ME 324 - Analyzing spring assembly using finite element method in MATLAB

WIJEKOON W.M.M.K.

E/20/442

SEMESTER 06

MATLAB code :

```
%   E number
E = 442;
fprintf('E number = %d\n', E);
fprintf('\n');

%   calculate R1,R2,R3
R1 = rem(E,2);
R2 = rem(E,3);
R3 = rem(E,4);

fprintf('R1 = %d\n',R1);
fprintf('R2 = %d\n',R2);
fprintf('R3 = %d\n',R3);
fprintf('\n');

%   Kalculate loacal element stifness matrixes
k1 = (4+R1)*100;
k2 = (3+R2)*100;
k3 = k2;
k4 = (2+R3)*100;
k5 = (4+R2)*100;
k6 = (3+R3)*100;

fprintf('k1 = %d\n', k1);
fprintf('k2 = %d\n', k2);
fprintf('k3 = %d\n', k3);
fprintf('k4 = %d\n', k4);
fprintf('k5 = %d\n', k5);
fprintf('k6 = %d\n', k6);
fprintf('\n');
```

```

s1 = SpringElementStiffness(k1);
disp('Stifness metric of element _1 =');
disp(s1);

s2 = SpringElementStiffness(k2);
disp('Stifness metric of element _2 =');
disp(s2);

s3 = SpringElementStiffness(k3);
disp('Stifness metric of element _3 =');
disp(s3);

s4 = SpringElementStiffness(k4);
disp('Stifness metric of element _4 =');
disp(s4);

s5 = SpringElementStiffness(k5);
disp('Stifness metric of element _5 =');
disp(s5);

s6 = SpringElementStiffness(k6);
disp('Stifness metric of element _6 =');
disp(s6);

%    find global stiffness matrix
K = zeros(5,5);

```

```
K=SpringAssemble(K,s1,1,2);
K=SpringAssemble(K,s2,2,3);
K=SpringAssemble(K,s3,2,3);
K=SpringAssemble(K,s4,2,4);
K=SpringAssemble(K,s5,3,4);
K=SpringAssemble(K,s6,4,5);

disp("Global matrix  K = ")
disp(K);

%   boundary cnditions
dx1 = 0;
dx2 = 0;

disp('boundary condistions');
fprintf('dx1 = %d\n',dx1);
fprintf('dx2 = %d\n',dx2);
fprintf('\n');

%   forces
Fx2 = 1000;
Fx3 = 0;
Fx4 = -2000;

disp('Applied forces = ');
fprintf('Fx2 = %d\n',Fx2);
fprintf('Fx3 = %d\n',Fx3);
fprintf('Fx4 = %d\n',Fx4);
fprintf('\n');
```

```
Force  = ['Fx1';'Fx2';'Fx3';'Fx4';'Fx5'];
Fs = [Fx2 ; Fx3 ; Fx4];
disp('Force matrix F = ');
disp(Force);
fprintf('\n');

%   sub stiffness and sub forces matrixes for solve
disp('sub force matrix  F_s =');
disp(Fs);
fprintf('\n');

Ks = K(2:4,2:4);
disp(' sub matrix for calculation K_s = ');
disp(Ks);
fprintf('\n');

Dx234 = Ks\Fs;
fprintf('Dx2 = %d\n',Dx234(1));
fprintf('Dx3 = %d\n',Dx234(2));
fprintf('Dx4 = %d\n',Dx234(3));
fprintf('\n');

%   displacement matrix
Dx = [0; Dx234; 0];
disp('displacement matrix Dx =');
disp(Dx);
fprintf('\n');
```

```
%Forces on node 1 and 5
F = K*Dx;
disp('Force matric Fx =');
disp(F);
fprintf('\n');

Fx1 = [F(1)]*1000;
Fx5 = [F(5)]*1000;

fprintf('Fx1 = %d\n',Fx1);
fprintf('Fx5 = %d\n',Fx5);
fprintf('\n');

%forces acting on each element
dx1 = [0; Dx(2)];
f1=SpringElementForces(s1,dx1);
fprintf('force on element 1= %d\n',f1);
fprintf('\n');

dx2 = [Dx(2); Dx(3)];
f2=SpringElementForces(s2,dx2);
fprintf('force on element 2= %d\n',f2);
fprintf('\n');

dx3 = [Dx(2); Dx(3)];
f3=SpringElementForces(s3,dx3);
fprintf('force on element 3= %d\n',f3);
fprintf('\n');
```

```
dx4 = [Dx(2); Dx(4)];
f4=SpringElementForces(s4,dx4);
fprintf('force on element 4= %d\n',f4);
fprintf('\n');

dx5 = [Dx(3); Dx(4)];
f5=SpringElementForces(s5,dx5);
fprintf('force on element 5= %d\n',f5);
fprintf('\n');

dx6 = [Dx(4); Dx(5)];
f6=SpringElementForces(s6,dx6);
fprintf('force on element 6= %d\n',f6);
fprintf('\n');
```

RESULTS

Displacements;

- Node 2 _ Dx2 = -2.481618e-01
- Node 3 _ Dx3 = -8.455882e-01
- Node 4 _ Dx4 = -1.801471e+00

Displacement Matrix

0

-0.2482

-0.8456

-1.8015

0

Force Matrix

0099.3

1000.0

0

-2000.0

900.7

Force acting on node 1 _ Fx1 = 99.3 N

Force acting on node 1 _ Fx1 = 900.7 N

Table 01 : Force acting on each element

Element number	Force (kN)	Type
1	9.926471e+01	Compressive
2	2.389706e+02	Compressive
3	2.389706e+02	Compressive
4	6.213235e+02	Compressive
5	4.779412e+02	Compressive
6	-9.007353e+02	Tensile