\frac{12\mathrm{I}}{\ell_e{}^2}\ell^2\right) & -\frac{6\mathrm{I}}{\ell_e}\ell & \left(A-\frac{12\mathrm{I}}{\ell_e{}^2}\right)\ell m & \left(Am^2-\frac{12\mathrm{I}}{\ell_e{}^2}\ell^2\right) & -\frac{6\mathrm{I}}{\ell_e}\ell & 3j-1 \\
-\frac{6\mathrm{I}}{\ell_e}m & \frac{6\mathrm{I}}{\ell_e}\ell & 2\mathrm{I} & \frac{6\mathrm{I}}{\ell_e}m & -\frac{6\mathrm{I}}{\ell_e}\ell & 4\mathrm{I} & 3j
\end{array}\right] \tag{16}$$

The values of **f** are added to the global load vector. Note here that positive p is in the y' direction.

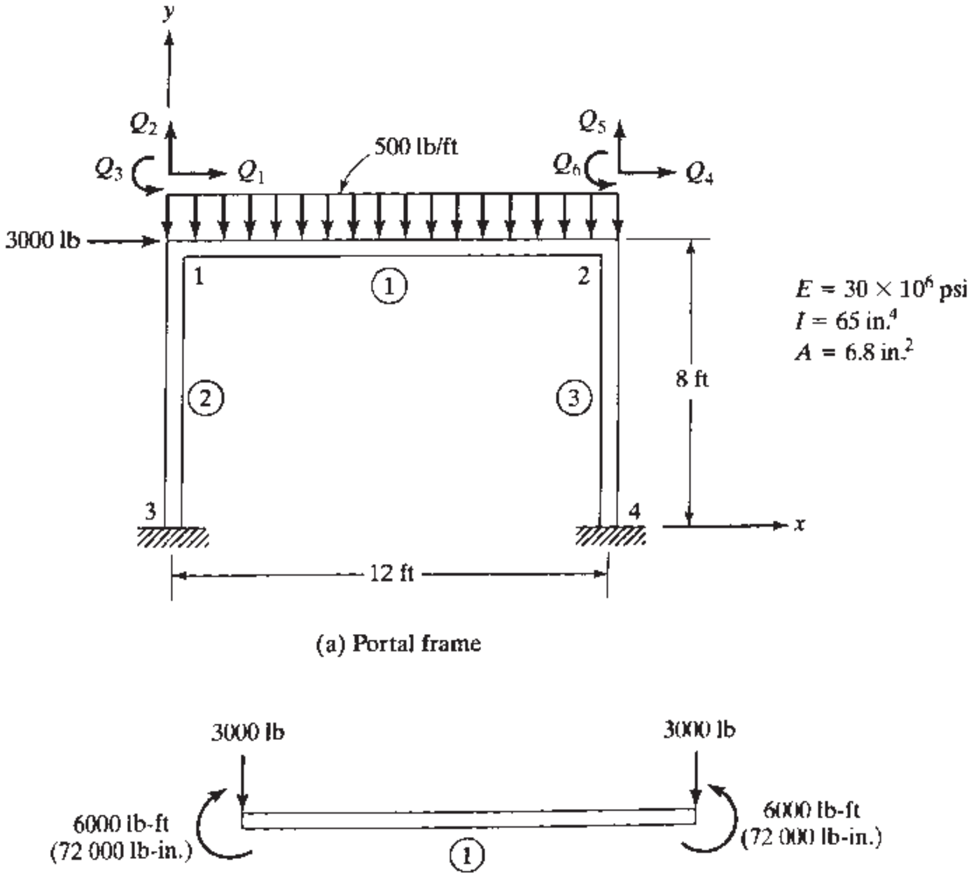
The point loads and couples are simply added to the global load vector. On gathering stiffnesses and loads, we get the system of equations

$$\mathbf{KQ} = \mathbf{F}$$

where the boundary conditions are considered by applying the penalty terms in the energy or Galerkin formulations.

Example 8.2

Determine the displacements and rotations of the joints for the portal frame shown in Fig. E8.2.



(a) Portal frame

(b) Equivalent load for element 1

FIGURE E8.2 (a) Portal frame. (b) Equivalent load for Element 1.

Solution We follow the steps given below:

Step 1. Connectivity

The connectivity is as follows:

	Node	
Element No.	1	2
1	1	2
2	3	1
3	4	2

Step 2. Element Stiffnesses

Element 1. Using the matrix given in Eq. 8.45 and noting that $\mathbf{k}^1 = \mathbf{k}'^1$, we find that

$$\mathbf{k}^1 = 10^4 \times \begin{array}{c} \begin{array}{cccccc} Q_1 & Q_2 & Q_3 & Q_4 & Q_5 & Q_6 \end{array} \\ \begin{bmatrix} 141.7 & 0 & 0 & -141.7 & 0 & 0 \\ 0 & 0.784 & 56.4 & 0 & -0.784 & 56.4 \\ 0 & 56.4 & 5417 & 0 & -56.4 & 2708 \\ -141.7 & 0 & 0 & 141.7 & 0 & 0 \\ 0 & -0.784 & -56.4 & 0 & 0.784 & -56.4 \\ 0 & 56.4 & 2708 & 0 & -56.4 & 5417 \end{bmatrix} \end{array}$$

Elements 2 and 3. Local element stiffnesses for elements 2 and 3 are obtained by substituting for E, A, I and ℓ_2 in matrix $\mathbf{k}'$ of Eq. 8.49:

$$\mathbf{k}'^2 = 10^4 \times \begin{bmatrix} 212.5 & 0 & 0 & -212.5 & 0 & 0 \\ 0 & 2.65 & 127 & 0 & -2.65 & 127 \\ 0 & 127 & 8125 & 0 & -127 & 4063 \\ -212.5 & 0 & 0 & 212.5 & 0 & 0 \\ 0 & -2.65 & -127 & 0 & 2.65 & -127 \\ 0 & 127 & 4063 & 0 & -127 & 8125 \end{bmatrix}$$

Transformation matrix **L**. We have noted that for element 1, $\mathbf{k} = \mathbf{k}^1$ For elements 2 and 3, which are oriented similarly with respect to the x- and y-axes, we have $\ell = 0, m = 1$. Then,

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Noting that $\mathbf{k}^2 = \mathbf{L}^{\mathrm{T}}\mathbf{k}'^2\mathbf{L}$, we get

$$\begin{array}{llll} e = 3 & Q_4 & Q_5 & Q_6 \\ e = 2 \rightarrow & Q_1 & Q_2 & Q_3 \end{array}$$

$$\mathbf{k} = 10^4 \times \begin{bmatrix} 2.65 & 0 & -127 & -2.65 & 0 & -127 \\ 0 & 212.5 & 0 & 0 & -212.5 & 0 \\ -127 & 0 & 8125 & 127 & 0 & 4063 \\ -2.65 & 0 & 127 & 2.65 & 0 & 127 \\ 0 & -212.5 & 0 & 0 & 212.5 & 0 \\ -127 & 0 & 4063 & 127 & 0 & 8125 \end{bmatrix}$$

Stiffness $\mathbf{k}^1$ has all its elements in the global locations. For elements 2 and 3, the shaded part of the stiffness matrix shown previously is added to the appropriate global locations of $\mathbf{K}$. The global stiffness matrix is given by

$$\mathbf{K} = 10^4 \times \begin{bmatrix} 144.3 & 0 & 127 & -141.7 & 0 & 0 \\ 0 & 213.3 & 56.4 & 0 & -0.784 & 56.4 \\ 127 & 56.4 & 13542 & 0 & -56.4 & 2708 \\ -141.7 & 0 & 0 & 144.3 & 0 & 127 \\ 0 & -0.784 & -56.4 & 0 & 213.3 & -56.4 \\ 0 & 56.4 & 2708 & 127 & -56.4 & 13542 \end{bmatrix}$$

From Fig. E8.2, the load vector can easily be written as

$$\mathbf{F} = \begin{Bmatrix} 3\,000 \\ -3\,000 \\ -72\,000 \\ 0 \\ -3\,000 \\ +72\,000 \end{Bmatrix}$$

The set of equations is given by

$$\mathbf{KQ} = \mathbf{F}$$

On solving, we get

$$\mathbf{Q} = \begin{Bmatrix} 0.092 \text{ in.} \\ -0.00104 \text{ in.} \\ -0.00139 \text{ rad} \\ 0.0901 \text{ in.} \\ -0.0018 \text{ in.} \\ -3.88 \times 10^{-5} \text{ rad} \end{Bmatrix}$$

■

8.8 THREE-DIMENSIONAL FRAMES

Three-dimensional frames, also called as *space frames*, are frequently encountered in the analysis of multistory buildings. They are also to be found in the modeling of car body and bicycle frames. A typical three-dimensional frame is shown in Fig. 8.11. Each node has six degrees of freedom (dofs) (as opposed to only three dofs in a plane frame). The dof numbering is shown in Fig. 8.11: for node I, dof $6I$–5, $6I$–4, and $6I$–3 represent

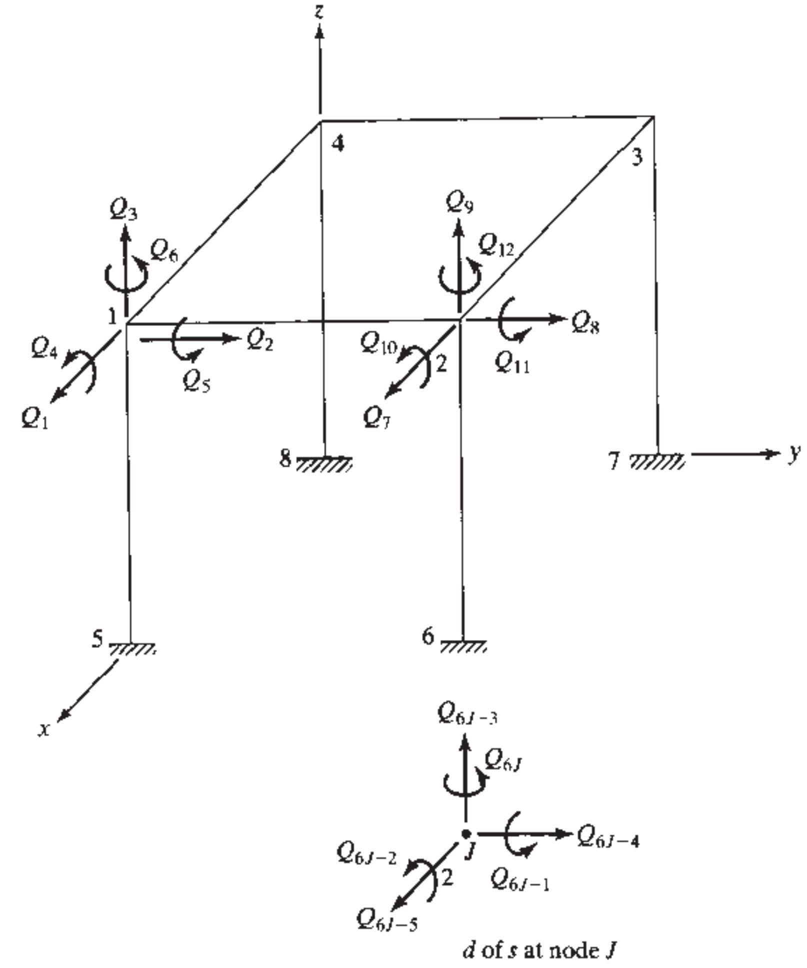
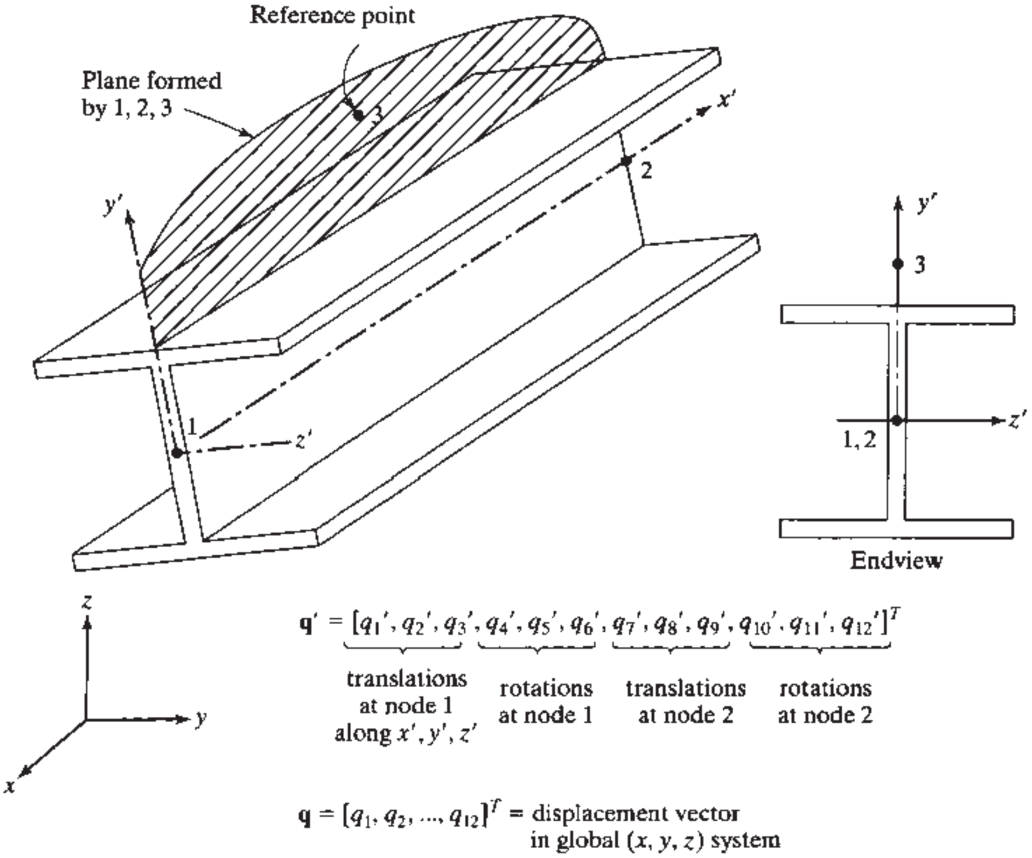


FIGURE 8.11 Degrees of freedom numbering for a three-dimensional frame.

the x-, y-, and z-translational dofs, while $6J$–2, $6J$–1, and $6J$ represent the rotational dofs along the x-, y-, and z-axes. The element displacement vectors in the local and global coordinate systems are denoted as $\mathbf{q}'$ and $\mathbf{q}$, respectively. These vectors are of dimension (12×1) as shown in Fig. 8.12.

Orientation of the local x'-, y'-, and z'-coordinate system is established with the use of three points. Points 1 and 2 are the ends of the element; the x'-axis is along the line from point 1 to point 2, just as in the case of two-dimensional frames. Point 3 is any *reference point* not lying along the line joining points 1 and 2. The y'-axis is to lie in the plane defined by points 1, 2, and 3. This is shown in Fig. 8.12. The z'-axis is then automatically defined from the fact that x', y', and z' form a right-handed system. We note that y', and z' are the principal axes of the cross section, with $I_{y'}$ and $I_{z'}$ the principal moments of inertia. The cross-sectional properties are specified by four parameters: area A and moments of inertia $I_{y'}$, $I_{z'}$, and J. The product GJ is the torsional stiffness, where G = shear modulus. For circular or tubular cross sections, J is the polar moment of inertia. For other cross-sectional shapes, such as an I-section, the torsional stiffness is given in strength of materials texts.



$$\mathbf{q}' = [\underbrace{q_1', q_2', q_3'}, \underbrace{q_4', q_5', q_6'}, \underbrace{q_7', q_8', q_9'}, \underbrace{q_{10}', q_{11}', q_{12}'}]^T$$

$$\mathbf{q} = [q_1, q_2, \ldots, q_{12}]^T = \text{displacement vector in global } (x, y, z) \text{ system}$$

FIGURE 8.12 Three-dimensional frame element in local and global coordinate systems.

The (12×12) element stiffness matrix $\mathbf{k}'$ in the local coordinate system is obtained by a straightforward generalization of Eq. 8.49 as

$$\mathbf{k}' = \begin{bmatrix} AS & 0 & 0 & 0 & 0 & 0 & -AS & 0 & 0 & 0 & 0 & 0 \\ & a_{z'} & 0 & 0 & 0 & b_{z'} & 0 & -a_{z'} & 0 & 0 & 0 & b_{z'} \\ & & a_{y'} & 0 & -b_{y'} & 0 & 0 & 0 & -a_{y'} & 0 & -b_{y'} & 0 \\ & & & TS & 0 & 0 & 0 & 0 & 0 & -TS & 0 & 0 \\ & & & & c_{y'} & 0 & 0 & 0 & b_{y'} & 0 & d_{y'} & 0 \\ & & & & & c_{z'} & 0 & -b_{z'} & 0 & 0 & 0 & d_{z'} \\ & & & & & & AS & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & a_{z'} & 0 & 0 & 0 & -b_{z'} \\ & & & & & & & & c_{y'} & 0 & b_{y'} & 0 \\ & & & & & & & & & TS & 0 & 0 \\ \text{Symmetric} & & & & & & & & & & c_{y'} & 0 \\ & & & & & & & & & & & c_{z'} \end{bmatrix} \tag{8.56}$$

where $AS = EA/l_e$, l_e = length of the element, $TS = GJ/l_e$, $a_{z'} = 12EI_{z'}/l_e^3$, $b_{z'} = 6EI_{z'}/l_e^3$, $c_{z'} = 4EI_{z'}/l_e$, $d_{z'} = 2EI_{z'}/l'_e$, $a_{y'} = 12EI_{y'}/l_e^3$, and so on. The global–local transformation matrix is given by

$$\mathbf{q}' = \mathbf{L}\mathbf{q} \tag{8.57}$$

The (12×12) transformation matrix $\mathbf{L}$ is defined from a $(3 \times 3)\boldsymbol{\lambda}$ matrix as

$$\mathbf{L} = \begin{bmatrix} \boldsymbol{\lambda} & & & \mathbf{0} \\ & \boldsymbol{\lambda} & & \\ & & \boldsymbol{\lambda} & \\ \mathbf{0} & & & \boldsymbol{\lambda} \end{bmatrix} \tag{8.58}$$

The $\boldsymbol{\lambda}$ is a matrix of direction cosines:

$$\boldsymbol{\lambda} = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{bmatrix} \tag{8.59}$$

Here, l_1, m_1, and n_1 are the cosines of the angles between the x'-axis and the global x-, y-, and z-axes, respectively. Similarly, l_2, m_2, and n_2, are the cosines of the angles between the y'-axis and the x-, y-, and z-axes, and l_3, m_3, and n_3 are associated with the z'-axis. These direction cosines and hence the $\boldsymbol{\lambda}$ matrix are obtainable from the coordinates of the points 1, 2, and 3 as follows. We have

$$l_1 = \frac{x_2 - x_1}{l_e} \qquad m_1 = \frac{y_2 - y_1}{l_e} \qquad n_1 = \frac{z_2 - z_1}{l_e}$$
$$l_e = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

Now, let $\mathbf{V}_{x'} = [l_1 \quad m_1 \quad n_1]^{\mathrm{T}}$ denote the unit vector along the x'-axis. Also, let

$$\mathbf{V}_{13} = \left[\frac{x_3 - x_1}{l_{13}} \quad \frac{y_3 - y_1}{l_{13}} \quad \frac{z_3 - z_1}{l_{13}}\right]$$

where l_{13} = distance between points 1 and 3. The unit vector along the z'-axis is now given by

$$\mathbf{V}_{z'} = [l_3 \quad m_3 \quad n_3]^{\mathrm{T}} = \frac{\mathbf{V}_{x'} \times \mathbf{V}_{13}}{|\mathbf{V}_{x'} \times \mathbf{V}_{13}|}$$

The cross product of any two vectors is given by the determinant

$$\mathbf{u} \times \mathbf{v} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ u_x & u_y & u_z \\ v_x & v_y & v_z \end{vmatrix} = \begin{vmatrix} u_y v_z - v_y u_z \\ v_x u_z - u_x v_z \\ u_x v_y - v_x u_y \end{vmatrix}$$

Finally, we have the direction cosines of the y'-axis given by

$$\mathbf{V}_{y'} = [l_2 \quad m_2 \quad n_2]^{\mathrm{T}} = \mathbf{V}_{z'} \times \mathbf{V}_{x'}$$

These calculations to define the $\mathbf{L}$ matrix are coded in program FRAME3D. The element stiffness matrix in global coordinates is

$$\mathbf{k} = \mathbf{L}^{\mathrm{T}}\mathbf{k}'\mathbf{L} \tag{8.60}$$

where $\mathbf{k}'$ has been defined in Eq. 8.56.

If a distributed load with components $w_{y'}$ and $w_{z'}$ (units of force/unit length) is applied on the element, then the equivalent point loads at the ends of the member are

$$\mathbf{f}' = \left[0, \frac{w_{y'}l_e}{2}, \frac{w_{z'}l_e}{2}, 0, \frac{-w_{z'}l_e^2}{12}, \frac{w_{y'}l_e^2}{12}, 0, \frac{w_{y'}l_e}{2}, \frac{w_{z'}l_e}{2}, 0, \frac{w_{z'}l_e^2}{12}, \frac{-w_{y'}l_e^2}{12}\right]^{\mathrm{T}} \tag{8.61}$$

These loads are transferred into global components by $\mathbf{f} = \mathbf{L}^{\mathrm{T}}\mathbf{f}'$. After enforcing boundary conditions and solving the system equations $\mathbf{KQ} = \mathbf{F}$, we can compute the member end forces from

$$\mathbf{R}' = \mathbf{k}'\mathbf{q}' + \text{fixed-end reactions} \tag{8.62}$$

where the fixed-end reactions are the negative of the $\mathbf{f}'$ vector and are only associated with those elements having distributed loads acting on them. The member end forces provide the bending moments and shear forces from which the beam stresses can be determined.

Example 8.3

Figure E8.3 shows a three-dimensional frame subjected to various loads. Our task is to run program FRAME3D to obtain the maximum bending moments in the structure. The input and output files are as given in the third data set, which follows the BEAM and FRAME2D data sets. From the output, we obtain the maximum $M_{y'} = 3.680\text{E} + 0.5$ N · m occurring in member 1 at node 1 (the first node) and maximum $M_{z'} = -1.413\text{E} + 0.5$ N · m occurring in member 3 at node 4. ■

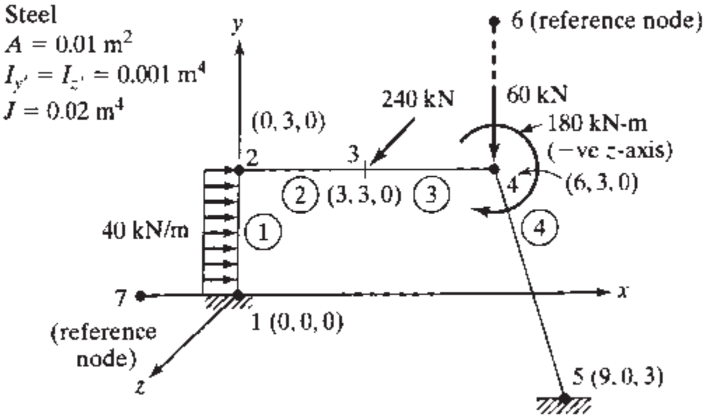


FIGURE E8.3

8.9 SOME COMMENTS

Symmetric beams and plane and space frames have been discussed in this chapter. In engineering applications, there are several challenging problems. such as frames and mechanisms with pin-jointed members, unsymmetric beams, buckling of members due to axial loads, shear considerations, and structures with large deformations. For help in formulating and analyzing such problems, the reader may refer to some advanced publications in mechanics of solids, structural analysis. elasticity and plasticity, and finite element analysis.

Input Data File

```
<< Beam Analysis >>
EXAMPLE 8.1
NN NE NM NDIM NEN NDN
 3  2  1  1    2   2
ND NL   NMPC
 4  4    0
Node#  Coordinates
 1        0
 2     1000
 3     2000
Elem#  N1   N2  Mat#  Mom_Inertia
 1      1    2    1     4e6
 2      2    3    1     4e6
DOF#  Displacement
 1     0
 2     0
 3     0
 5     0
DOF#  Load
 3     -6000
 4     -1e6
 5     -6000
 6      1e6
MAT#    E
 1    200000
 Multi-point Constraints  B1*Qi+B2*Qj=B3
```

```
Program Beam - CHANDRUPATLA & BELEGUNDU
Output
EXAMPLE 8.1
NODE#    Displ.        Rotation(radians)
    1  2.0089E-11    6.6961E-09
    2  -1.2723E-10  -2.6786E-04
    3  -8.0357E-11   4.4643E-04
DOF#    Reaction
    1  -1.2857E+03
    2  -4.2855E+05
    3  8.1428E+03
    5  5.1429E+03
```

```
<<2-D  Frame Analysis >>
EXAMPLE 8.2
NN NE NM NDIM NEN NDN
4   3  1  2    2   3
ND NL   NMPC
6  1     0
Node#   X    Y
 1      0   96
 2    144   96
 3      0    0
 4    144    0
```

continued

```
ELEM#  N1  N2  MAT#  Area  Inertia  Distr_load
 1      2   1   1    6.8    65       41.6667
 2      3   1   1    6.8    65         0.
 3      4   2   1    6.8    65         0.
DOF#  Displacement
 7      0
 8      0
 9      0
10      0
11      0
12      0
DOF#  Load
 1    3000
MAT#   E
 1    30e6
B1  i  B2 j  B3  (Multi-point constr. B1*Qi+B2*Qj=B3)
```

```
Program Frame2D - CHANDRUPATLA & BELEGUNDU
Output
EXAMPLE 8.2
NODE#   X-Displ      Y-Displ     Z-Rotation
    1  9.1770E-02  -1.0358E-03  -1.3874E-03
    2  9.0122E-02  -1.7877E-03  -3.8835E-05
    3  4.9167E-10  -1.6255E-09  -4.4410E-08
    4  1.7237E-09  -2.8053E-09  -8.3320E-08
Member End-Forces
Member#  1
2.3342E+03  -7.9884E+02  -3.9255E+04
-2.3342E+03  7.9884E+02  -7.5778E+04
Member#  2
2.2012E+03  6.6580E+02  6.0139E+04
-2.2012E+03  -6.6580E+02  3.7778E+03
Member#  3
3.7988E+03  2.3342E+03  1.1283E+05
-3.7988E+03  -2.3342E+03  1.1125E+05
DOF#           Reaction
    7  -6.6580E+02
    8  2.2012E+03
    9  6.0139E+04
   10  -2.3342E+03
   11  3.7988E+03
   12  1.1283E+05
```

```
<<3-D Frame Analysis >>
EXAMPLE 8.3
NN NE NM NDIM NEN NDN   NNREF
 5  4  1  3    2   6     2
ND NL  NMPC
12  3   0
Node#  X   Y   Z
 1     0   0   0
 2     0   3   0
 3     3   3   0
```

continued

```
4      6   3   0
 5      9   0   3
 6      6   6   0
 7     -3   0   0
Elem#  N1  N2  Ref_Pt  Mat#   Area    Iy     Iz      J      UDLy'  UDLz'
 1      1   2    7      1     .01    1E-3   1E-3   2E-3  -40000.  0.
 2      2   3    6      1     .01    1E-3   1E-3   2E-3    0.     0.
 3      3   4    6      1     .01    1E-3   1E-3   2E-3    0.     0.
 4      4   5    6      1     .01    1E-3   1E-3   2E-3    0.     0.
DOF#  Displacement
 1       0
 2       0
 3       0
 4       0
 5       0
 6       0
25       0
26       0
27       0
28       0
29       0
30       0
DOF#  Load
15   240000
20  -60000
24  -180000
MAT# Prop1(E)  Prop2(G)
 1   200E9     80E9
B1  i  B2 j  B3  (Multi-point constr. B1*Qi+B2*Qj=B3)
```

```
Program Frame3D - CHANDRUPATLA & BELEGUNDU
Output
EXAMPLE 8.3
Node#)  X-Displ     Y-Displ     Z-Displ     X-Rot       Y-Rot       Z-Rot
 1 ) 3.127E-09  1.972E-09  9.900E-09  2.760E-08  -7.145E-09  5.348E-09
 2 ) -1.868E-03  3.944E-05  5.310E-03  2.550E-03  -1.786E-03  1.108E-03
 3 ) -1.985E-03  3.141E-03  9.842E-03  2.025E-03  -2.452E-04  7.624E-04
 4 ) -2.103E-03  3.431E-03  6.241E-03  1.500E-03  1.836E-03  -7.662E-04
 5 ) 5.873E-09  -6.472E-09  8.100E-09  6.985E-09  8.429E-09  -1.101E-09
Member End-Forces
Member#  1
-2.629E+04  -1.830E+04  -1.320E+05  9.526E+04  3.680E+05  -1.013E+05
2.629E+04  1.830E+04  1.320E+05  -9.526E+04  2.800E+04  4.641E+04
Member#  2
7.830E+04  -2.629E+04  -1.320E+05  2.800E+04  9.526E+04  -1.641E+04
-7.830E+04  2.629E+04  1.320E+05  -2.800E+04  3.007E+05  -6.247E+04
Member#  3
7.830E+04  -2.629E+04  1.080E+05  2.800E+04  -3.007E+05  6.247E+04
-7.830E+04  2.629E+04  -1.080E+05  -2.800E+04  -2.328E+04  -1.413E+05
Member#  4
1.574E+05  5.600E+03  2.100E+04  -1.959E+04  1.465E+04  -4.713E+04
-1.574E+05  -5.600E+03  -2.100E+04  1.959E+04  -1.238E+05  7.623E+04
```

PROBLEMS

8.1. Find the deflection at the load and the slopes at the ends for the steel shaft shown in Fig. P8.1. Consider the shaft to be simply supported at bearings A and B.

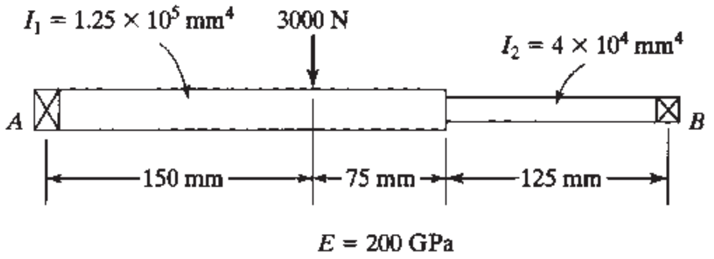


FIGURE P8.1 Problems 8.1 and 8.4.

8.2. A three-span beam is shown in Fig. P8.2. Determine the deflection curve of the beam and evaluate the reactions at the supports.

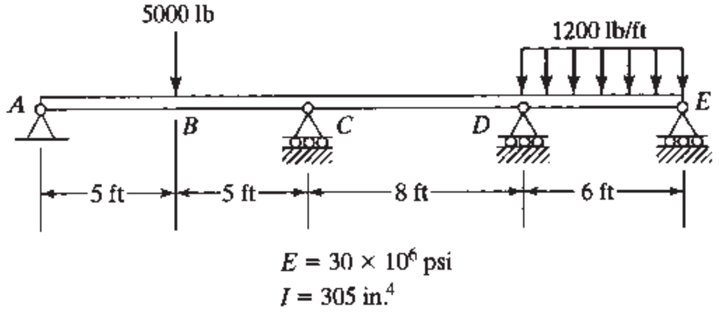


FIGURE P8.2

8.3. A reinforced concrete slab floor is shown in Fig. P8.3. Using a unit width of the slab in the z direction, determine the deflection curve of the neutral surface under its own weight.

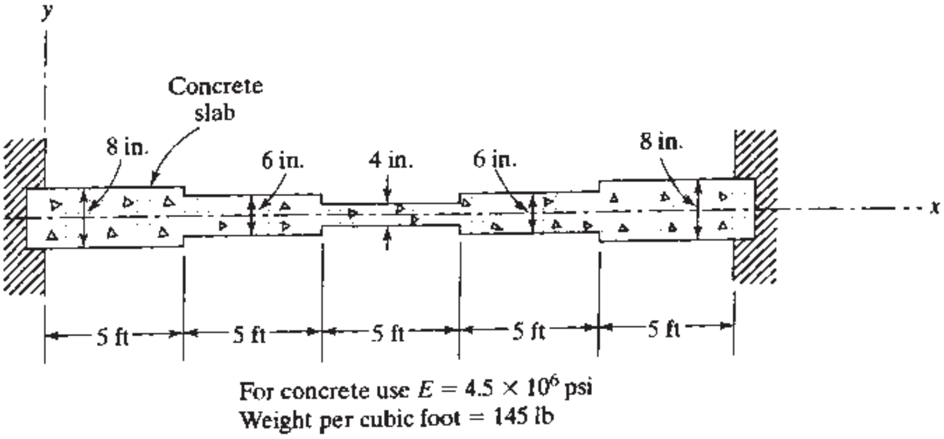
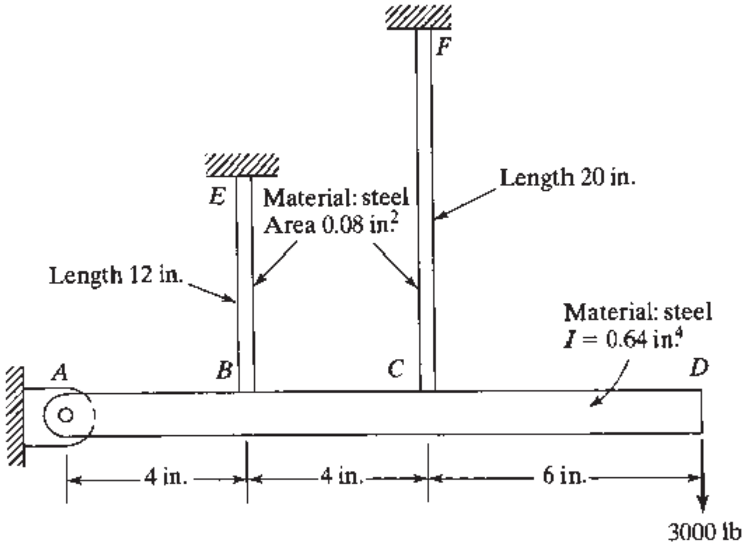

y
Concrete slab
8 in.
6 in.
4 in.
6 in.
8 in.
x
5 ft
5 ft
5 ft
5 ft
5 ft
For concrete use $E = 4.5 \times 10^6$ psi
Weight per cubic foot = 145 lb

FIGURE P8.3

8.4. In the shaft shown in Fig. P8.1, determine the deflection at the loads and the slopes at the ends if the bearings at A and B have radial stiffnesses of 20 and 12 kN/mm, respectively.

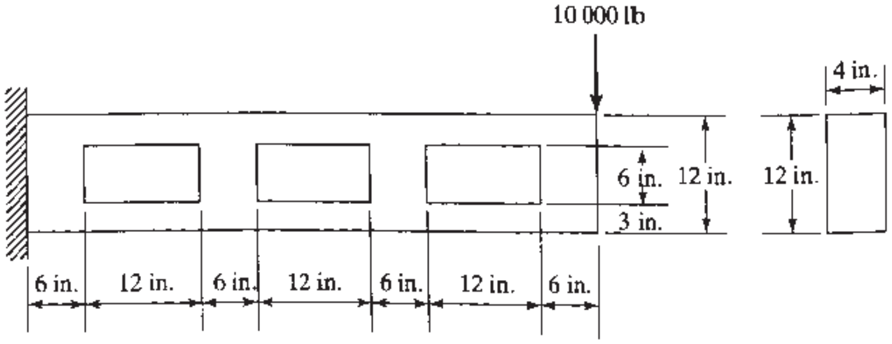
8.5. Figure P8.5 shows a beam AD pinned at A and welded at B and C to long and slender rods BE and CF. A load of 3000 lb is applied at D as shown. Model the beam AD using beam elements and determine deflections at B, C, and D and stresses in rods BE and CF.



E for steel = 30 × 10⁶ psi

FIGURE P8.5

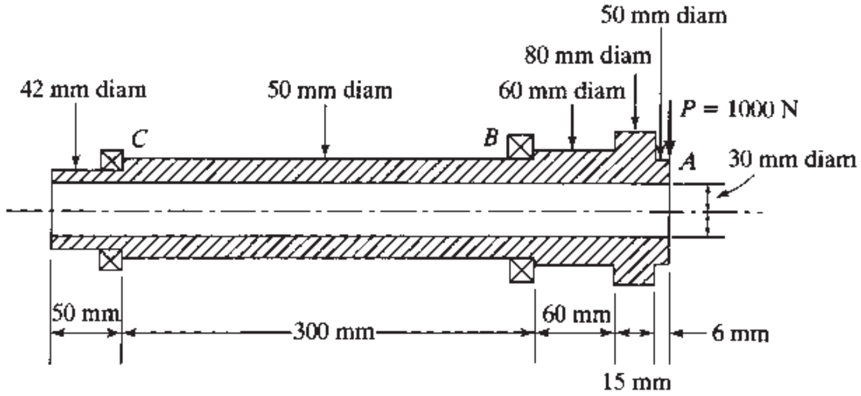
8.6. Figure P8.6 shows a cantilever beam with three rectangular openings. Find the deflections for the beam shown and compare the deflections with a beam without openings.



$E = 30 \times 10^6$ psi

FIGURE P8.6

8.7. A simplified section of a machine tool spindle is shown in Fig. P8.7. Bearing B has a radial stiffness of 60 N/μm and a rotational stiffness (against moment) of 8 × 10⁵ N.m/rad. Bearing C has a radial stiffness of 20 N/μm and its rotational stiffness can be neglected. For a load of 1000 N, as shown, determine the deflection and slope at A. Also, give the deflected shape of the spindle center line (1 μm = 10^{-6} m).



Machine tool spindle

FIGURE P8.7

8.8. Determine the deflection at the center of BC for the frame shown in Fig. P8.8, using program FRAME2D. Also determine the reactions at A and D.

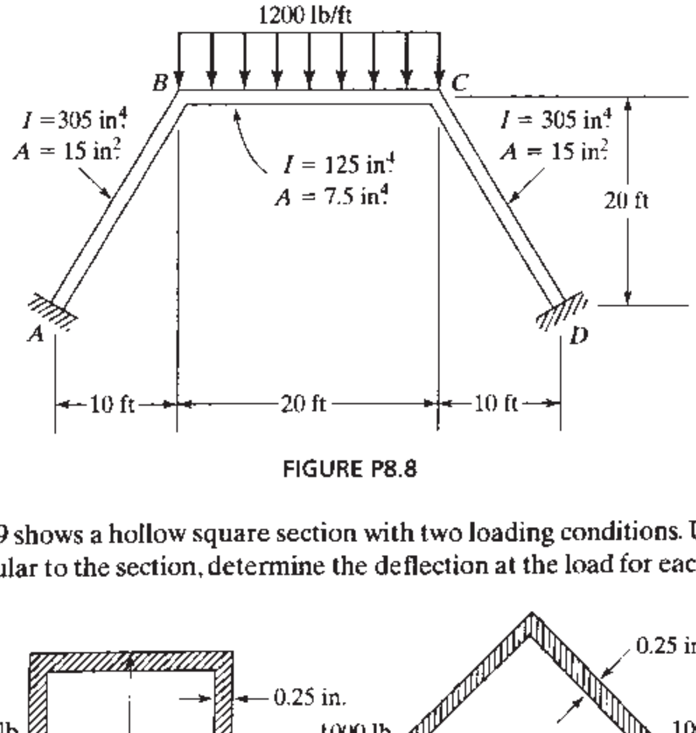


FIGURE P8.8

8.9. Figure P8.9 shows a hollow square section with two loading conditions. Using a 1-in. width perpendicular to the section, determine the deflection at the load for each of the two cases.

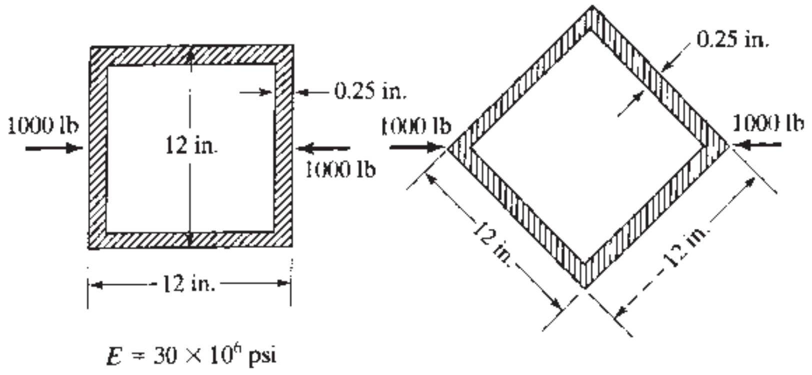


FIGURE P8.9

8.10. Figure P8.10 shows a five-member steel frame subjected to loads at the free end. The cross section of each member is a tube of wall thickness $t = 1$ cm and mean radius $R = 6$ cm. Determine the following:

(a) the displacement of node 3 and

(b) the maximum axial compressive stress in a member.

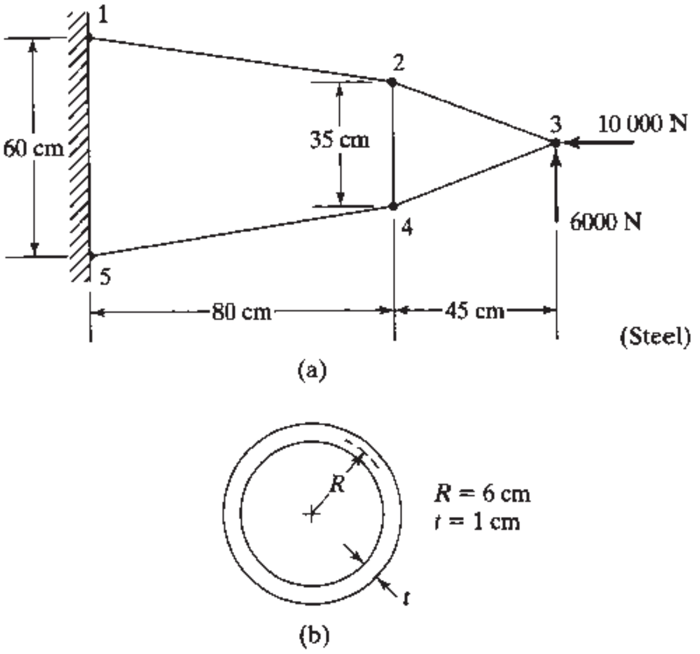


FIGURE P8.10

8.11. Dimensions of a common paper staple are shown in Fig. P8.11. While the staple is penetrating into the paper, a force of about 120 N is applied. Find the deformed shape for the following cases:

(a) load uniformly distributed on the horizontal member and pinned condition at A at entry;

(b) load as in (a) with fixed condition at A after some penetration;

(c) load divided into two point loads, with A pinned; and

(d) load as in (c) with A fixed.

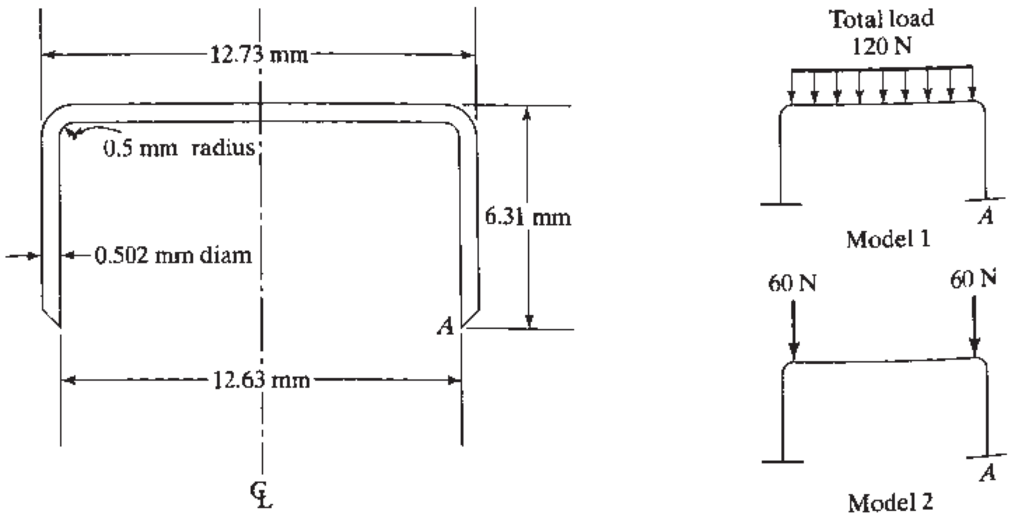


FIGURE P8.11

8.12. A commonly used street light arrangement is shown in Fig. P8.12. Assuming fixed condition at A, compare the deformed shapes for the following two cases:

(a) without the rod BC (that is, only member ACD supports the light) and

(b) with tie rod BC.

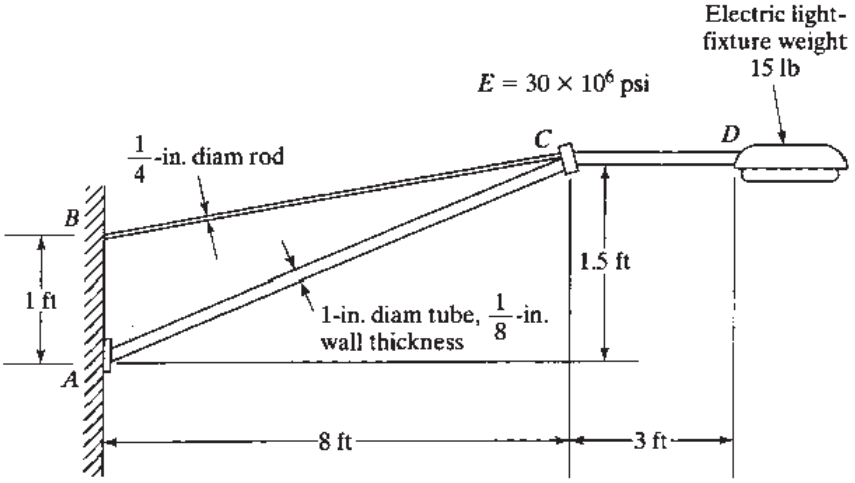


FIGURE P8.12

8.13. Figure P8.13a shows a cab of a van. A simplified finite element frame model is shown in Fig. P8.13b. The model consists of 28 nodes. x–z is a plane of symmetry; thus, nodes $1' - 13'$ have the same x- and z-coordinates as nodes 1–13, with y-coordinates reversed in sign. Each beam element is made of steel with $A = 0.2$ in.^{2}, $I_{y'} = I_{z'} = 0.003$ in.^{4}, and $J = 0.006$ in.^{4}. The loading corresponds to a frontal impact test based on Swedish standards and consists of a load at node 1 (only) with components $F_x = -3194.0$ lb and $F_y = -856.0$ lb. Treat nodes $11, 11'$, 12, and $12'$ as fixed (boundary conditions). Nodal coordinates in inches are as follows:

Node	x	y	z	Node	x	y	z
1	58.0	38.0	0	9	0	38.0	75.0
2	48.0	38.0	0	10	58.0	17.0	42.0
3	31.0	38.0	0	11	58.0	17.0	0
4	17.0	38.0	22.0	12	0	17.0	0
5	0	38.0	24.0	13	0	17.0	24.0
6	58.0	38.0	42.0	14	18.0	0	72.0
7	48.0	38.0	42.0	15	0	0	37.5
8	36.0	38.0	70.0				

(*Note*: Number the nodes to keep bandwidth to a minimum.)

Determine the deflections at nodes 1, 2, 6, 7, 10, and 11 and the location and magnitude of the maximum bending moments using program FRAME3D.

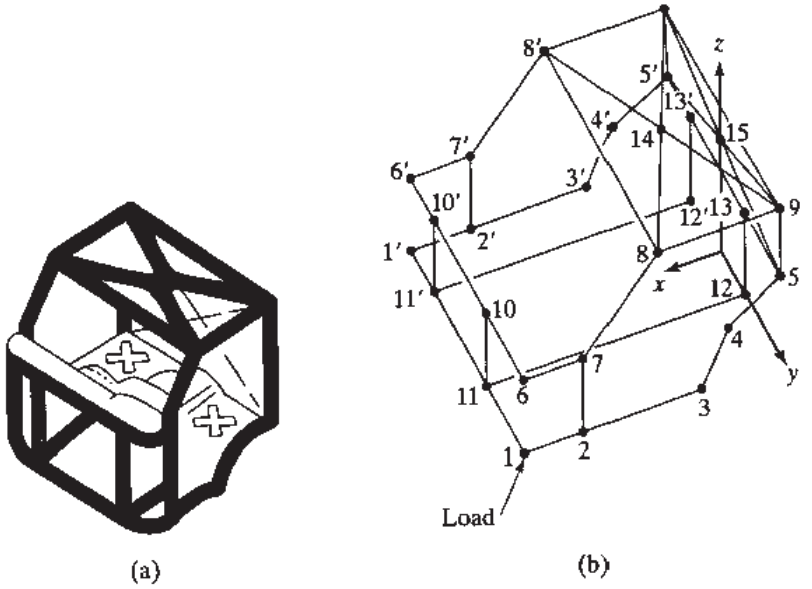
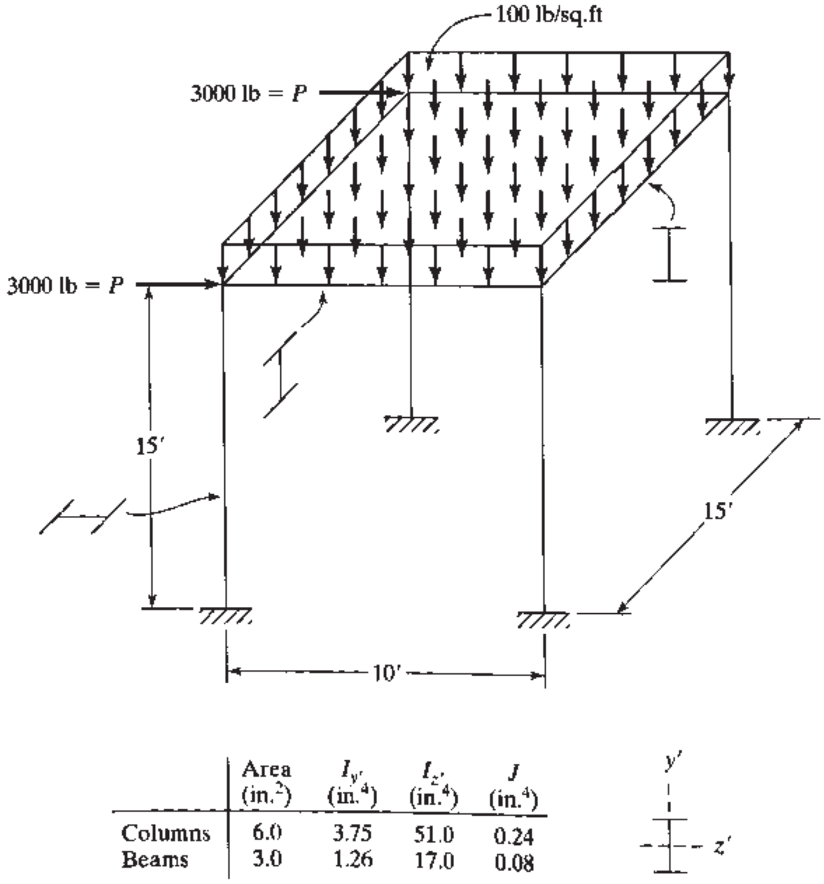


FIGURE P8.13 (a) Van Frame. (b) Frame finite element model.

8.14. Consider the steel frame in Figure P8.14, which is subjected to a wind load and roof load as shown. Determine the bending moments in the structure (maximum $M_{y'}$ and $M_{z'}$).



	Area (in.^{2})	$I_{y'}$ (in.^{4})	$I_{z'}$ (in.^{4})	J (in.^{4})
Columns	6.0	3.75	51.0	0.24
Beams	3.0	1.26	17.0	0.08

FIGURE P8.14

Program Listings

```
'*************************************
'*           PROGRAM BEAM            *
'*        Beam Bending Analysis      *
'* T.R.Chandrupatla and A.D.Belegundu *
'*************************************
'============ MAIN PROGRAM ===============
Private Sub cmdStart_Click()
     Call InputData
     Call Bandwidth
     Call Stiffness
     Call ModifyForBC
     Call BandSolver
     Call ReactionCalc
     Call Output
     cmdView.Enabled = True
     cmdStart.Enabled = False
End Sub
```

```
'============== ELEMENT STIFFNESS AND ASSEMBLY =============
Private Sub Stiffness()
     ReDim S(NQ, NBW)
     '----- Global Stiffness Matrix -----
     For N = 1 To NE
         picBox.Print "Forming Stiffness Matrix of Element "; N
         N1 = NOC(N, 1)
         N2 = NOC(N, 2)
         M = MAT(N)
         EL = Abs(X(N1) - X(N2))
         EIL = PM(M, 1) * SMI(N) / EL ^ 3
         '----- Element Stiffness Matrix -----
         SE(1, 1) = 12 * EIL
         SE(1, 2) = EIL * 6 * EL
         SE(1, 3) = -12 * EIL
         SE(1, 4) = EIL * 6 * EL
            SE(2, 1) = SE(1, 2)
            SE(2, 2) = EIL * 4 * EL * EL
            SE(2, 3) = -EIL * 6 * EL
            SE(2, 4) = EIL * 2 * EL * EL
         SE(3, 1) = SE(1, 3)
         SE(3, 2) = SE(2, 3)
         SE(3, 3) = EIL * 12
         SE(3, 4) = -EIL * 6 * EL
            SE(4, 1) = SE(1, 4)
            SE(4, 2) = SE(2, 4)
            SE(4, 3) = SE(3, 4)
            SE(4, 4) = EIL * 4 * EL * EL
         picBox.Print ".... Placing in Global Locations"
         Call PlaceGlobal(N)
     Next N
End Sub
'===========================================================
'********     PROGRAM FRAME2D      ********
'*       2-D   FRAME ANALYSIS BY FEM      *
'*  T.R.Chandrupatla and A.D.Belegundu    *
'*****************************************
'============ MAIN PROGRAM ===============
Private Sub cmdStart_Click()
     Call InputData
     Call Bandwidth
     Call Stiffness
     Call AddLoads
     Call ModifyForBC
     Call BandSolver
     Call EndActions
     Call ReactionCalc
     Call Output
     cmdView.Enabled = True
     cmdStart.Enabled = False
End Sub
'================================================
```

```
'============== ELEMENT STIFFNESS AND ASSEMBLY =============
Private Sub Stiffness()
     ReDim S(NQ, NBW)
     '----- Global Stiffness Matrix -----
     For N = 1 To NE
        picBox.Print "Forming Stiffness Matrix of Element "; N
        ISTF = 2
        Call Elstif(N)
        picBox.Print ".... Placing in Global Locations"
        Call PlaceGlobal(N)
     Next N
End Sub
```

```
'============== ELEMENT STIFFNESS =============
Private Sub Elstif(N)
     '----- Element Stiffness Matrix -----
     I1 = NOC(N, 1): I2 = NOC(N, 2): M = MAT(N)
     X21 = X(I2, 1) - X(I1, 1)
     Y21 = X(I2, 2) - X(I1, 2)
     EL = Sqr(X21 * X21 + Y21 * Y21)
     EAL = PM(M, 1) * ARIN(N, 1) / EL
     EIZL = PM(M, 1) * ARIN(N, 2) / EL
     For I = 1 To 6
       For J = 1 To 6
         SEP(I, J) = 0!
       Next J
     Next I
```

continued

```
     SEP(1, 1) = EAL: SEP(1, 4) = -EAL: SEP(4, 4) = EAL
     SEP(2, 2) = 12 * EIZL / EL ^ 2: SEP(2, 3) = 6 * EIZL / EL
     SEP(2, 5) = -SEP(2, 2): SEP(2, 6) = SEP(2, 3)
     SEP(3, 3) = 4 * EIZL: SEP(3, 5) = -6 * EIZL / EL: SEP(3, 6) = 2 * EIZL
     SEP(5, 5) = 12 * EIZL / EL ^ 2: SEP(5, 6) = -6 * EIZL / EL
     SEP(6, 6) = 4 * EIZL
     For I = 1 To 6
     For J = I To 6
       SEP(J, I) = SEP(I, J)
     Next J: Next I
'----- CONVERT ELEMENT STIFFNESS MATRIX TO GLOBAL SYSTEM
     DCOS(1, 1) = X21 / EL: DCOS(1, 2) = Y21 / EL: DCOS(1, 3) = 0
     DCOS(2, 1) = -DCOS(1, 2): DCOS(2, 2) = DCOS(1, 1): DCOS(2, 3) = 0
     DCOS(3, 1) = 0: DCOS(3, 2) = 0: DCOS(3, 3) = 1
     For I = 1 To 6
     For J = 1 To 6
       ALAMBDA(I, J) = 0!
     Next J: Next I
     For K = 1 To 2
       IK = 3 * (K - 1)
       For I = 1 To 3
       For J = 1 To 3
         ALAMBDA(I + IK, J + IK) = DCOS(I, J)
       Next J: Next I
     Next K
     If ISTF = 1 Then Exit Sub
     For I = 1 To 6
     For J = 1 To 6
       SE(I, J) = 0
       For K = 1 To 6
         SE(I, J) = SE(I, J) + SEP(I, K) * ALAMBDA(K, J)
       Next K
     Next J: Next I
     For I = 1 To 6: For J = 1 To 6: SEP(I, J) = SE(I, J): Next J: Next I
     For I = 1 To 6: For J = 1 To 6: SE(I, J) = 0
       For K = 1 To 6
          SE(I, J) = SE(I, J) + ALAMBDA(K, I) * SEP(K, J)
       Next K
     Next J: Next I
End Sub
```

```
'======== LOADS DUE TO UNIFORMLY DISTRIBUTED LOAD ========
Private Sub AddLoads()
'----- Loads due to uniformly distributed load on element
     For N = 1 To NE
     If Abs(UDL(N)) > 0 Then
       ISTF = 1
       Call Elstif(N)
       I1 = NOC(N, 1): I2 = NOC(N, 2)
       X21 = X(I2, 1) - X(I1, 1): Y21 = X(I2, 2) - X(I1, 2)
       EL = Sqr(X21 * X21 + Y21 * Y21)
```

continued

```
   ED(1) = 0: ED(4) = 0
   ED(2) = UDL(N) * EL / 2: ED(5) = ED(2)
   ED(3) = UDL(N) * EL ^ 2 / 12: ED(6) = -ED(3)
   For I = 1 To 6
    EDP(I) = 0
    For K = 1 To 6
     EDP(I) = EDP(I) + ALAMBDA(K, I) * ED(K)
    Next K
   Next I
   For I = 1 To 3
    F(3 * I1 - 3 + I) = F(3 * I1 - 3 + I) + EDP(I)
    F(3 * I2 - 3 + I) = F(3 * I2 - 3 + I) + EDP(I + 3)
   Next I
  End If
  Next N
End Sub
```

```
'=============== MEMBER END FORCES =============
Private Sub EndActions()
  ReDim EF(NE, 6)
  '----- Calculating Member End-Forces
  For N = 1 To NE
   ISTF = 1
   Call Elstif(N)
   I1 = NOC(N, 1): I2 = NOC(N, 2)
   For I = 1 To 3
    ED(I) = F(3 * I1 - 3 + I): ED(I + 3) = F(3 * I2 - 3 + I)
   Next I
   For I = 1 To 6: EDP(I) = 0
    For K = 1 To 6
     EDP(I) = EDP(I) + ALAMBDA(I, K) * ED(K)
    Next K: Next I
   '----- END FORCES DUE TO DISTRIBUTED LOADS
   If Abs(UDL(N)) > 0 Then
    ED(1) = 0: ED(4) = 0: ED(2) = -UDL(N) * EL / 2: ED(5) = ED(2)
    ED(3) = -UDL(N) * EL ^ 2 / 12: ED(6) = -ED(3)
   Else
    For K = 1 To 6: ED(K) = 0: Next K
   End If
   For I = 1 To 6 : EF(N, I) = ED(I)
    For K = 1 To 6
     EF(N, I) = EF(N, I) + SEP(I, K) * EDP(K)
    Next K: Next I
  Next N
End Sub
```

```
'*********       PROGRAM FRAME3D      ********
'*      3-D   FRAME ANALYSIS BY FEM          *
'*  T.R.Chandrupatla and A.D.Belegundu       *
'*******************************************
'============ MAIN PROGRAM ==============
Private Sub cmdStart_Click()
```

```
     Call InputData
     Call Bandwidth
     Call Stiffness
     Call AddLoads
     Call ModifyForBC
     Call BandSolver
     Call EndActions
     Call ReactionCalc
     Call Output
     cmdView.Enabled = True
     cmdStart.Enabled = False
End Sub
'==================================================
```

```
'============== ELEMENT STIFFNESS AND ASSEMBLY =============
Private Sub Stiffness()
     ReDim S(NQ, NBW)
     '----- Global Stiffness Matrix -----
     For N = 1 To NE
        picBox.Print "Forming Stiffness Matrix of Element "; N
        ISTF = 2
        Call Elstif(N)
        picBox.Print ".... Placing in Global Locations"
        Call PlaceGlobal(N)
     Next N
End Sub
'=========================================================
```

```
'============== ELEMENT STIFFNESS =============
Private Sub Elstif(N)
     '----- Element Stiffness Matrix -----
     I1 = NOC(N, 1): I2 = NOC(N, 2): I3 = NOC(N, 3): M = MAT(N)
     X21 = X(I2, 1) - X(I1, 1)
     Y21 = X(I2, 2) - X(I1, 2)
     Z21 = X(I2, 3) - X(I1, 3)
     EL = Sqr(X21 * X21 + Y21 * Y21 + Z21 * Z21)
     EAL = PM(M, 1) * ARIN(N, 1) / EL
     EIYL = PM(M, 1) * ARIN(N, 2) / EL: EIZL = PM(M, 1) * ARIN(N, 3) / EL
     GJL = PM(M, 2) * ARIN(N, 4) / EL
     For I = 1 To 12
     For J = 1 To 12
       SEP(I, J) = 0!
     Next J: Next I
     SEP(1, 1) = EAL: SEP(1, 7) = -EAL: SEP(7, 7) = EAL
     SEP(4, 4) = GJL: SEP(4, 10) = -GJL: SEP(10, 10) = GJL
     SEP(2, 2) = 12 * EIZL / EL ^ 2: SEP(2, 6) = 6 * EIZL / EL
     SEP(2, 8) = -SEP(2, 2): SEP(2, 12) = SEP(2, 6)
     SEP(3, 3) = 12 * EIYL / EL ^ 2: SEP(3, 5) = -6 * EIYL / EL
     SEP(3, 9) = -SEP(3, 3): SEP(3, 11) = SEP(3, 5)
     SEP(5, 5) = 4 * EIYL: SEP(5, 9) = 6 * EIYL / EL: SEP(5, 11) = 2 * EIYL
```

continued

```
     SEP(6, 6) = 4 * EIZL: SEP(6, 8) = -6 * EIZL / EL: SEP(6, 12) = 2 * EIZL
     SEP(8, 8) = 12 * EIZL / EL ^ 2: SEP(8, 12) = -6 * EIZL / EL
     SEP(9, 9) = 12 * EIYL / EL ^ 2: SEP(9, 11) = 6 * EIYL / EL
     SEP(11, 11) = 4 * EIYL: SEP(12, 12) = 4 * EIZL
     For I = 1 To 12
     For J = I To 12
       SEP(J, I) = SEP(I, J)
     Next J: Next I
'--- CONVERT ELEMENT STIFFNESS MATRIX TO GLOBAL SYSTEM
     DCOS(1, 1) = X21 / EL: DCOS(1, 2) = Y21 / EL: DCOS(1, 3) = Z21 / EL
     EIP1 = X(I3, 1) - X(I1, 1): EIP2 = X(I3, 2) - X(I1, 2)
     EIP3 = X(I3, 3) - X(I1, 3)
     C1 = DCOS(1, 2) * EIP3 - DCOS(1, 3) * EIP2
     C2 = DCOS(1, 3) * EIP1 - DCOS(1, 1) * EIP3
     C3 = DCOS(1, 1) * EIP2 - DCOS(1, 2) * EIP1
     CC = Sqr(C1 * C1 + C2 * C2 + C3 * C3)
     DCOS(3, 1) = C1 / CC: DCOS(3, 2) = C2 / CC: DCOS(3, 3) = C3 / CC
     DCOS(2, 1) = DCOS(3, 2) * DCOS(1, 3) - DCOS(1, 2) * DCOS(3, 3)
     DCOS(2, 2) = DCOS(1, 1) * DCOS(3, 3) - DCOS(3, 1) * DCOS(1, 3)
     DCOS(2, 3) = DCOS(3, 1) * DCOS(1, 2) - DCOS(1, 1) * DCOS(3, 2)
     For I = 1 To 12: For J = 1 To 12
       ALAMBDA(I, J) = 0!
     Next J: Next I
     For K = 1 To 4
       IK = 3 * (K - 1)
       For I = 1 To 3
       For J = 1 To 3
         ALAMBDA(I + IK, J + IK) = DCOS(I, J)
       Next J: Next I
     Next K
     If ISTF = 1 Then Exit Sub
     For I = 1 To 12
     For J = 1 To 12
       SE(I, J) = 0
       For K = 1 To 12
         SE(I, J) = SE(I, J) + SEP(I, K) * ALAMBDA(K, J)
       Next K
     Next J: Next I
     For I = 1 To 12: For J = 1 To 12: SEP(I, J) = SE(I, J): Next J: Next I
     For I = 1 To 12
     For J = 1 To 12
       SE(I, J) = 0
       For K = 1 To 12
         SE(I, J) = SE(I, J) + ALAMBDA(K, I) * SEP(K, J)
       Next K
     Next J: Next I
End Sub
```

```
'========= LOADS DUE TO UNIFORMLY DISTRIBUTED LOAD ========
Private Sub AddLoads()
'----- Loads due to uniformly distributed load on element
     For N = 1 To NE
     If Abs(UDL(N, 1)) > 0 Or Abs(UDL(N, 2)) > 0 Then
       ISTF * 1
       Call Elstif(N)
       I1 = NOC(N, 1): I2 = NOC(N, 2)
       X21 = X(I2, 1) - X(I1, 1)
       Y21 = X(I2, 2) - X(I1, 2)
       Z21 = X(I2, 3) - X(I1, 3)
       EL = Sqr(X21 * X21 + Y21 * Y21 + Z21 * 221)
       ED(1) = 0: ED(4) = 0: ED(7) = 0: ED(10) = 0
       ED(2) = UDL(N, 1) * EL / 2: ED(8) = ED(2)
       ED(6) = UDL(N, 1) * EL ^ 2 / 12: ED(12) = -ED(6)
       ED(3) = UDL(N, 2) * EL / 2: ED(9) = ED(3)
       ED(5) = -UDL(N, 2) * EL ^ 2 / 12: ED(11) = -ED(5)
       For I = 1 To 12
         EDP(I) = 0
         For K = 1 To 12
           EDP(I) = EDP(I) + ALAMBDA(K, I) * ED(K)
         Next K
       Next I
       For I = 1 To 6
         F(6 * I1 - 6 + I) = F(6 * I1 - 6 + I) + EDP(I)
         F(6 * I2 - 6 + I) = F(6 * I2 - 6 + I) + EDP(I + 6)
       Next I
     End If
     Next N
End Sub
```

```
'============== MEMBER END FORCES ============
Private Sub EndActions()
     ReDim EF(NE, 12)
     '----- Calculating Member End-Forces
     For N = 1 To NE
       ISTF = 1
       Call Elstif(N)
       I1 = NOC(N, 1): I2 = NOC(N, 2)
       For I = 1 To 6
         ED(I) = F(6 * I1 - 6 + I): ED(I + 6) = F(6 * I2 - 6 + I)
       Next I
       For I = 1 To 12
         EDP(I) = 0
         For K = 1 To 12
           EDP(I) = EDP(I) + ALAMBDA(I, K) * ED(K)
         Next K
       Next I
```

continued

```
'--- END FORCES DUE TO DISTRIBUTED LOADS
       If Abs(UDL(N, 1)) > 0 Or Abs(UDL(N, 2)) > 0 Then
         ED(1) = 0: ED(4) = 0: ED(7) = 0: ED(10) = 0
         ED(2) = -UDL(N, 1) * EL / 2: ED(8) = ED(2)
         ED(6) = -UDL(N, 1) * EL ^ 2 / 12: ED(12) = -ED(6)
         ED(3) = -UDL(N, 2) * EL / 2: ED(9) = ED(3)
         ED(5) = UDL(N, 2) * EL ^ 2 / 12: ED(11) = -ED(5)
       Else
         For K = 1 To 12: ED(K) = 0: Next K
       End If
       For I = 1 To 12
         EF(N, I) = ED(I)
         For K = 1 To 12
           EF(N, I) = EF(N, I) + SEP(I, K) * EDP(K)
         Next K
       Next I
     Next N
End Sub
```