Istanbul Technical University
Graduate School of Science Engineering and Technology

MKC525E
Finite Element Analysis in Engineering

Homework 3

Erdem Çalışkan
503191531
01/07/2020

1 Question 1

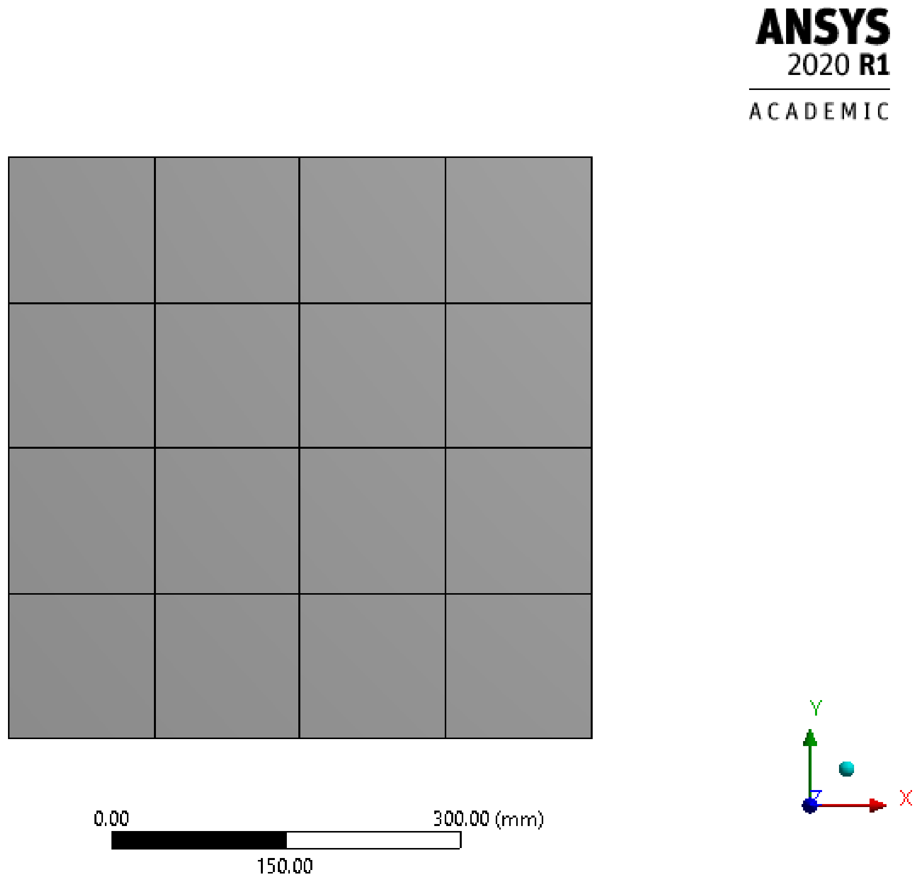

Figure 1: A plate (500x500 mm) having thickness of 2 mm is divided into 4 quadrilateral elements.

Topology matrix:

$$\begin{array}{cccccccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 13 & 14 & 15 & 10 & 11 & 12 \\
4 & 5 & 6 & 7 & 8 & 9 & 16 & 17 & 18 & 13 & 14 & 15 \\
10 & 11 & 12 & 13 & 14 & 15 & 22 & 23 & 24 & 19 & 20 & 21 \\
13 & 14 & 15 & 16 & 17 & 18 & 25 & 26 & 27 & 22 & 23 & 24
\end{array} \tag{1}$$

1.1 All edges simply supported

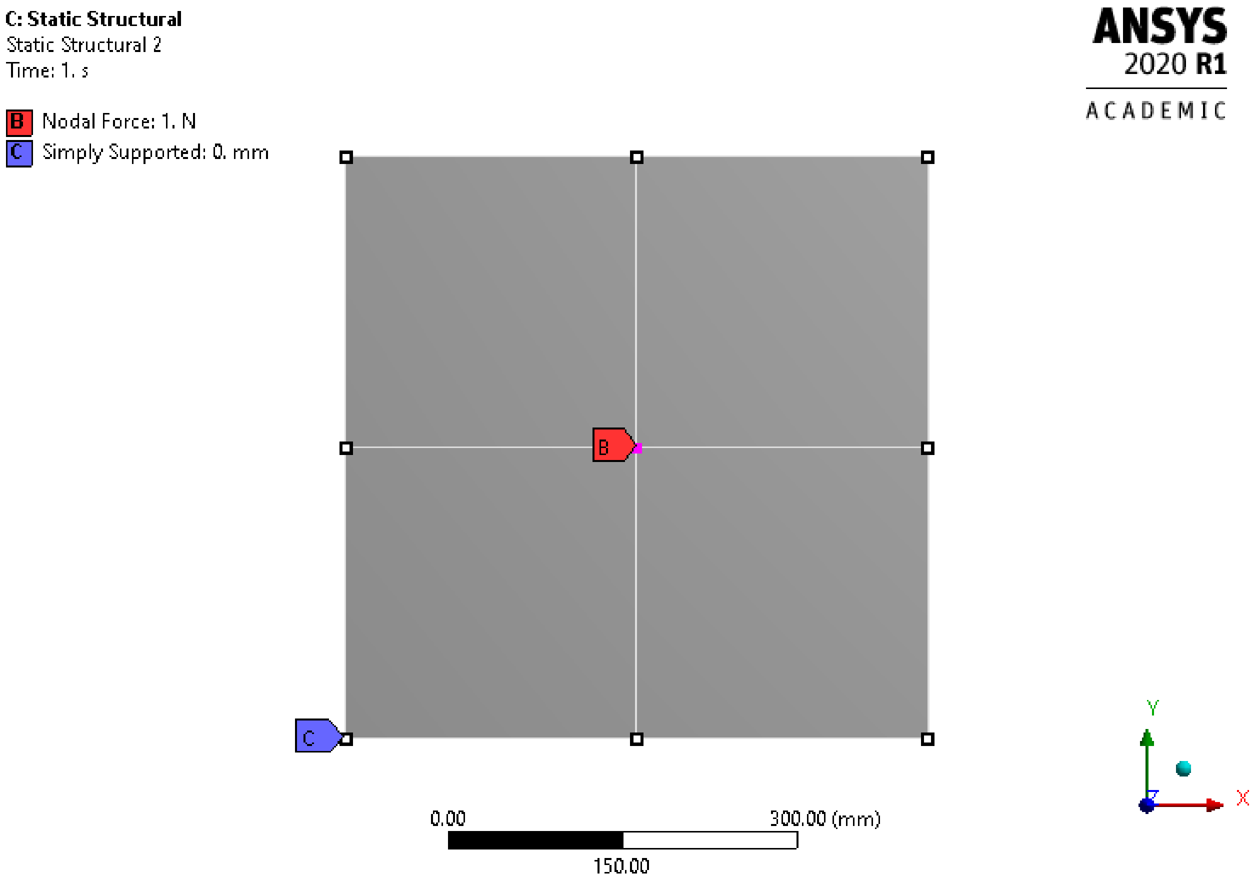

Figure 2: Boundary conditions for all edges simply supported.

Nodal DOF	Node 7	Node 8	Node 9
ω_1	0	0	0
θ_x	8.08E-05	-4.41E-05	8.08E-05
θ_y	-4.61E-04	-6.10E-20	4.61E-04
Nodal DOF	Node 4	Node 5	Node 6
ω_1	0	0.01002415	0
θ_x	1.28E-05	1.75E-05	1.28E-05
θ_y	4.34E-04	6.12E-20	-4.34E-04
Nodal DOF	Node 1	Node 2	Node 3
ω_1	0	0	0
θ_x	-9.68E-05	-3.22E-05	-9.68E-05
θ_y	-4.64E-04	-8.13E-20	4.64E-04

Table 1: Deflections for all edges simply supported.

1.2 Right edge clamped

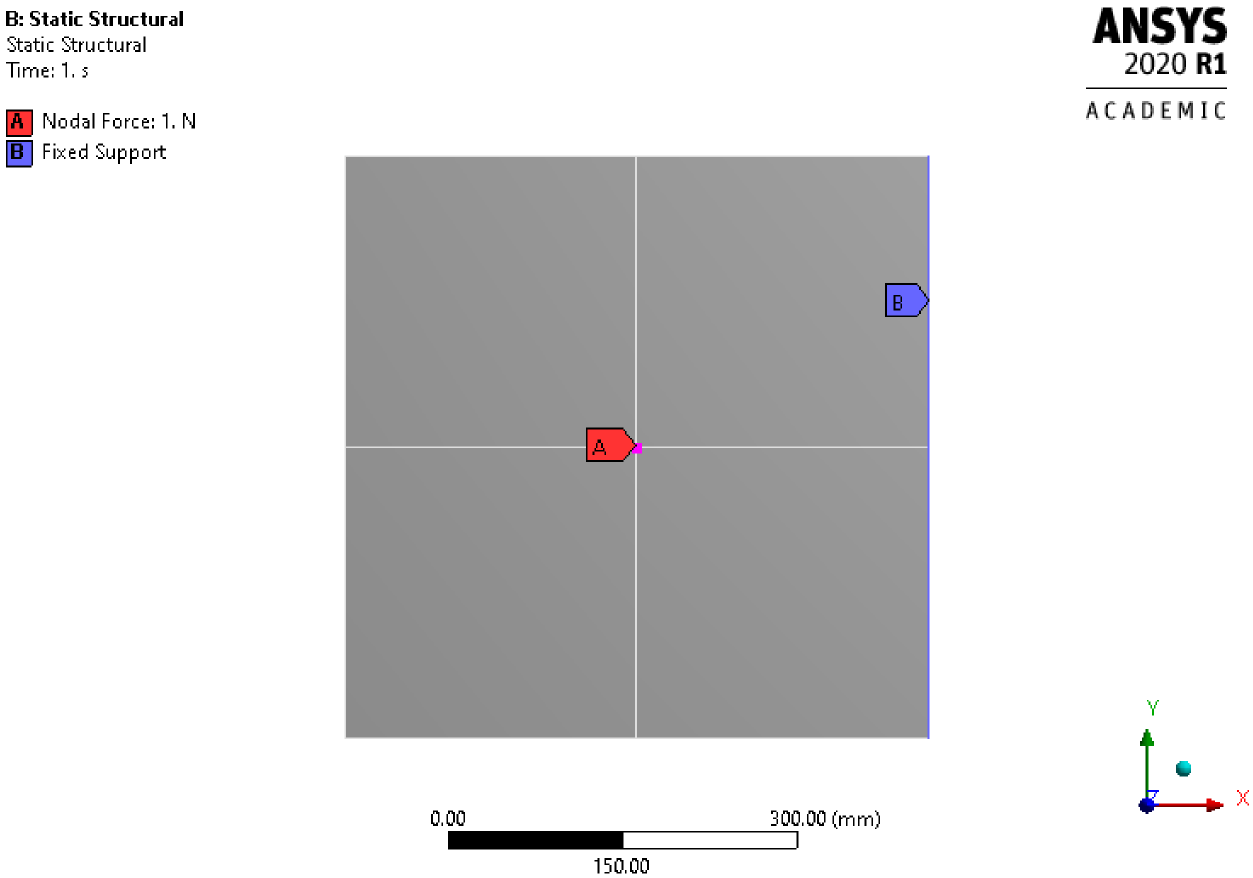

Figure 3: Boundary conditions for all edges clamped.

Nodal DOF	Node 7	Node 8	Node 9
ω_1	0.070474754	0.052606438	0
θ_x	0.000124691	0.000149244	0
θ_y	0.000316421	-0.000258474	0
Nodal DOF	Node 4	Node 5	Node 6
ω_1	0.025463514	0.067778679	0
θ_x	-0.000112087	-7.85E-05	0
θ_y	-0.000403946	0.000404517	0
Nodal DOF	Node 1	Node 2	Node 3
ω_1	-0.098508864	-0.051977319	0
θ_x	0.000709631	0.000431472	0
θ_y	0.00030366	-0.000494191	0

Table 2: Deflections for right edge clamped.

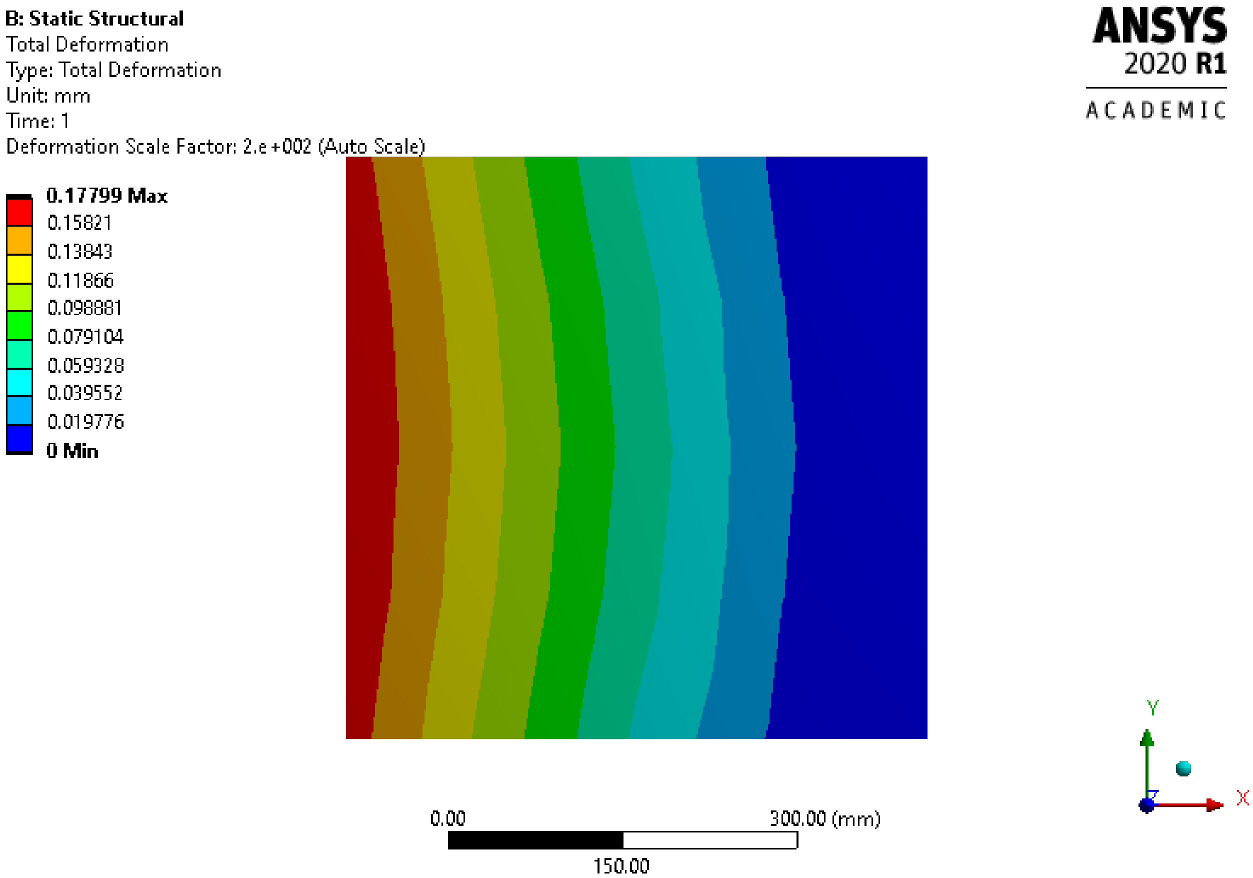

Figure 4: Deflections for right edge clamped.

2 Strains and stresses

2.1 All edges simply supported

Nodal DOF	Node 7	Node 8	Node 9
ϵ_x	6.72E-10	1.17E-09	-2.09E-08
ϵ_y	-1.66E-09	-2.02E-08	-1.58E-09
xy	-1.85E-08	2.21E-08	-2.17E-08
Nodal DOF	Node 4	Node 5	Node 6
ϵ_x	1.92E-08	-2.09E-08	1.17E-09
ϵ_y	2.02E-09	2.01E-08	1.93E-09
xy	1.81E-08	-2.17E-08	2.21E-08
Nodal DOF	Node 1	Node 2	Node 3
ϵ_x	7.99E-10	1.92E-08	7.99E-10
ϵ_y	2.01E-08	-1.66E-09	-1.58E-09
xy	-2.17E-08	-1.85E-08	-2.17E-08

Table 3: Strains for all edges simply supported.

Nodal DOF	Node 7	Node 8	Node 9
σ_{xx}	4.41E-05	-1.07E-03	-4.90E-03
σ_{yy}	-3.35E-04	-4.55E-03	-1.75E-03
τ_{xy}	-1.51E-03	1.80E-03	-1.77E-03
Nodal DOF	Node 4	Node 5	Node 6
σ_{xx}	4.53E-03	-3.46E-03	3.97E-04
σ_{yy}	1.74E-03	3.23E-03	5.20E-04
τ_{xy}	1.48E-03	-1.77E-03	1.80E-03
Nodal DOF	Node 1	Node 2	Node 3
σ_{xx}	1.52E-03	4.29E-03	7.83E-05
σ_{yy}	4.67E-03	8.96E-04	-3.09E-04
τ_{xy}	-1.77E-03	-1.51E-03	-1.77E-03

Table 4: Stresses for all edges simply supported.

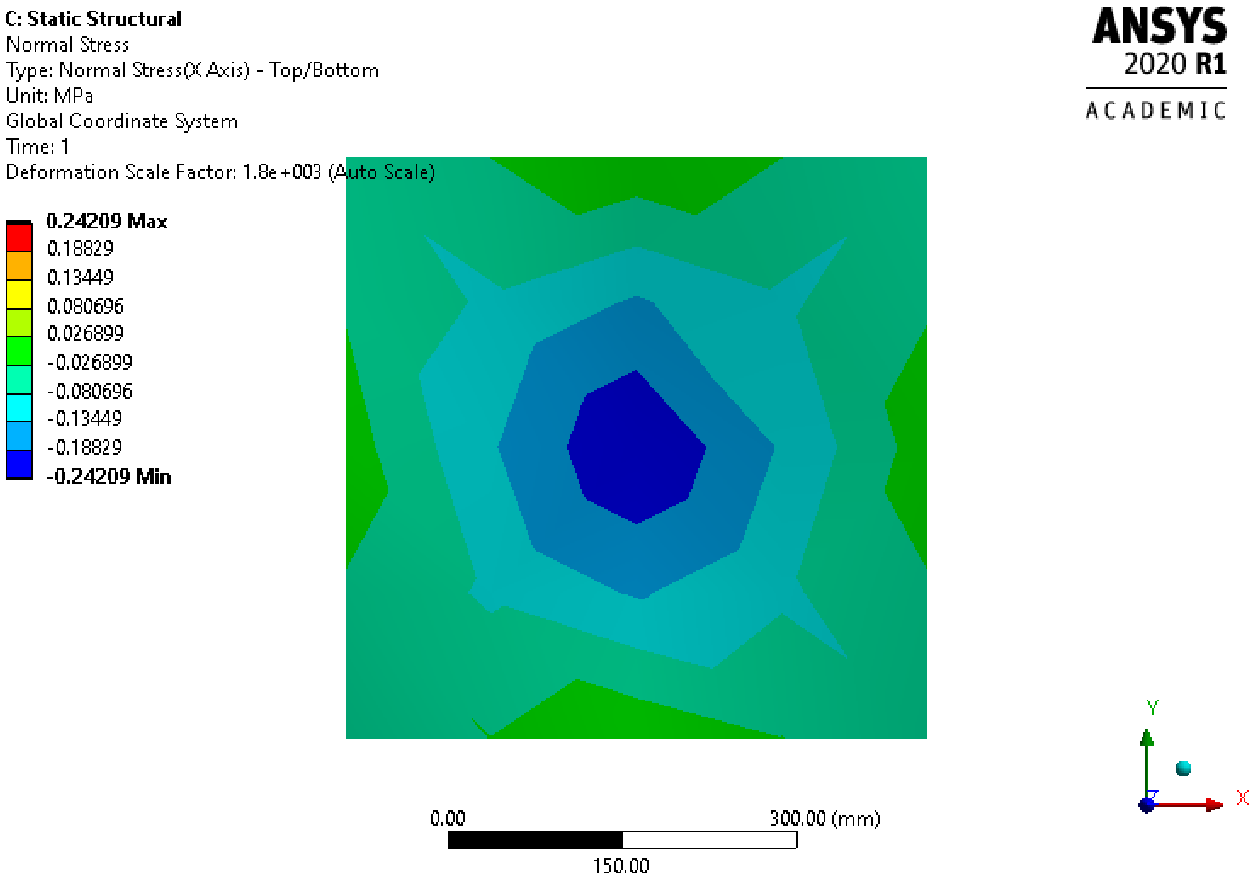

Figure 5: Normal stress (x-axis)

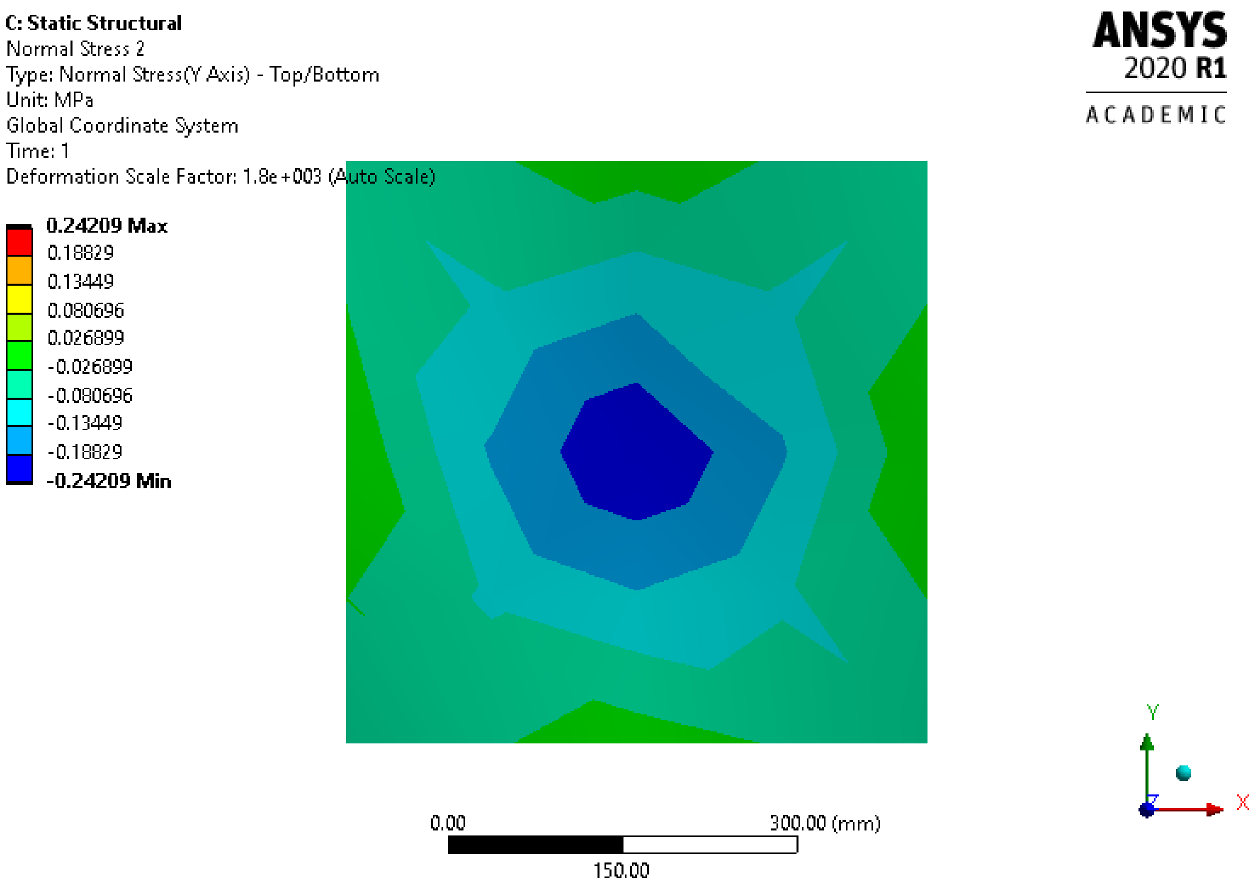

Figure 6: Normal stress (y-axis)

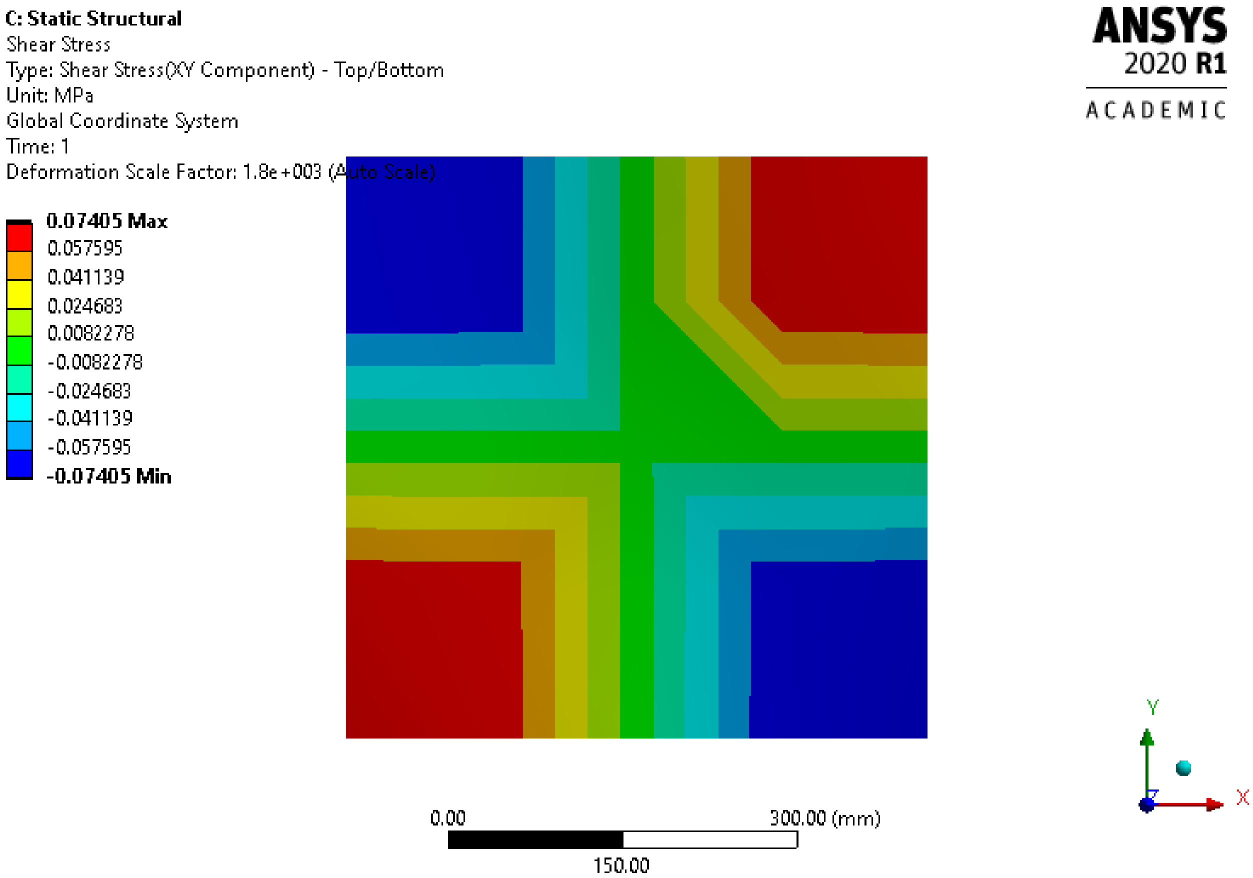

Figure 7: Shear stress

2.2 Right edge clamped

Nodal DOF	Node 7	Node 8	Node 9
ϵ_x	-3.68E-08	-1.05E-07	-1.36E-07
ϵ_y	9.19E-08	-3.12E-08	0
$_{xy}$	-1.23E-07	3.12E-08	-2.40E-07
Nodal DOF	Node 4	Node 5	Node 6
ϵ_x	8.63E-08	-1.36E-07	-1.05E-07
ϵ_y	2.45E-07	2.40E-07	0
$_{xy}$	-4.60E-09	-2.40E-07	3.12E-08
Nodal DOF	Node 1	Node 2	Node 3
ϵ_x	1.04E-07	8.63E-08	1.04E-07
ϵ_y	2.40E-07	9.19E-08	0
$_{xy}$	-2.40E-07	-1.23E-07	-2.40E-07

Table 5: Strains for right edge clamped.

Nodal DOF	Node 7	Node 8	Node 9
σ_{xx}	-2.33E-03	-2.61E-02	-3.12E-02
σ_{yy}	1.86E-02	-1.41E-02	-9.06E-03
τ_{xy}	-1.00E-02	2.54E-03	-1.96E-02
Nodal DOF	Node 4	Node 5	Node 6
σ_{xx}	3.61E-02	-1.53E-02	-2.41E-02
σ_{yy}	6.19E-02	4.60E-02	-6.98E-03
τ_{xy}	-3.74E-04	-1.96E-02	2.54E-03
Nodal DOF	Node 1	Node 2	Node 3
σ_{xx}	3.98E-02	2.59E-02	2.39E-02
σ_{yy}	6.20E-02	2.68E-02	6.92E-03
τ_{xy}	-1.96E-02	-1.00E-02	-1.96E-02

Table 6: Stresses for right edge clamped.

Gauss integration points: Derivatives are calcuted at this points, so they are the most accurate points for these properties.

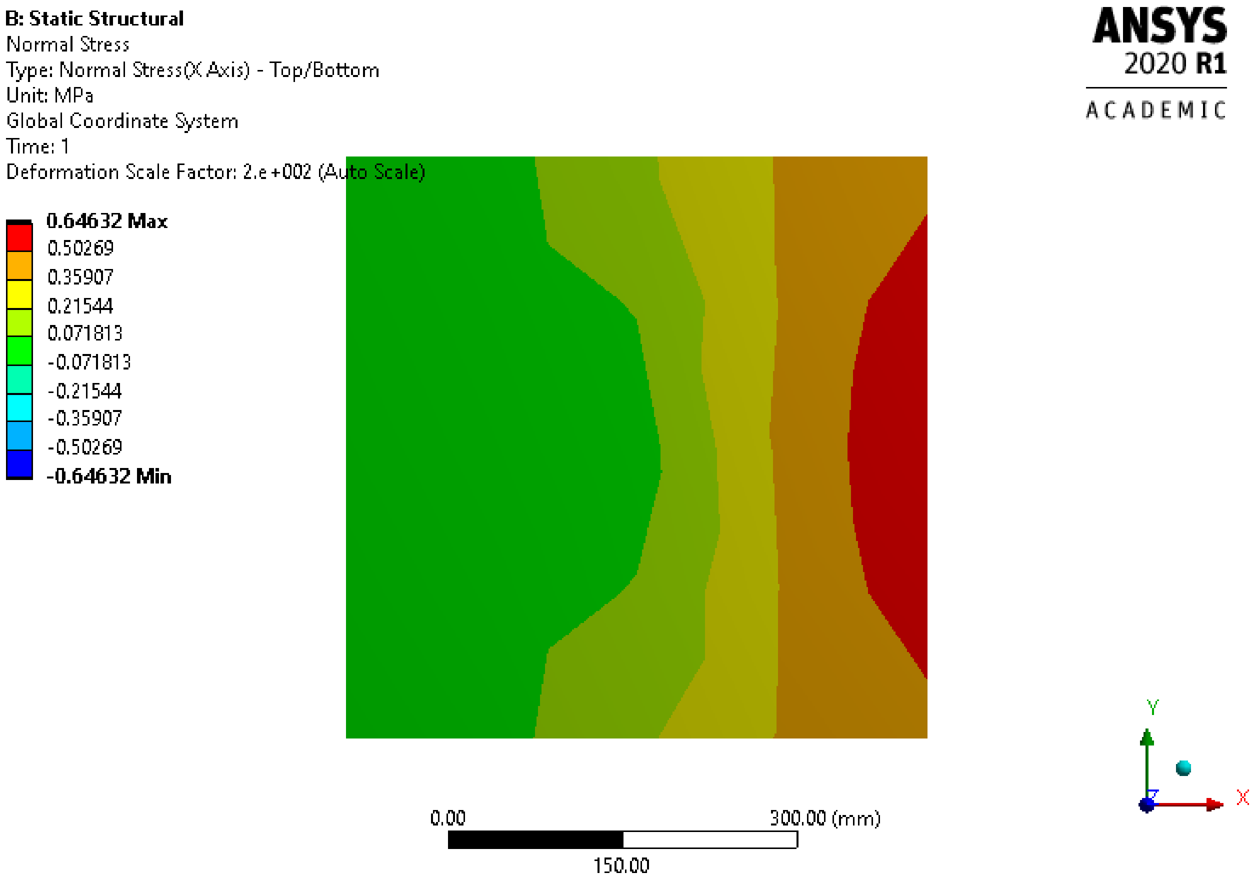

Figure 8: Normal stress (x-axis)

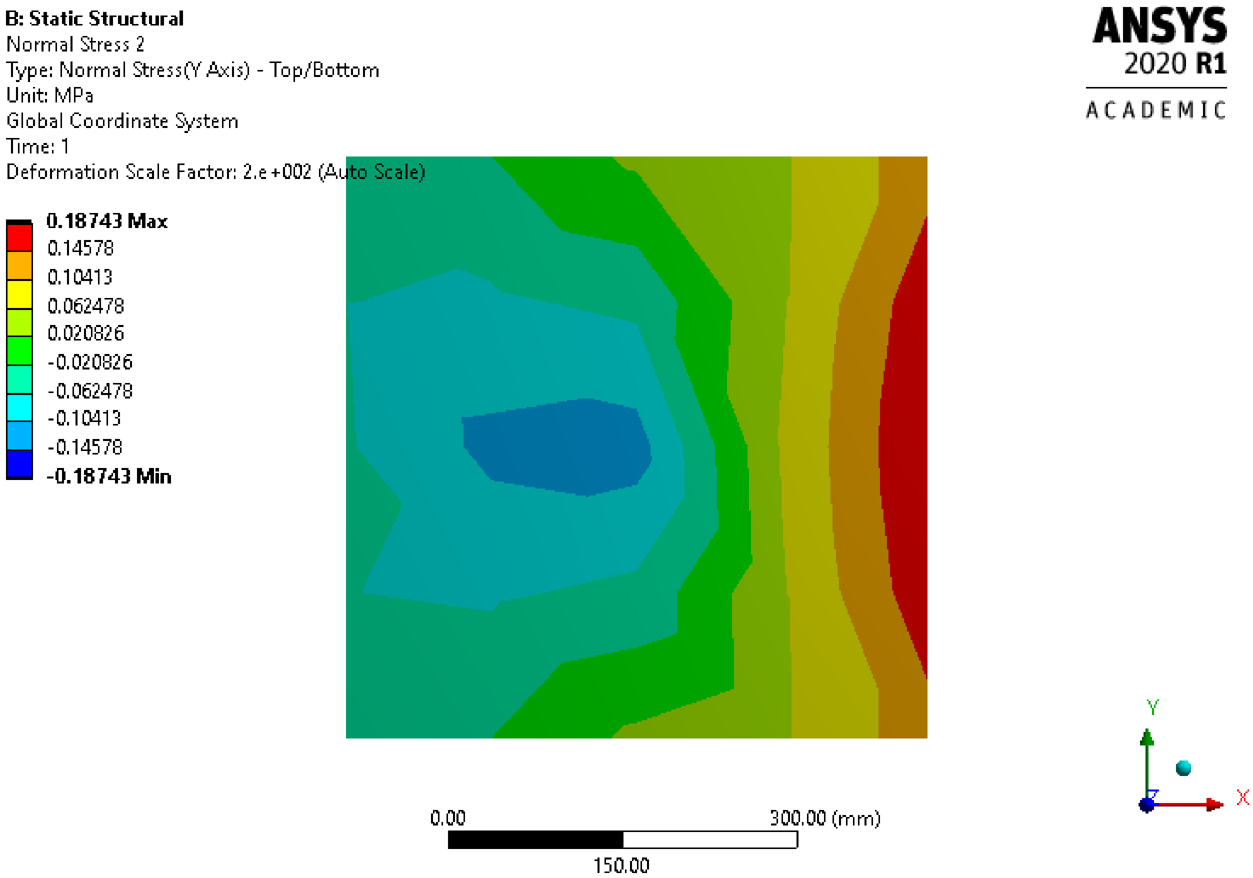

Figure 9: Normal stress (y-axis)

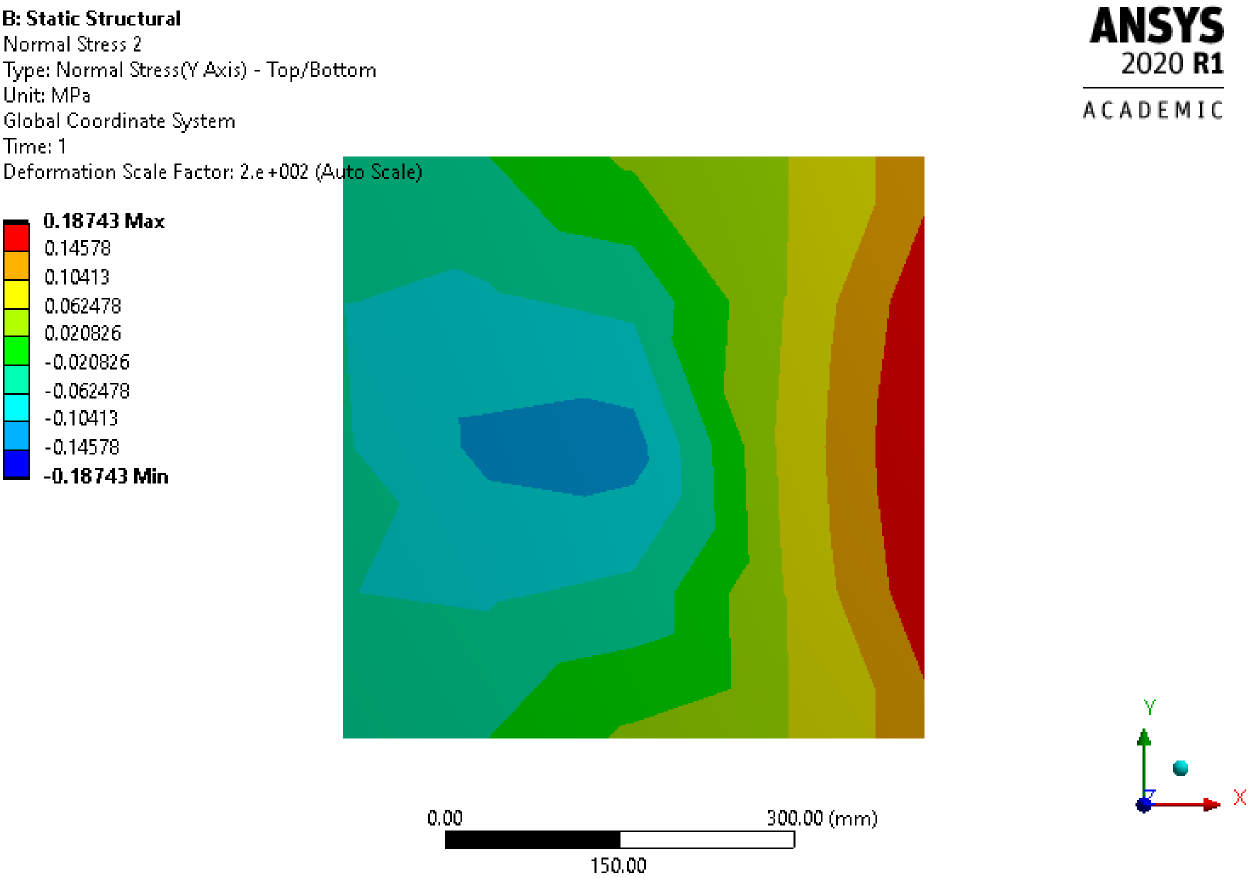

Figure 10: Shear stress

Matlab Scripts

Listing 1: Matlab code for question 1

```
% HW3

clc
clear all
close all
format compact
format short e
tic

% input data for control parameters
nel=4;                    % number of elements
nnel=4;                   % number of nodes per element
ndof=3;                   % number of dots per node
nnode=9;                  % total number of nodes in system
sdof=nnode*ndof;          % total system dof
edof=nnel*ndof;           % degrees of freedom per element
emodule=2.10e5;            % elastic modulus
v=0.29;             % Poisson's ratio
t=2;                      % plate thickness

E = emodule;

a=500;
b=500;
ro=a/b;
rom=b/a;
% input data for nodal coordinate values
% gcoord(ij) where i-> node no. and j-> x or y

top = [ 1  2  3  4  5  6 13 14 15 10 11 12
        4  5  6  7  8  9 16 17 18 13 14 15
       10 11 12 13 14 15 22 23 24 19 20 21
       13 14 15 16 17 18 25 26 27 22 23 24];

F = (42-12*v+60*ro^2+60*rom^2)*t^2/a/b;
G = (30*ro+3*rom+12*v*rom)*t^2/b;
H = (30*rom+3*ro+12*v*ro)*t^2/a;
I = (-42+12*v-60*ro^2+30*rom^2)*t^2/a/b;
J = (30*ro+3*(1-v*rom)*t^2)/b;
K = (15*rom-3*ro-12*v*ro)*t^2/a;
L = (-42+12*v-60*rom^2+30*rom^2)*t^2/a/b;
M = (-15*ro+3*rom+12*v*rom)*t^2/b;
N = (30*rom+3*(1-v)*ro)*t^2/a;
O = (42-12*v-30*ro^2-30*rom^2)*t^2/a/b;
P = (-15*ro+3*(1-v)*rom)*t^2/a;
Q = (15*rom-3*(1-v)*ro)*t^2/a;
R = (20*ro+4*(1-v)*ro)*t^2;
S = (10*ro-(1-v)*rom)*t^2;
T = (10*ro-4*(1-v)*rom)*t^2;
U = (5*ro+(1-v)*rom)*t^2;
V = (20*ro+4*(1-v)*ro)*t^2;
W = (20*rom+4*(1-v)*ro)*t^2;
X = (10*rom-4*(1-v)*ro)*t^2;
Y = (10*rom-(1-v)*ro)*t^2;
Z = 15*v*t^2;
```

```
Ke = emodule*t/(180*(1-v^2))*[ F   G -H   L -M -N   O -P -Q   I   J -K;
                               G   R -Z -M -T   0   P   U   0 -J   S   0;
                              -H -Z   V   N   0   X   Q   0   Y -K   0   W;
                               L -M   N   F   G   H   I   J   K   O -P   Q;
                              -M -T   0   G   R   Z -J   S   0   P   U   0;
                              -N   0   X   H   Z   V   K   0   W -Q   0   Y;
                               O   P   Q   I -J   K   F -G   H   L   M   N;
                              -P   U   0   J   S   0 -G   R -Z   M   T   0;
                              -Q   0   Y   K   0   W   H -Z   V -N   0   X;
                               I -J -K   O   P -Q   L   M -N   F -G -H;
                               J   S   0 -P   U   0   M   T   0 -G   R   Z;
                              -K   0   W   Q   0   Y   N   0   X -H   Z   V];

Ke=double(Ke);

Kglob=zeros(nnode*ndof);
for nel=1:4
    for i=1:12
        for j=1:12
            Kglob(top(nel,i),top(nel,j))= ...
            Kglob(top(nel,i),top(nel,j))+Ke(i,j);
        end
    end
end

% force vector

Fglob=zeros(nnode*ndof,1);
Fglob(13) = 1;


% %Boundary conditions (All edges are simply supported)
% BCs=[1;4;7;10;16;19; 22; 25];

% Boundary conditions (Right edge is clamped)
BCs=[7; 8; 9; 16; 17; 18; 25; 26; 27];

activeDof=setdiff(1:nnode*ndof',BCs);
Kglobactive=Kglob(activeDof,activeDof);
Fglobactive=Fglob(activeDof,1);

%Solve
d = Kglobactive\Fglobactive
do(activeDof,1)= d

A=zeros(12,12,4);
nodexcoors = [0 a 2*a];
nodeycoors = [0 b 2*b];

nc=          [ 0   0
               0   a
               0 2*a
               b   0
               b   a
               b 2*a
             2*b   0
             2*b   a
```

```
                  2*b   2*a];

en =       [1  2  4  5
            2  3  5  6
            4  5  8  7
            5  6  9  8];

for el = 1:4
A(:,:,el) = ...
[1 nc(en(el,1),1) nc(en(el,1),2) nc(en(el,1),1)^2 nc(en(el,1),1)*nc(en(el,1),2) nc(en(el,1),2)^2 nc(en(el,1),1)^3 nc(en(el,1),1)^2*nc(en(el,1),2) nc(en(el,1),1)*nc(en(el,1),2)^2 nc(en(el,1),2)^3 nc(en(el,1),1)^3*nc(en(el,1),2) nc(en(el,1),1)*nc(en(el,1),2)^3;
0 0 1 0 nc(en(el,1),1) 2*nc(en(el,1),2) 0 nc(en(el,1),1)^2 2*nc(en(el,1),1)*nc(en(el,1),2) 3*nc(en(el,1),2) nc(en(el,1),1)^3 3*nc(en(el,1),1)*nc(en(el,1),2)^2;
0 -1 0 -2*nc(en(el,1),1) -nc(en(el,1),2) 0 -3*nc(en(el,1),1)^2 -2*nc(en(el,1),1)*nc(en(el,1),2) -nc(en(el,1),2)^2 0 -3*nc(en(el,1),1)^2*nc(en(el,1),2) -nc(en(el,1),2)^3;
1 nc(en(el,2),1) nc(en(el,2),2) nc(en(el,2),1)^2 nc(en(el,2),1)*nc(en(el,2),2) nc(en(el,2),2)^2 nc(en(el,2),1)^3 nc(en(el,2),1)^2*nc(en(el,2),2) nc(en(el,2),1)*nc(en(el,2),2)^2 nc(en(el,2),2)^3 nc(en(el,2),1)^3*nc(en(el,2),2) nc(en(el,2),1)*nc(en(el,2),2)^3;
0 0 1 0 nc(en(el,2),1) 2*nc(en(el,2),2) 0 nc(en(el,2),1)^2 2*nc(en(el,2),1)*nc(en(el,2),2) 3*nc(en(el,2),2) nc(en(el,2),1)^3 3*nc(en(el,2),1)*nc(en(el,2),2)^2;
0 -1 0 -2*nc(en(el,2),1) -nc(en(el,2),2) 0 -3*nc(en(el,2),1)^2 -2*nc(en(el,2),1)*nc(en(el,2),2) -nc(en(el,2),2)^2 0 -3*nc(en(el,2),1)^2*nc(en(el,2),2) -nc(en(el,2),2)^3;
1 nc(en(el,3),1) nc(en(el,3),2) nc(en(el,3),1)^2 nc(en(el,3),1)*nc(en(el,3),2) nc(en(el,3),2)^2 nc(en(el,3),1)^3 nc(en(el,3),1)^2*nc(en(el,3),2) nc(en(el,3),1)*nc(en(el,3),2)^2 nc(en(el,3),2)^3 nc(en(el,3),1)^3*nc(en(el,3),2) nc(en(el,3),1)*nc(en(el,3),2)^3;
0 0 1 0 nc(en(el,3),1) 2*nc(en(el,3),2) 0 nc(en(el,3),1)^2 2*nc(en(el,3),1)*nc(en(el,3),2) 3*nc(en(el,3),2) nc(en(el,3),1)^3 3*nc(en(el,3),1)*nc(en(el,3),2)^2;
0 -1 0 -2*nc(en(el,3),1) -nc(en(el,3),2) 0 -3*nc(en(el,3),1)^2 -2*nc(en(el,3),1)*nc(en(el,3),2) -nc(en(el,3),2)^2 0 -3*nc(en(el,3),1)^2*nc(en(el,3),2) -nc(en(el,3),2)^3;
1 nc(en(el,4),1) nc(en(el,4),2) nc(en(el,4),1)^2 nc(en(el,4),1)*nc(en(el,4),2) nc(en(el,4),2)^2 nc(en(el,4),1)^3 nc(en(el,4),1)^2*nc(en(el,4),2) nc(en(el,4),1)*nc(en(el,4),2)^2 nc(en(el,4),2)^3 nc(en(el,4),1)^3*nc(en(el,4),2) nc(en(el,4),1)*nc(en(el,4),2)^3;
0 0 1 0 nc(en(el,4),1) 2*nc(en(el,4),2) 0 nc(en(el,4),1)^2 2*nc(en(el,4),1)*nc(en(el,4),2) 3*nc(en(el,4),2) nc(en(el,4),1)^3 3*nc(en(el,4),1)*nc(en(el,4),2)^2;
0 -1 0 -2*nc(en(el,4),1) -nc(en(el,4),2) 0 -3*nc(en(el,4),1)^2 -2*nc(en(el,4),1)*nc(en(el,4),2) -nc(en(el,4),2)^2 0 -3*nc(en(el,4),1)^2*nc(en(el,4),2) -nc(en(el,4),2)^3;                    ];

end

do = [do
      0
      0
      0];

alfa=zeros(12,1,4);
```

```
el=0;
for el=1:4
    alfa(:,:,el)=A(:,:,el)\do(top(el,:))
end

ex=zeros(9,4);
ey=zeros(9,4);
gxy=zeros(9,4);

el=0;
z=1;

for el=1:4
    for i = 1:9
        ex(i,el)=2*z*(alfa(4,1,el)+3*alfa(7,1,el)*nc(i,1)+alfa(8,1,el)*nc(i,2)
            +3*alfa(11,1,el)*nc(i,1)*nc(i,2));
        ey(i,el)=2*z*(alfa(6,1,el)+3*alfa(10,1,el)*nc(i,2)+alfa(9,1,el)*nc(i,1)
            +3*alfa(12,1,el)*nc(i,1)*nc(i,2));
        gxy(i,el)=2*z*(alfa(5,1,el)+2*alfa(8,1,el)*nc(i,1)+2*alfa(9,1,el)*nc(i
            ,2)+3*alfa(11,1,el)*nc(i,1)^2+3*alfa(12,1,el)*nc(i,2));
    end
end

maxdo = max(do) %Maximum deflection

%Reaction forces and moments at nodes
Freaction=Kglob*do-Fglob;


gpoints = [ -0.577350269189626            -0.577350269189626;
             0.577350269189626            -0.577350269189626;
             0.577350269189626             0.577350269189626;
            -0.577350269189626             0.577350269189626];

gweights = [1 1;
            1 1;
            1 1;
            1 1];


E1 = E/(1-v^2);

ff = zeros(12,1);

N = zeros(1,12);
n = sym('n',[12 3]);
ns = sym('ns',[12 3]);
nt = sym('nt',[12 3]);
ns2 = sym('ns2',[12 3]);
nt2 = sym('nt2',[12 3]);
nsnt = sym('nsnt',[12 3]);

syms s t
for i=1:length (gweights)
    for j=1:length(gweights)
        for node=1:4
            if node==1
            sc=-1;
```

```
            tc=-1;
            elseif node==2
            sc=1;
            tc=-1;
            elseif node==3
            sc=1;
            tc=1;
            else
            sc=-1;
            tc=1;
            end
            n(node,:)=[(1/4)*(1+sc*s)*(1+tc*t),...
                       (1/4)*(1+sc*s)*(1+tc*t),...
                       (1/4)*(1+sc*s)*(1+tc*t)];
        end

        ns=diff(n,s);
        nt=diff(n,t);
        ns2=diff(n,s,2);
        nt2=diff(n,t,2);
        nsnt=diff(ns,t);

        N=[n(1,:) n(2,:) n(3,:) n(4,:)];
        Ns2=[ns(1,:) ns(2,:) ns(3,:) ns(4,:)];
        Nt2=[nt(1,:) nt(2,:) nt(3,:) nt(4,:)];
        Nsnt=[nsnt(1,:) nsnt(2,:) nsnt(3,:) nsnt(4,:)];

        B=[Ns2/a^2; Nt2/b^2; 2/(a*b)*Nsnt];

        D=[ E1 E1*v 0;...
            E1*v E1 0;...
            0 0 E/(2*(1+v))];

        %Element consistent load vector
        ff=a*b*N'*1*gweights(i)*gweights(j);
        ff=subs(ff,s,gpoints(i));
        ff=subs(ff,t,gpoints(j));
    end
end

%Strain at nodes (eps x, eps y, gama xy)
B1=subs(B,s,-1);
B1=subs(B1,t,1);
B1=double(B1);

strain=zeros(3,1,9);

e=0;

for i=[4 5 7 8] %Nodes [1:4 6:9 11:14 16:19]
    e=e+1;
    strain(:,:,i)=B1*do(top(e,:));
end

B2=subs(B,s,1);
B2=subs(B2,t,1);
B2=double(B2);

```

```
e=0;

for i=[9 6] %Nodes [5 10 15 20]
    e=e+2;
    strain(:,:,i)=B2*do(top(e,:));
end

B3=subs(B,s,-1);
B3=subs(B3,t,-1);
B3=double(B3);

e=1;
for i=[1 2] %Nodes [21 24]
    e=e+1;
    strain(:,:,i)=B3*do(top(e,:));
end

B4=subs(B,s,1);
B4=subs(B4,t,-1);
B4=double(B4);
strain(:,:,3)=B4*do(top(2,:)); %Node 25

%Stress at nodes (sigma x, sigma y,tau xy)
stress=zeros(3,1,9);
for i=1:9
    stress(:,:,i)=D*strain(:,:,i)
end


toc
```