

Lab Assignment 3, 03/28/2019, 1800 -- 2000

**Due 2000**

**Lab Grading Policy: Attendance 20%, Basic 80%, Bonus 20%**

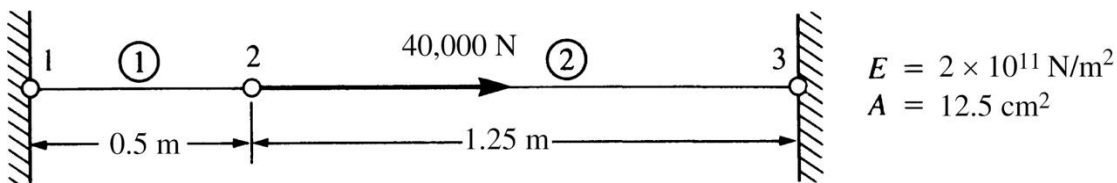
You are expected to complete the basic part during the Lab. In case you have difficulty in finishing the basic part on time, you should upload them before 2100 on Saturday and a penalty of 20% discount will be applied on your score. You are encouraged to complete the bonus part (no penalty applied). Basic and/or bonus parts should be submitted by **2100 on Saturday and no late submission is permitted**. We will in general post the reference solutions by **Sunday**.

---

Download `BarSimple.zip` from the course website and unzip it. You will find a folder containing a `problem1dSimple.m` file with two functions, `solution.m` and `outputDisplacementsReactions.m`.

1. (40%) Modify the MATLAB codes and determine the nodal displacements, the force in each element, and the reactions. Be aware of unit consistency.

(a)



Sample output:

```
Displacements
node    displacements
1:      0.0000e+00
2:      5.7143e-05
3:      0.0000e+00
Reactions
node    reactions
1:     -2.8571e+04
3:     -1.1429e+04
Element Forces
element node_I node_J   force I   force J
1       1       2     -2.8571e+04  2.8571e+04
2       2       3      1.1429e+04 -1.1429e+04
% probla 20190328
clear
clc
% SETTING
```

```

E = [2e11,2e11];           % E: modulus of elasticity (N/m^2)
A = 12.5e-4;              % A: area of cross section (m^2)
L = [0.5,1.25];          % L: length of bar (m)

numberElements = 2;      % numberElements: number of elements
numberNodes = 3;        % numberNodes: number of nodes
elementNodes = [1 2;2 3]; % node ID
nodeCoordinates = [0,0.5,1.75]; % node coordinate

force = zeros(numberNodes,1); % initialized
stiffness = zeros(numberNodes,numberNodes); % initialized
k = cell(numberElements,1);

% MAIN CODING
% F matrix
force(2)=40000;          % NBC : applied load at node 2
force_fef = zeros(numberNodes,1);
% K matrix
for e=1:numberElements
    elementDof = elementNodes(e,:); % element degrees of freedom (Dof)
    k{e} = E(e)*A/L(e)*[1 -1;-1 1]; % element stiffness for 1D
    stiffness(elementDof,elementDof) = stiffness(elementDof,elementDof) +
k{e}; % assemble
end
% Essential BCs
prescribedDof=[1;3]; % EBC node ID
% Solution
displacements=solution(numberNodes,prescribedDof,stiffness,force);

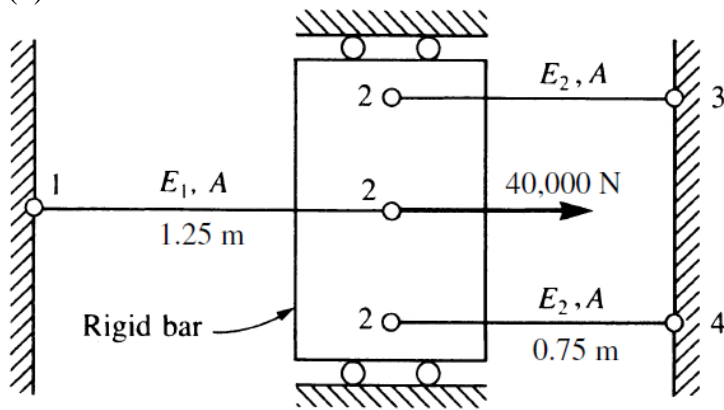
% OUTPUT

outputDisplacementsReactions(displacements,stiffness,numberNodes,prescribedDof,force)
outputElementForces(numberElements,elementNodes,displacements,k)

function outputElementForces(numberElements,elementNodes,displacements,k)
% output of Element Forces in tabular form
%-----
disp('Element Forces')
fprintf('element  node_I  node_J  force I      force J\n')
for e=1:numberElements
    elementDof = elementNodes(e,:); % element degrees of freedom (Dof)
    elementforces = k{e}*displacements(elementDof);
    fprintf('  %d      %d      %d      %11.4e      %11.4e\n',...
e,elementDof,elementforces)
end
end

```

(b)



$$E_1 = 2 \times 10^{11} \text{ N/m}^2$$
$$E_2 = 6.67 \times 10^{10} \text{ N/m}^2$$
$$A = 12.5 \text{ cm}^2$$

Sample output:

```
Displacements
node      displacements
1:        0.0000e+00
2:        9.4712e-05
3:        0.0000e+00
4:        0.0000e+00
Reactions
node      reactions
1:       -1.8942e+04
3:       -1.0529e+04
4:       -1.0529e+04
Element Forces
element  node_I  node_J  force I  force J
1        1      2      -1.8942e+04  1.8942e+04
2        2      3       1.0529e+04 -1.0529e+04
3        2      4       1.0529e+04 -1.0529e+04
```

```
% prob1b 20190328
clear
clc
% SETTING
E = [2e11,6.67e10,6.67e10]; % E: modulus of elasticity (N/m^2)
A = 12.5e-4; % A: area of cross section (m^2)
L = [1.25,0.75,0.75]; % L: length of bar (m)

numberElements = 3; % numberElements: number of elements
numberNodes = 4; % numberNodes: number of nodes
elementNodes = [1 2;2 3;2 4]; % node ID
nodeCoordinates = [0,1.25,2,2]; % node coordinate

force = zeros(numberNodes,1); % initialized
stiffness = zeros(numberNodes,numberNodes); % initialized
k = cell(numberElements,1);

% MAIN CODING
% F matrix
force(2)=40000; % NBC : applied load at node 2
force_fef = zeros(numberNodes,1);
% K matrix
for e=1:numberElements
```

```

        elementDof = elementNodes(e,:); % element degrees of freedom (Dof)
        k{e} = E(e)*A/L(e)*[1 -1;-1 1]; % element stiffness for 1D
        stiffness(elementDof,elementDof) = stiffness(elementDof,elementDof) +
k{e}; % assemble
    end
% Essential BCs
    prescribedDof=[1;3;4]; % EBC node ID
% Solution
    displacements=solution(numberNodes,prescribedDof,stiffness,force);

% OUTPUT

outputDisplacementsReactions(displacements,stiffness,numberNodes,prescribedDof,force)
    outputElementForces(numberElements,elementNodes,displacements,k)

```

2. (40%) Let us reconsider Problem 1(a). In addition to the setup, we would also like to impose a displacement at node 3 so  $u_3 = 1.0E - 6 \text{ m}$ . Modify the MATLAB codes and determine the nodal displacements, the force in each element, and the reactions. Below is a sample output:

```

Displacements
node    displacements
1:      0.0000e+00
2:      5.7429e-05
3:      1.0000e-06
Reactions
node    reactions
1:      -2.8714e+04
3:      -1.1286e+04
Element Forces
element node_I node_J    force I    force J
1       1       2    -2.8714e+04  2.8714e+04
2       2       3     1.1286e+04 -1.1286e+04

```

```

% prob4
clear
clc

% SETTING
E = [2e11,2e11]; % E: modulus of elasticity (N/m^2)
A = 12.5e-4; % A: area of cross section (m^2)
L = [0.5,1.25]; % L: length of bar (m)

numberElements = 2; % numberElements: number of elements
numberNodes = 3; % numberNodes: number of nodes
elementNodes = [1 2;2 3]; % node ID
nodeCoordinates = [0,0.5,1.75]; % node coordinate

force = zeros(numberNodes,1); % initialized
stiffness = zeros(numberNodes,numberNodes); % initialized
k = cell(numberElements,1);
displacements = zeros(numberNodes,1); % initialized

% MAIN CODING
% F matrix
    force(2) = 40000; % NBC : applied load at node 2
% K matrix

```

```

for e=1:numberElements
    elementDof = elementNodes(e, :); % element degrees of freedom (Dof)
    k{e} = E(e)*A/L(e)*[1 -1;-1 1]; % element stiffness for 1D
    stiffness(elementDof,elementDof) = stiffness(elementDof,elementDof) +
k{e}; % assemble
end
% Essential BCs
prescribedDof=[1;3]; % EBC node ID
displacements(3) = 1e-6; % EBC of u3 = 10^-6 m
% Solution

displacements=solutionMOD(displacements,numberNodes,prescribedDof,stiffness,force);

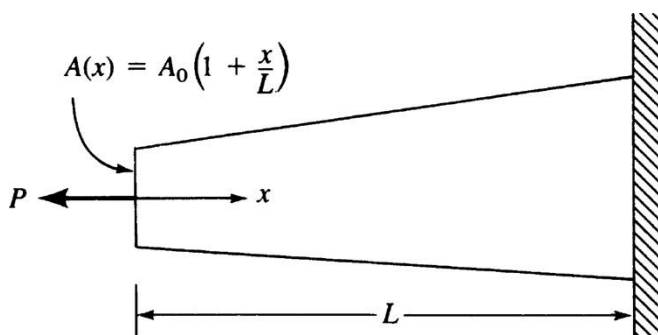
% OUTPUT

outputDisplacementsReactions(displacements,stiffness,numberNodes,prescribedDof,force)
outputElementForces(numberElements,elementNodes,displacements,k)

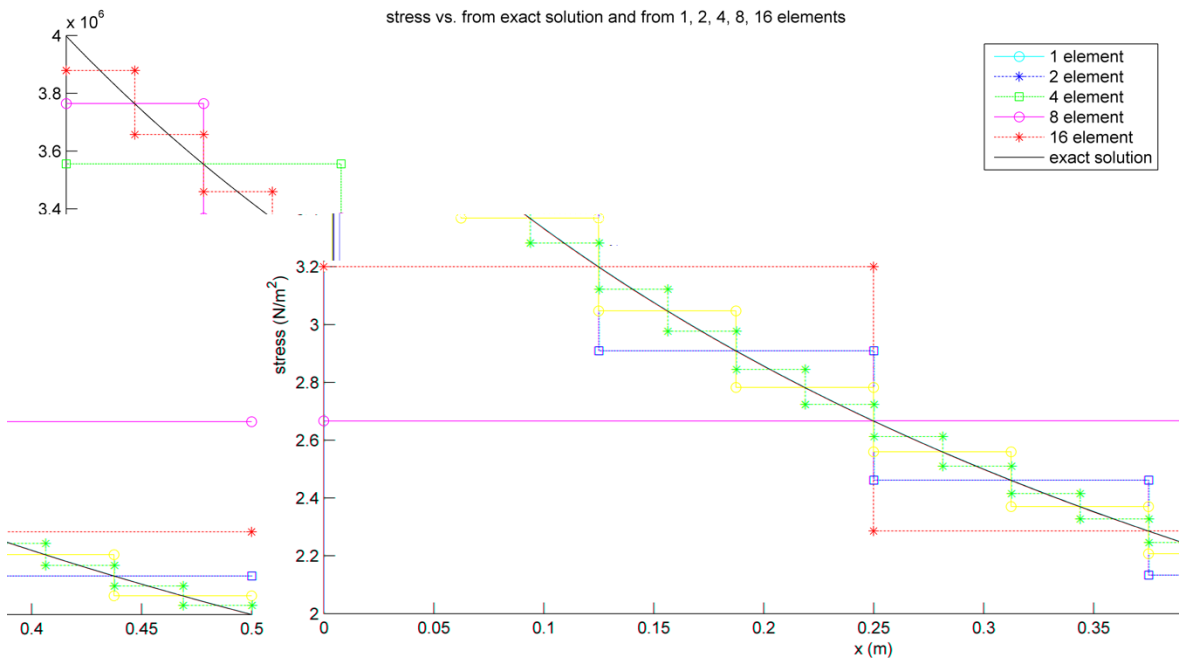
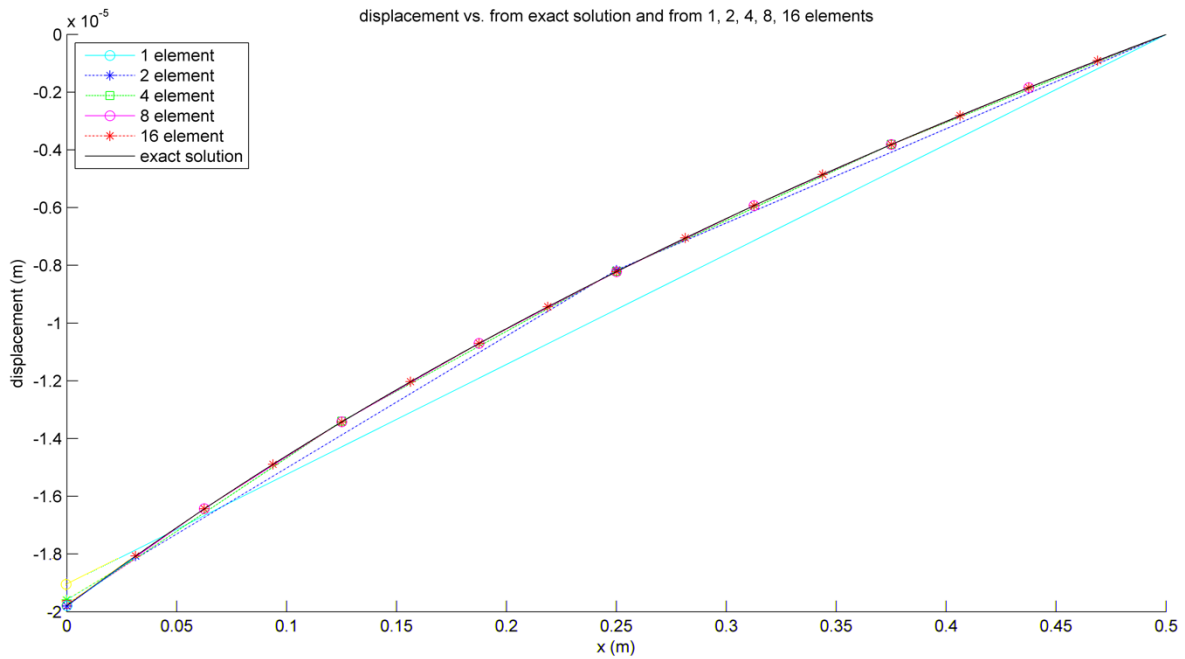
function
displacements=solutionMOD(displacements,GDof,prescribedDof,stiffness,force)
% function to find solution in terms of global displacements
%-----
activeDof = setdiff((1:GDof)',prescribedDof);
% force consider settlement at EBC
F = force(activeDof)-...
stiffness(activeDof,prescribedDof)*displacements(prescribedDof);
U = stiffness(activeDof,activeDof)\F;
displacements(activeDof) = U;

```

3. **(Bonus 20%)** Modify the MATLAB codes and solve for the axial displacement and stress in the tapered bar shown below using 1, 2, 4, 8, 16 constant-area elements. Evaluate the area at the center of each element length. Use that area for each element. Let  $A_0 = 12.5 \text{ cm}^2$ ,  $L = 0.5 \text{ m}$ ,  $E = 70 \text{ GPa}$  and  $P = 5000 \text{ N}$ .



Compare your finite element results (displacements and stresses) with the exact solution. **Notice the discontinuity nature of stress between elements.** Sample MATLAB plots look like:



```
% Bonus 20190328
close all
clear
clc
jj=0;
for i = [1,2,4,8,16]
    fprintf('==== %d elements ==== \n', i)
    % SETTING
    % Element number
    numberElements = i;
    numberNodes = numberElements+1;
    elementNodes = [(1:numberNodes-1)', (2:numberNodes)'];
    % Nodes Coordinate
    L = 0.5;
    nodeCoordinates = (0:L/numberElements:L)';
    % Element length
```

```

    Le = L/numberElements*ones(numberElements,1);
% Young Modulus
    E = ones(numberElements,1)*70e+9;
% Section Area
    A0 = 12.5e-4;
    A = A0*(1+(nodeCoordinates(1:end-1)+Le/2)./L);

force = zeros(numberNodes,1);           % initialized
stiffness = zeros(numberNodes,numberNodes); % initialized
k = cell(numberElements,1);
stress = zeros(numberElements*2,1);

% MAIN CODING
% F matrix
force(1) = -5000;                       % NBC : applied load at node 2
% K matrix
for e=1:numberElements
    elementDof = elementNodes(e,:);      % element degrees of freedom (Dof)
    k{e} = E(e)*A(e)/Le(e)*[1 -1;-1 1];  % element stiffness for 1D
    stiffness(elementDof,elementDof) = stiffness(elementDof,elementDof) +
k{e};  % assemble
end
% Essential BCs
prescribedDof=elementNodes(end,end);    % EBC node ID
% Solution
displacements=solution(numberNodes,prescribedDof,stiffness,force);

% OUTPUT

outputDisplacementsReactions(displacements,stiffness,numberNodes,prescribedDof,force)
outputElementForces(numberElements,elementNodes,displacements,k)

% PLOT-1
% stress
plot_x = zeros(numberElements*2,1);
for e=1:numberElements
    plot_x(e*2-1:e*2) = nodeCoordinates(e:e+1);
    elementDof = elementNodes(e,:);
    B = 1/Le(e)*[1 -1;-1 1];             % d/dx for 1D
    True_Stress = E(e)*B*displacements(elementDof);
    plot_stress(e*2-1:e*2) = [True_Stress(2);True_Stress(2)];
end

% plot
jj = jj+1;
COLOR = {'c-o','b:*','g:s','m-o','r:*'};
figure(1)
    p(jj) = plot(nodeCoordinates,displacements,COLOR{jj});hold on
figure(2)
    pp(jj) = plot(plot_x,plot_stress,COLOR{jj});hold on

end
% Exact Solution
syms u(x) x
E = 70e+9;
A = A0*(1+x/L);
ODE = diff(E*A*diff(u,x,1),x,1);
Du = diff(u,x,1);
u_exact(x) = dsolve(ODE==0,u(L)==0,Du(0)==5000/E/A0);
stress_exact(x) = E*diff(u_exact,x,1);
% PLOT-2

```

```

plotx_exact = 0:0.01:L;
figure(1)
p(6) = plot(plotx_exact,u_exact(plotx_exact),'k-');
legend([p(1) p(2) p(3) p(4) p(5) p(6)],...
       '1 element','2 element','4 element','8 element','16 element','exact
Solution','location','northwest')
title('displacement vs. from exat solution and from 1, 2, 4, 8, 16
elements','fontsize',8);xlabel('x (m)');ylabel('displacement (m)')
figure(2)
pp(6) = plot(plotx_exact,stress_exact(plotx_exact),'k-');
legend([pp(1) pp(2) pp(3) pp(4) pp(5) pp(6)],...
       '1 element','2 element','4 element','8 element','16 element','exact
Solution','location','northeast')
title('stress vs. from exat solution and from 1,2,4,8,16
elements','fontsize',8);xlabel('x (m)');ylabel('stress (N/m^2)')

```