Computer Implementation 2.4 ***(Matlab)*** *Solution of axial deformation problems (p. 163)*

A *Matlab* implementation for analysis of axial deformation problems is presented in this example. It can betreated as a template for solution of other similar problems. First we define three simple functions to compute the axial deformation element stiffness matrix, load vector, and element results as follows.

MatlabFiles\Chap2\AxialDefElement.m

```
function k = AxialDefElement(e, A, coord)
% k = AxialDefElement(e, A, coord)
% Generates stiffness matrix of an axial deformation element
% e = modulus of elasticity
% A = Area of cross-section
% coord = coordinates at the element ends

x1=coord(1); x2=coord(2);
k = e*A/(x2-x1)*[1,-1; -1,1];
```

MatlabFiles\Chap2\AxialDefLoad.m

```
function rq = AxialDefLoad(q, coord)
% rq = AxialDefLoad(q, coord)
% Generates equivalent load vector for an axial deformation element
% q = uniformly distributed load
% coord = coordinates at the element ends

x1=coord(1); x2=coord(2);
rq = q*(x2-x1)/2*[1;1];
```

MatlabFiles\Chap2\AxialDefResults.m

```
function results = AxialDefResults(e, A, coord, dn)
% results = AxialDefResults(e, A, coord, dn)
% e = modulus of elasticity
% A = Area of cross-section
% coord = coordinates at the element ends
% dn = displacements at element ends
% The output variables are axial strain, axial stress,
% and axial force.

x1=coord(1); x2=coord(2); L=x2-x1;
eps= [-1,1]/L*dn;
sigma = e*eps;
force = sigma*A;
results=[eps, sigma, force];
```

Using these functions, and following procedures discussed in Chapter 1, the global equations for the four element model can be assembled as follows.

MatlabFiles\Chap1\AxialDefBarEx.m

```
% Tapered bar example
e = 70*10^3; P = 20*1000;
nodes = [0:150:600];
A = [2175, 1725, 1275, 825];
lmm = [1,2; 2,3; 3,4; 4,5];
K=zeros(5);
% Generate stiffness matrix for each element and assemble it.
for i=1:4
   lm=lmm(i,:);
   k=AxialDefElement(e, A(i), nodes(lm));
   K(lm, lm) = K(lm, lm) + k;
end
K
% Define the load vector
R = zeros(5,1); R(3)=P

% Nodal solution and reactions
[d, reactions] = NodalSoln(K, R, [1,5], zeros(2,1))
results=[];
for i=1:4
   results = [results; AxialDefResults(e, A(i), ...
         nodes(lmm(i,:)), d(lmm(i,:)))];
end
format short g
results
plot(nodes,d),title('Axial displacement'), xlabel('x'),ylabel('u')
```

```
>> AxialDefBarEx

K =

    1015000   -1015000          0          0          0
   -1015000    1820000    -805000          0          0
          0    -805000    1400000    -595000          0
          0          0    -595000     980000    -385000
          0          0          0    -385000     385000


R =

          0
```

```
       0
   20000
       0
       0


d =

         0
  0.012958
  0.029296
  0.017787
         0


reactions =

    -13152
   -6847.9


results =

  8.6385e-005      6.0469      13152
   0.00010892      7.6244      13152
 -7.6727e-005     -5.3709    -6847.9
  -0.00011858     -8.3005    -6847.9
```

A plot of the nodal values is as follows.