

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

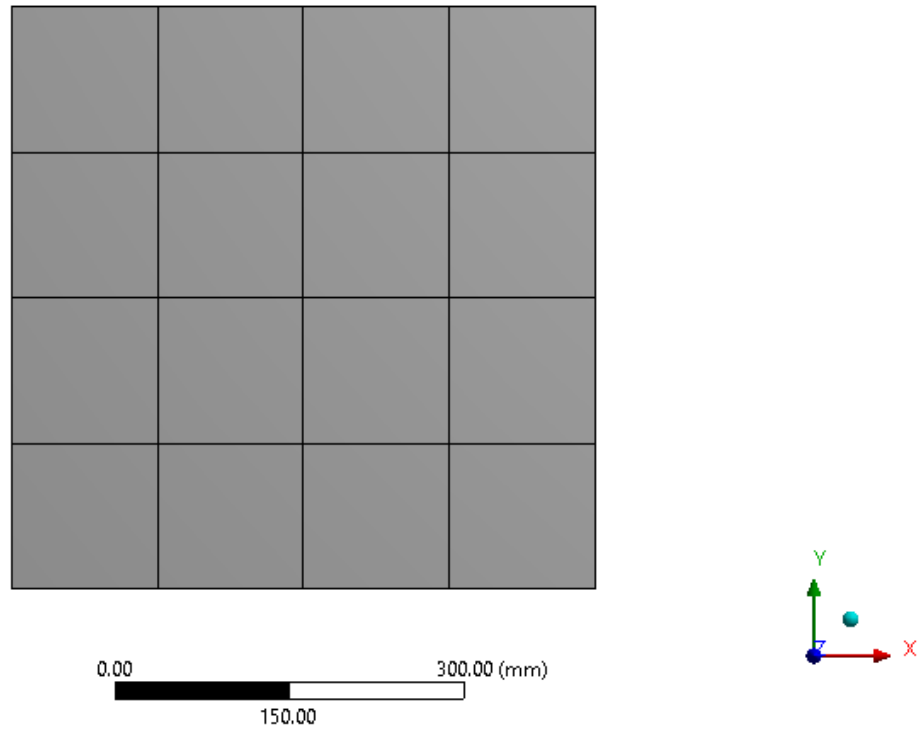


Figure 1: A plate (500x500 mm) having thickness of 2 mm is divided into 4 quadrilateral elements.

Topology matrix:

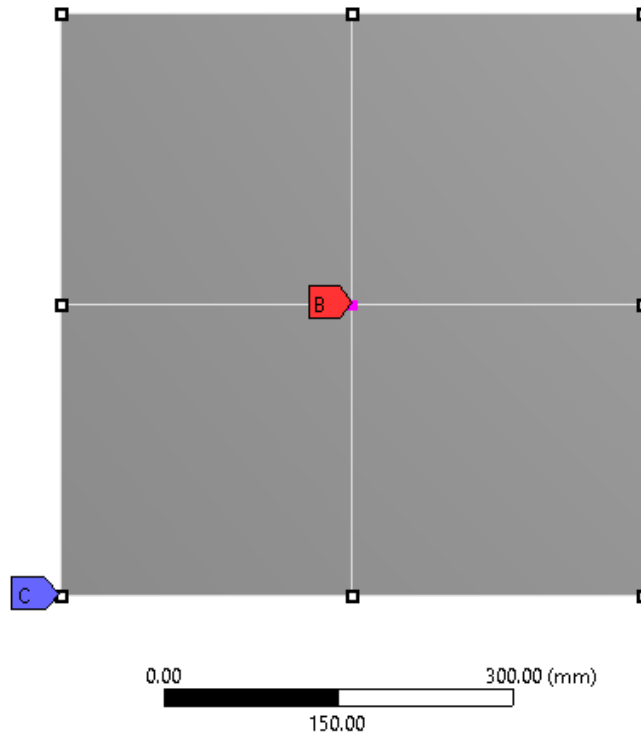
1	2	3	4	5	6	13	14	15	10	11	12
4	5	6	7	8	9	16	17	18	13	14	15
10	11	12	13	14	15	22	23	24	19	20	21
13	14	15	16	17	18	25	26	27	22	23	24

(1)

1.1 All edges simply supported

C: Static Structural
 Static Structural 2
 Time: 1. s

B Nodal Force: 1. N
 C Simply Supported: 0. mm



ANSYS
 2020 R1
 ACADEMIC

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

B: Static Structural
 Static Structural
 Time: 1. s

A Nodal Force: 1. N
B Fixed Support

ANSYS
 2020 R1
 ACADEMIC

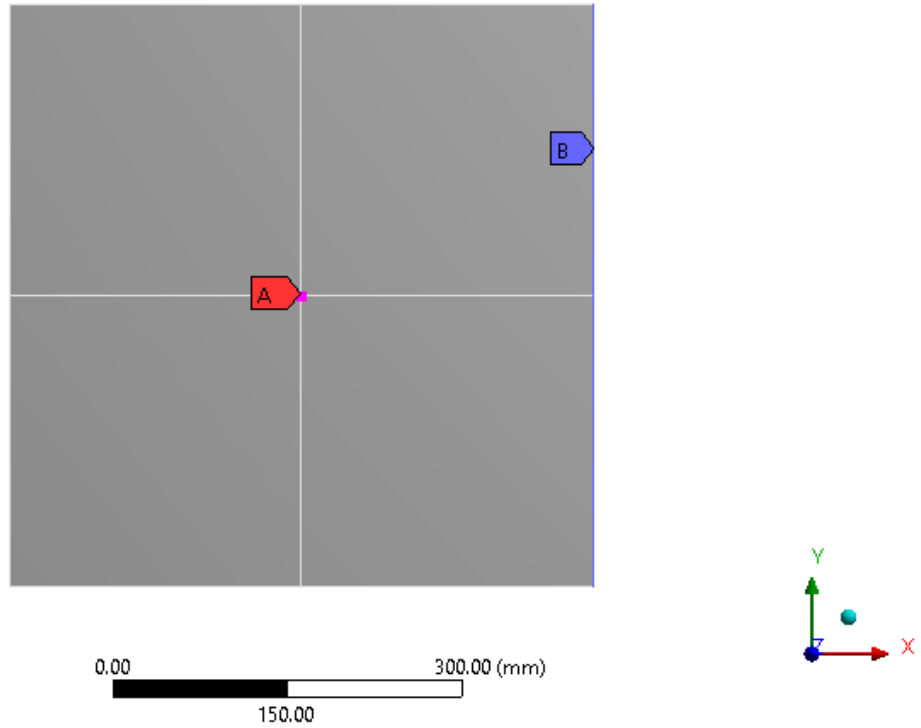


Figure 3: Boundary conditions for all edges clamped.

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.

B: Static Structural
Total Deformation
Type: Total Deformation
Unit: mm
Time: 1
Deformation Scale Factor: 2.e+002 (Auto Scale)

ANSYS
2020 R1
ACADEMIC

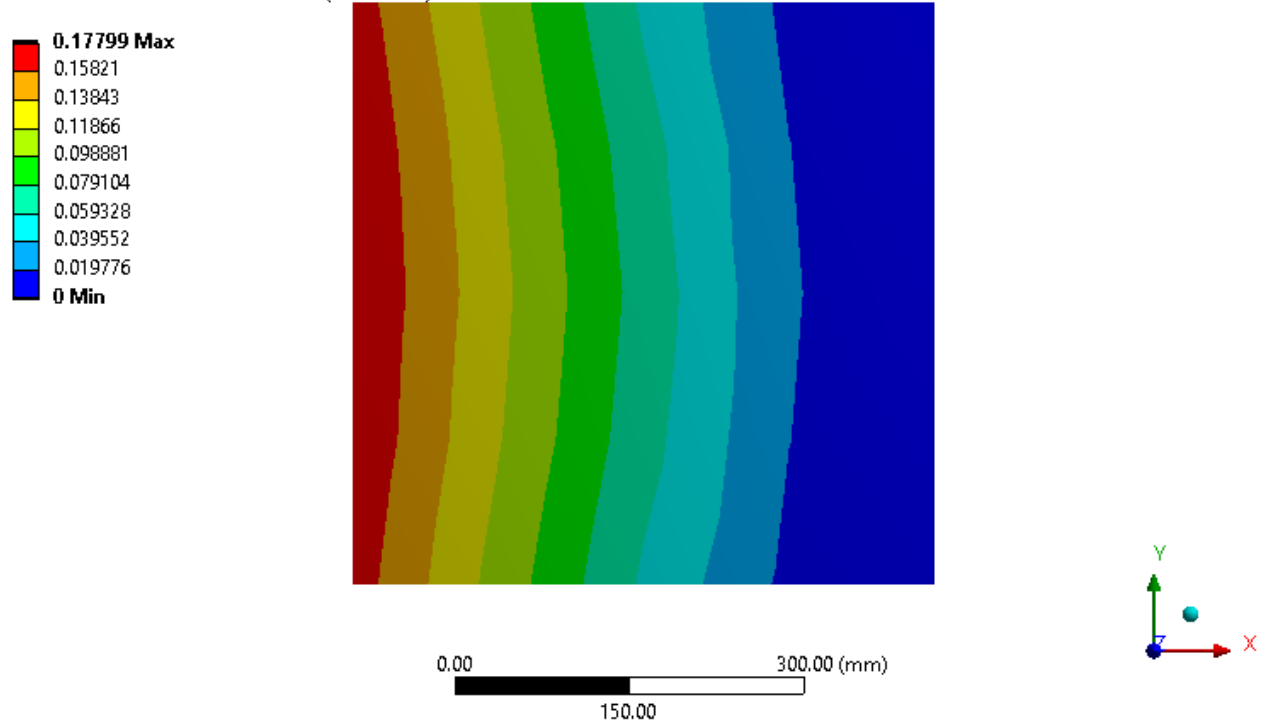


Figure 4: Deflections for right edge clamped.

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
ϵ_{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
ϵ_{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
ϵ_{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.

C: Static Structural

Normal Stress

Type: Normal Stress(X Axis) - Top/Bottom

Unit: MPa

Global Coordinate System

Time: 1

Deformation Scale Factor: 1.8e+003 (Auto Scale)

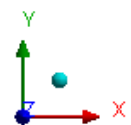
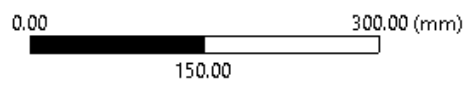
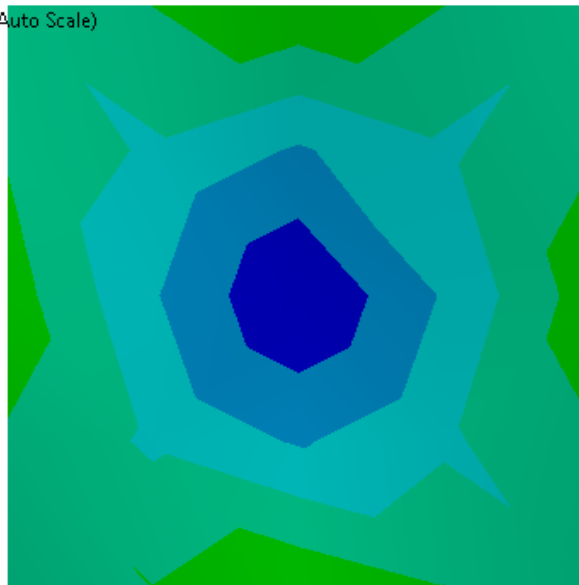
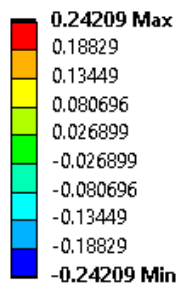


Figure 5: Normal stress (x-axis)

C: Static Structural

Normal Stress 2

Type: Normal Stress(Y Axis) - Top/Bottom

Unit: MPa

Global Coordinate System

Time: 1

Deformation Scale Factor: 1.8e+003 (Auto Scale)

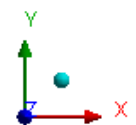
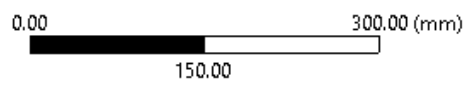
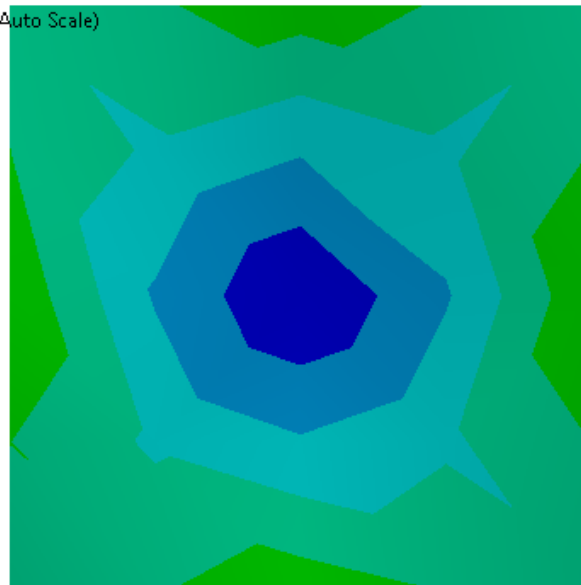
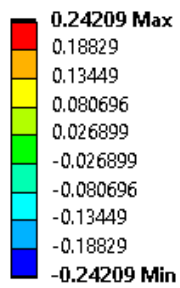


Figure 6: Normal stress (y-axis)

C: Static Structural

Shear Stress

Type: Shear Stress(XY Component) - Top/Bottom

Unit: MPa

Global Coordinate System

Time: 1

Deformation Scale Factor: 1.8e+003 (Auto Scale)

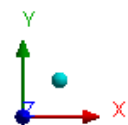
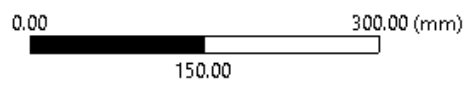
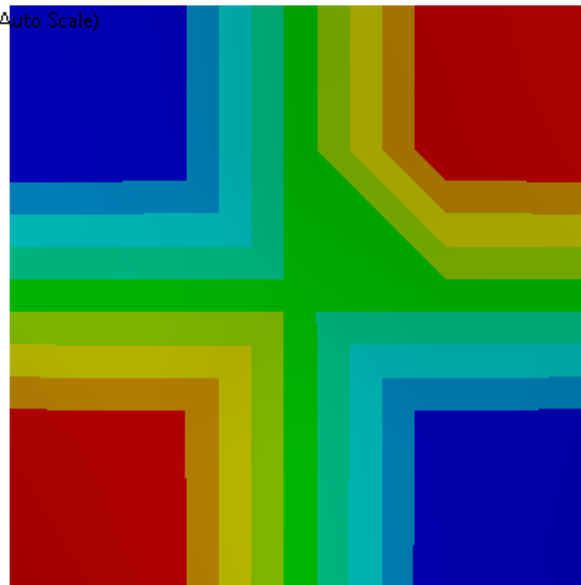
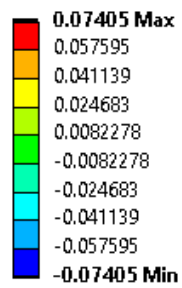


Figure 7: Shear stress

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
ϵ_{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
ϵ_{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
ϵ_{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 calculated at this points, so they are the most accurate points for these properties.

B: Static Structural

Normal Stress

Type: Normal Stress(X Axis) - Top/Bottom

Unit: MPa

Global Coordinate System

Time: 1

Deformation Scale Factor: 2.e+002 (Auto Scale)

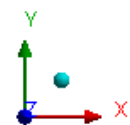
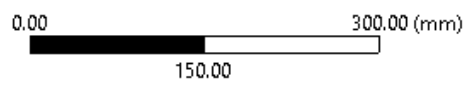
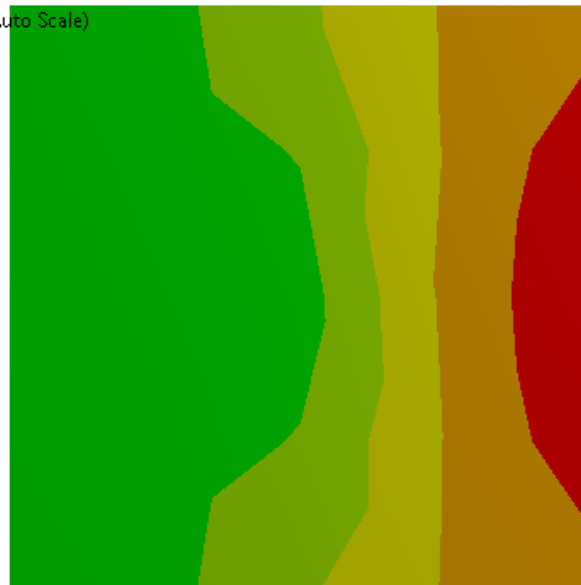
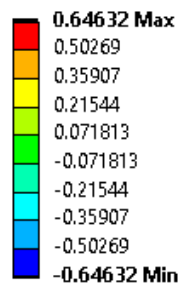


Figure 8: Normal stress (x-axis)

B: Static Structural

Normal Stress 2

Type: Normal Stress(Y Axis) - Top/Bottom

Unit: MPa

Global Coordinate System

Time: 1

Deformation Scale Factor: 2.e+002 (Auto Scale)

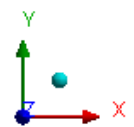
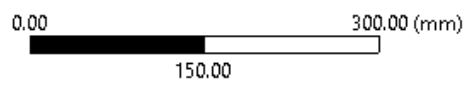
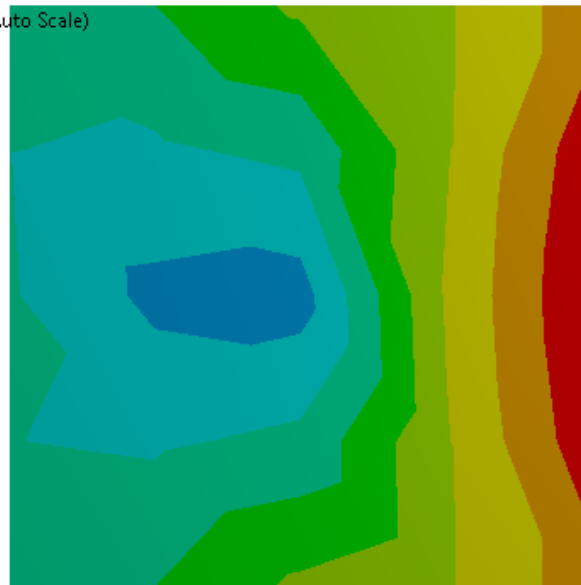
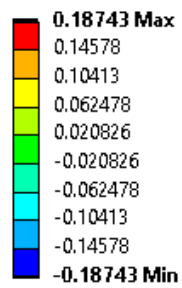


Figure 9: Normal stress (y-axis)

B: Static Structural

Normal Stress 2

Type: Normal Stress(Y Axis) - Top/Bottom

Unit: MPa

Global Coordinate System

Time: 1

Deformation Scale Factor: 2.e+002 (Auto Scale)

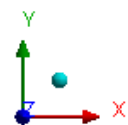
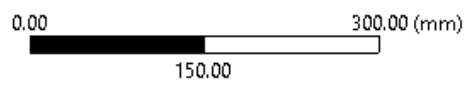
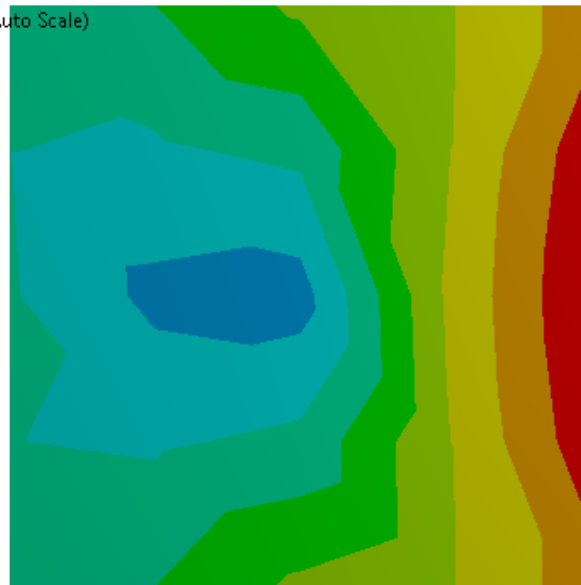
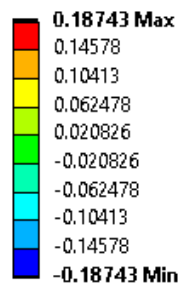


Figure 10: Shear stress

Matlab Scripts

Listing 1: Matlab code for question 1

```
1 % HW3
2
3 clc
4 clear all
5 close all
6 format compact
7 format short e
8 tic
9
10 % input data for control parameters
11 nel=4; % number of elements
12 nnel=4; % number of nodes per element
13 ndof=3; % number of dots per node
14 nnode=9; % total number of nodes in system
15 sdof=nnode*ndof; % total system dof
16 edof=nnel*ndof; % degrees of freedom per element
17 emodule=2.10e5; % elastic modulus
18 v=0.29; % Poisson's ratio
19 t=2; % plate thickness
20
21 E = emodule;
22
23 a=500;
24 b=500;
25 ro=a/b;
26 rom=b/a;
27 % input data for nodal coordinate values
28 % gcoord(ij) where i-> node no. and j-> x or y
29
30 top = [ 1 2 3 4 5 6 13 14 15 10 11 12
31         4 5 6 7 8 9 16 17 18 13 14 15
32         10 11 12 13 14 15 22 23 24 19 20 21
33         13 14 15 16 17 18 25 26 27 22 23 24];
34
35 F = (42-12*v+60*ro^2+60*rom^2)*t^2/a/b;
36 G = (30*ro+3*rom+12*v*rom)*t^2/b;
37 H = (30*rom+3*ro+12*v*ro)*t^2/a;
38 I = (-42+12*v-60*ro^2+30*rom^2)*t^2/a/b;
39 J = (30*ro+3*(1-v)*rom)*t^2/b;
40 K = (15*rom-3*ro-12*v*ro)*t^2/a;
41 L = (-42+12*v-60*rom^2+30*rom^2)*t^2/a/b;
42 M = (-15*ro+3*rom+12*v*rom)*t^2/b;
43 N = (30*rom+3*(1-v)*ro)*t^2/a;
44 O = (42-12*v-30*ro^2-30*rom^2)*t^2/a/b;
45 P = (-15*ro+3*(1-v)*rom)*t^2/a;
46 Q = (15*rom-3*(1-v)*ro)*t^2/a;
47 R = (20*ro+4*(1-v)*ro)*t^2;
48 S = (10*ro-(1-v)*rom)*t^2;
49 T = (10*ro-4*(1-v)*rom)*t^2;
50 U = (5*ro+(1-v)*rom)*t^2;
51 V = (20*ro+4*(1-v)*ro)*t^2;
52 W = (20*rom+4*(1-v)*ro)*t^2;
53 X = (10*rom-4*(1-v)*ro)*t^2;
54 Y = (10*rom-(1-v)*ro)*t^2;
55 Z = 15*v*t^2;
```

```

56 |
57 | Ke = emodule*t/(180*(1-v^2))*[ F G -H L -M -N O -P -Q I J -K;
58 |   G R -Z -M -T 0 P U 0 -J S 0;
59 |   -H -Z V N 0 X Q 0 Y -K 0 W;
60 |   L -M N F G H I J K O -P Q;
61 |   -M -T 0 G R Z -J S 0 P U 0;
62 |   -N 0 X H Z V K 0 W -Q 0 Y;
63 |   O P Q I -J K F -G H L M N;
64 |   -P U 0 J S 0 -G R -Z M T 0;
65 |   -Q 0 Y K 0 W H -Z V -N 0 X;
66 |   I -J -K O P -Q L M -N F -G -H;
67 |   J S 0 -P U 0 M T 0 -G R Z;
68 |   -K 0 W Q 0 Y N 0 X -H Z V];
69 |
70 | Ke=double(Ke);
71 |
72 | Kglob=zeros(nnode*ndof);
73 | for nel=1:4
74 |     for i=1:12
75 |         for j=1:12
76 |             Kglob(top(nel,i),top(nel,j))= ...
77 |                 Kglob(top(nel,i),top(nel,j))+Ke(i,j);
78 |         end
79 |     end
80 | end
81 |
82 | % force vector
83 |
84 | Fglob=zeros(nnode*ndof,1);
85 | Fglob(13) = 1;
86 |
87 |
88 | %%Boundary conditions (All edges are simply supported)
89 | % BCs=[1;4;7;10;16;19; 22; 25];
90 |
91 | % Boundary conditions (Right edge is clamped)
92 | BCs=[7; 8; 9; 16; 17; 18; 25; 26; 27];
93 |
94 | activeDof=setdiff(1:nnode*ndof',BCs);
95 | Kglobactive=Kglob(activeDof,activeDof);
96 | Fglobactive=Fglob(activeDof,1);
97 |
98 | %Solve
99 | d = Kglobactive\Fglobactive
100 | do(activeDof,1)= d
101 |
102 | A=zeros(12,12,4);
103 | nodexcoors = [0 a 2*a];
104 | nodeycoors = [0 b 2*b];
105 |
106 | nc=
107 |     [ 0 0
108 |       0 a
109 |       0 2*a
110 |       b 0
111 |       b a
112 |       b 2*a
113 |       2*b 0
114 |       2*b a

```

```

114         2*b  2*a];
115
116 en =      [1  2  4  5
117            2  3  5  6
118            4  5  8  7
119            5  6  9  8];
120
121 for el = 1:4
122 A(:, :, el) = ...
123 [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)
124   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),
125   ,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(
126   el,1),1)*nc(en(el,1),2)^3;
127 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(
128   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)
129   ^2;
130 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)
131   *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
132   (el,1),2)^3;
133 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)
134   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),
135   ,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(
136   el,2),1)*nc(en(el,2),2)^3;
137 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(
138   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)
139   ^2;
140 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)
141   *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
142   (el,2),2)^3;
143 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)
144   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),
145   ,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(
146   el,3),1)*nc(en(el,3),2)^3;
147 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(
148   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)
149   ^2;
150 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)
151   *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
152   (el,3),2)^3;
153 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)
154   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),
155   ,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(
156   el,4),1)*nc(en(el,4),2)^3;
157 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(
158   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)
159   ^2;
160 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)
161   *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
162   (el,4),2)^3;
163     ];
164
165 end
166
167 do = [do
168       0
169       0
170       0];
171
172 alfa=zeros(12,1,4);

```



```

144 el=0;
145 for el=1:4
146     alfa (:, :, el)=A (:, :, el)\do(top(el, :))
147 end
148
149 ex=zeros(9,4);
150 ey=zeros(9,4);
151 gxy=zeros(9,4);
152
153 el=0;
154 z=1;
155
156 for el=1:4
157     for i = 1:9
158         ex(i, el)=2*z*( alfa(4,1, el)+3*alfa(7,1, el)*nc(i,1)+alfa(8,1, el)*nc(i,2)
159             +3*alfa(11,1, el)*nc(i,1)*nc(i,2));
160         ey(i, el)=2*z*( alfa(6,1, el)+3*alfa(10,1, el)*nc(i,2)+alfa(9,1, el)*nc(i,1)
161             +3*alfa(12,1, el)*nc(i,1)*nc(i,2));
162         gxy(i, el)=2*z*( alfa(5,1, el)+2*alfa(8,1, el)*nc(i,1)+2*alfa(9,1, el)*nc(i
163             ,2)+3*alfa(11,1, el)*nc(i,1)^2+3*alfa(12,1, el)*nc(i,2));
164     end
165 end
166
167 maxdo = max(do) %Maximum deflection
168
169 %Reaction forces and moments at nodes
170 Freaction=Kglob*do-Fglob;
171
172 gpoints = [ -0.577350269189626      -0.577350269189626;
173             0.577350269189626      -0.577350269189626;
174             0.577350269189626       0.577350269189626;
175             -0.577350269189626      0.577350269189626];
176
177 gweights = [1 1;
178             1 1;
179             1 1;
180             1 1];
181
182 E1 = E/(1-v^2);
183
184 ff = zeros(12,1);
185
186 N = zeros(1,12);
187 n = sym('n',[12 3]);
188 ns = sym('ns',[12 3]);
189 nt = sym('nt',[12 3]);
190 ns2 = sym('ns2',[12 3]);
191 nt2 = sym('nt2',[12 3]);
192 nsnt = sym('nsnt',[12 3]);
193
194 syms s t
195 for i=1:length(gweights)
196     for j=1:length(gweights)
197         for node=1:4
198             if node==1
199                 sc=-1;

```

```

199         tc=-1;
200         elseif node==2
201             sc=1;
202             tc=-1;
203         elseif node==3
204             sc=1;
205             tc=1;
206         else
207             sc=-1;
208             tc=1;
209         end
210         n(node,:) = [(1/4)*(1+sc*s)*(1+tc*t) ,...
211                    (1/4)*(1+sc*s)*(1+tc*t) ,...
212                    (1/4)*(1+sc*s)*(1+tc*t) ];
213     end
214
215     ns=diff(n,s);
216     nt=diff(n,t);
217     ns2=diff(n,s,2);
218     nt2=diff(n,t,2);
219     nsnt=diff(ns,t);
220
221     N=[n(1,:) n(2,:) n(3,:) n(4,:) ];
222     Ns2=[ns(1,:) ns(2,:) ns(3,:) ns(4,:) ];
223     Nt2=[nt(1,:) nt(2,:) nt(3,:) nt(4,:) ];
224     Nsnt=[nsnt(1,:) nsnt(2,:) nsnt(3,:) nsnt(4,:) ];
225
226     B=[Ns2/a^2; Nt2/b^2; 2/(a*b)*Nsnt];
227
228     D=[ E1 E1*v 0;...
229         E1*v E1 0;...
230         0 0 E/(2*(1+v)) ];
231
232     %Element consistent load vector
233     ff=a*b*N'*1*gweights(i)*gweights(j);
234     ff=subs(ff,s,gpoints(i));
235     ff=subs(ff,t,gpoints(j));
236 end
237 end
238
239 %Strain at nodes (eps x, eps y, gama xy)
240 B1=subs(B,s,-1);
241 B1=subs(B1,t,1);
242 B1=double(B1);
243
244 strain=zeros(3,1,9);
245
246 e=0;
247
248 for i=[4 5 7 8] %Nodes [1:4 6:9 11:14 16:19]
249     e=e+1;
250     strain(:,:,i)=B1*do(top(e,:));
251 end
252
253 B2=subs(B,s,1);
254 B2=subs(B2,t,1);
255 B2=double(B2);
256

```

```

257 e=0;
258
259 for i=[9 6] %Nodes [5 10 15 20]
260     e=e+2;
261     strain (:,:,i)=B2*do(top(e,:));
262 end
263
264 B3=subs(B,s,-1);
265 B3=subs(B3,t,-1);
266 B3=double(B3);
267
268 e=1;
269 for i=[1 2] %Nodes [21 24]
270     e=e+1;
271     strain (:,:,i)=B3*do(top(e,:));
272 end
273
274 B4=subs(B,s,1);
275 B4=subs(B4,t,-1);
276 B4=double(B4);
277 strain (:,:,3)=B4*do(top(2,:)); %Node 25
278
279 %Stress at nodes (sigma x, sigma y,tau xy)
280 stress=zeros(3,1,9);
281 for i=1:9
282     stress (:,:,i)=D*strain (:,:,i)
283 end
284
285
286 toc

```