

UCL Mechanical Engineering 2020/2021

ENGF0004 Coursework 2

NCWT3

April 19, 2021

1 Question 1

1.1 a

For the line integral to be independent from the path of integration, the following conditions must be fulfilled:

$$I = \int_A^B \left(\frac{\partial u}{\partial x} dx + \frac{\partial u}{\partial y} dy \right) \quad (1.1)$$

$$P(x, y) = \frac{\partial u}{\partial x} \text{ and } Q(x, y) = \frac{\partial u}{\partial y} \quad (1.2)$$

$$\frac{\partial P(x, y)}{\partial y} = \frac{\partial Q(x, y)}{\partial x} \quad (1.3)$$

Considering the integral:

$$I = \int_A^B \left[e^{-\alpha xy} \left(\frac{\alpha - 2}{x} \right) dx - \frac{1}{\alpha y} (e^{-\alpha xy} - 1) dy \right] \quad (1.4)$$

$$P(x, y) = e^{-\alpha xy} \left(\frac{\alpha - 2}{x} \right) \text{ and } Q(x, y) = -\frac{1}{\alpha y} (e^{-\alpha xy} - 1) \quad (1.5)$$

$$\frac{\partial P(x, y)}{\partial y} = -\alpha x \left(\frac{\alpha - 2}{x} \right) e^{-\alpha xy} = (2\alpha - \alpha^2) e^{-\alpha xy} \quad (1.6)$$

$$\frac{\partial Q(x, y)}{\partial x} = -\frac{1}{\alpha y} (-\alpha y) e^{-\alpha xy} = e^{-\alpha xy} \quad (1.7)$$

$$\therefore 2\alpha e^{-\alpha xy} - \alpha^2 e^{-\alpha xy} = e^{-\alpha xy} \quad (1.8)$$

$$e^{-\alpha xy} (\alpha^2 - 2\alpha + 1) = 0 \quad (1.9)$$

$$e^{-\alpha xy} = 0 \rightarrow \text{no solutions} \quad (1.10)$$

$$(\alpha - 1)^2 = 0 \quad (1.11)$$

$$\alpha = 1 \quad (1.12)$$

1.2 b

Calculating the line integral of 1.13 from $O(0, 0)$ to $A(1, e - 1)$ along $y = e^x - 1$:

$$I = \int_O^A (ye^{-2x}) (dx + dy) \quad (1.13)$$

$$y = e^x - 1 \quad (1.14)$$

$$dy = e^x dx \quad (1.15)$$

$$I = \int_0^1 \left((e^x - 1)(e^{-2x}) + (e^x - 1)(e^{-2x})(e^x) \right) dx \quad (1.16)$$

$$= \int_0^1 (e^{-x} - e^{-x} - e^{-2x} + 1) dx \quad (1.17)$$

$$= \int_0^1 (1 - e^{-2x}) dx \quad (1.18)$$

$$= \left[x + \frac{e^{-2x}}{2} \right]_0^1 \quad (1.19)$$

$$= 1 + \frac{e^{-2}}{2} - 0 - \frac{1}{2} \quad (1.20)$$

$$I = \frac{1}{2} (e^{-2} + 1) \quad (1.21)$$

1.3 c

1.3.1 i

$$\underline{F}(x, y, z) = \begin{pmatrix} \frac{y}{x^2} \\ \frac{x}{y^2} \end{pmatrix} \quad (1.22)$$

$$\nabla \cdot \underline{F} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} \cdot \begin{pmatrix} \frac{y}{x^2} \\ \frac{x}{y^2} \end{pmatrix} \quad (1.23)$$

$$= \frac{\partial}{\partial x} \left(\frac{y}{x^2} \right) + \frac{\partial}{\partial y} \left(\frac{x}{y^2} \right) \quad (1.24)$$

$$= -\frac{2y}{x^3} - \frac{2x}{y^3} \quad (1.25)$$

$$= -2 \left(\frac{y}{x^3} + \frac{x}{y^3} \right) \quad (1.26)$$

1.3.2 ii

$$I = \int_1^2 \int_1^2 \left(-2 \left(\frac{y}{x^3} + \frac{x}{y^3} \right) \right) dx dy \quad (1.27)$$

$$= \int_1^2 \left[-2 \left(\frac{y}{-2x^2} + \frac{x^2}{2y^3} \right) \right]_1^2 dy \quad (1.28)$$

$$= \int_1^2 \left[-2 \left(-\frac{y}{8} + \frac{2}{y^3} + \frac{y}{2} - \frac{1}{2y^3} \right) \right] dy \quad (1.29)$$

$$= \int_1^2 \left(-\frac{3y}{4} - \frac{3}{y^3} \right) dy \quad (1.30)$$

$$= \left[-\frac{3y^2}{8} + \frac{3}{2y^2} \right]_1^2 \quad (1.31)$$

$$= -\frac{3}{2} + \frac{3}{8} + \frac{3}{8} - \frac{3}{2} \quad (1.32)$$

$$I = -\frac{9}{4} \quad (1.33)$$

1.4 d

1.4.1 i

$$I = \int (\sin x \cos y dy + \cos x \sin y dx) \quad (1.34)$$

$$y = 0 \quad dy = 0 \quad (1.35)$$

$$I_{AB} = \int_{x=0}^{\pi} (\sin x) dx = [-\cos x]_0^{\pi} = 2 \quad (1.36)$$

$$x = \pi \quad dx = 0 \quad (1.37)$$

$$I_{BC} = \int_{y=0}^{\pi} (-\sin y) dy = [\cos y]_0^{\pi} = -2 \quad (1.38)$$

$$\therefore I = I_{AB} + I_{BC} = 2 - 2 = 0 \quad (1.39)$$

1.4.2 ii

$$I = \int (\sin x \cos y dy + \cos x \sin y dx) \quad (1.40)$$

$$y = x \quad dy = dx \quad (1.41)$$

$$I_{AC} = \int_0^{\pi} (\sin x \cos x + \sin x \cos x) dx \quad (1.42)$$

$$= \int_0^{\pi} (\sin (2x)) dx \quad (1.43)$$

$$I_{AC} = \left[-\frac{1}{2} \cos (2x) \right]_0^{\pi} = \frac{1}{2} - \frac{1}{2} = 0 \quad (1.44)$$

$$(1.45)$$

1.5 e

1.5.1 i

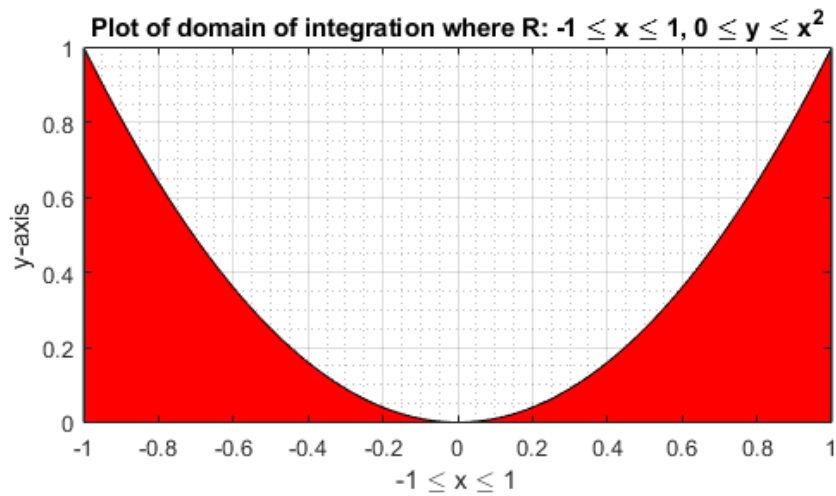


Figure 1: Domain of integration where $R: -1 \leq x \leq 1, 0 \leq y \leq x^2$.

```
1  clc
2  clear
3  close all
4
5  %mesh
6  m = -2:0.1:2;
7  [x,y] = meshgrid(m);
8
9  %function
10 z = x.*y.*exp(-sqrt(x.^2 + y.^2));
11
12 %gradient function
13 [gx,gy] = gradient(z,0.2,0.2);
14 contour(m,m,z)
15 hold on
16 quiver(m,m,gx,gy)
17 hold off
18
19 %formatting
20 axis('image');
21 xlabel('x-axis')
22 ylabel('y-axis')
23 title('Plot of scalar function and gradient vectors')
24 grid on
25 grid minor
```

1.5.2 ii

$y = 0 \rightarrow y = x^2$ and $x = -1 \rightarrow x = 1$

$$I = \int_{-1}^1 \int_0^{x^2} \left(y \frac{\sin(\pi x)}{x} \right) dy dz \quad (1.46)$$

$$I = \int_{-1}^1 \left[y^2 \frac{\sin(\pi x)}{2x} \right]_0^{x^2} dx \quad (1.47)$$

$$I = \int_{-1}^1 \left[\frac{x^4 \sin(\pi x)}{2x} - 0 \right] dx \quad (1.48)$$

$$I = \int_{-1}^1 \left(\frac{x^3 \sin(\pi x)}{2} \right) dx \quad (1.49)$$

Integration by parts thrice:

$$u_x = \frac{x^3}{2} \quad u'_x = \frac{3x^2}{2} \quad (1.50)$$

$$v_x = -\frac{\cos(\pi x)}{\pi} \quad v'_x = \sin(\pi x) \quad (1.51)$$

$$I = \left[-\frac{x^3 \cos(\pi x)}{2\pi} \right]_{-1}^1 + \int_{-1}^1 \left(\frac{3x^2 \cos(\pi x)}{2\pi} \right) dx \quad (1.52)$$

$$u_x = \frac{3x^2}{2\pi} \quad u'_x = \frac{3x}{\pi} \quad (1.53)$$

$$v_x = \frac{\sin(\pi x)}{\pi} \quad v'_x = \cos(\pi x) \quad (1.54)$$

$$I = -\frac{\cos \pi}{2\pi} - \frac{\cos(-\pi)}{2\pi} + \left[\frac{3x^2 \sin(\pi x)}{2x^2} \right]_{-1}^1 - \int_{-1}^1 \left(\frac{3x \sin(\pi x)}{\pi^2} \right) dx \quad (1.55)$$

$$I = \frac{1}{2\pi} + \frac{1}{2\pi} + 0 - 0 - \int_{-1}^1 \left(\frac{3x \sin(\pi x)}{\pi^2} \right) dx \quad (1.56)$$

$$u_x = \frac{3x}{\pi^2} \quad u'_x = \frac{3}{\pi^2} \quad (1.57)$$

$$v_x = -\frac{\cos(\pi x)}{\pi} \quad v'_x = \sin(\pi x) \quad (1.58)$$

$$I = \frac{1}{\pi} + \left[\frac{3x \cos(\pi x)}{\pi^3} \right]_{-1}^1 - \int_{-1}^1 \left(\frac{3 \cos(\pi x)}{\pi^3} \right) dx \quad (1.59)$$

$$I = \frac{1}{\pi} - \frac{3}{\pi^3} - \frac{3}{\pi^3} - \left[\frac{3 \sin \pi}{\pi^4} - \frac{3 \sin(-\pi)}{\pi^4} \right] \quad (1.60)$$

$$I = \frac{1}{\pi} - \frac{6}{\pi^3} = 0.125 \quad (1.61)$$

1.5.3 iii

Utilising symmetry, we can select the limits $x = 0, x = 1$:

$$I = \int_0^1 \int_{\sqrt{y}}^1 (x^2 + y^2) dx dy \quad (1.62)$$

$$= \int_0^1 \left[\frac{x^3}{3} + xy^2 \right]_{\sqrt{y}}^1 dy \quad (1.63)$$

$$= \int_0^1 \left(\frac{1}{3} + y^2 - \frac{y^{\frac{3}{2}}}{3} - y^{\frac{5}{2}} \right) dy \quad (1.64)$$

$$= \left[\frac{y}{3} + \frac{y^3}{3} - \frac{2y^{\frac{5}{2}}}{15} - \frac{2y^{\frac{7}{2}}}{7} \right]_0^1 \quad (1.65)$$

$$= \frac{1}{3} + \frac{1}{3} - \frac{2}{15} - \frac{2}{7} \quad (1.66)$$

$$I = \frac{26}{105} \quad (1.67)$$

Doubling this value yields:

$$I = \frac{52}{105} \quad (1.68)$$

1.6 f

1.6.1 i

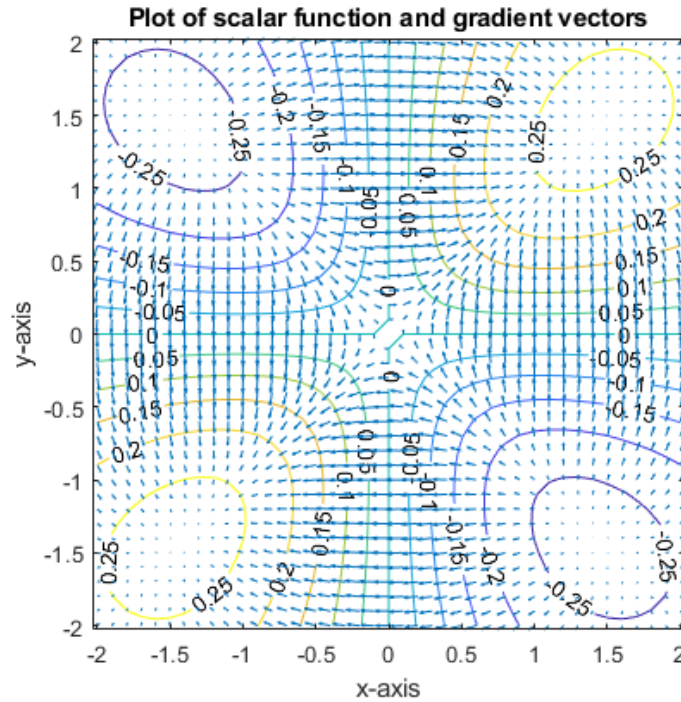


Figure 2:

```

2  clear
3  close all
4
5  %mesh
6  m = -2:0.1:2;
7  [x,y] = meshgrid(m);
8
9  %function
10 z = x.*y.*exp(-sqrt(x.^2 + y.^2));
11
12 %gradient function
13 [gx,gy] = gradient(z,0.2,0.2);
14 contour(m,m,z,'showtext','on')
15 hold on
16 quiver(m,m,gx,gy)
17 hold off
18
19 %formatting
20 axis('image');
21 xlabel('x-axis')
22 ylabel('y-axis')
23 title('Plot of scalar function and gradient vectors')

```

1.6.2 ii

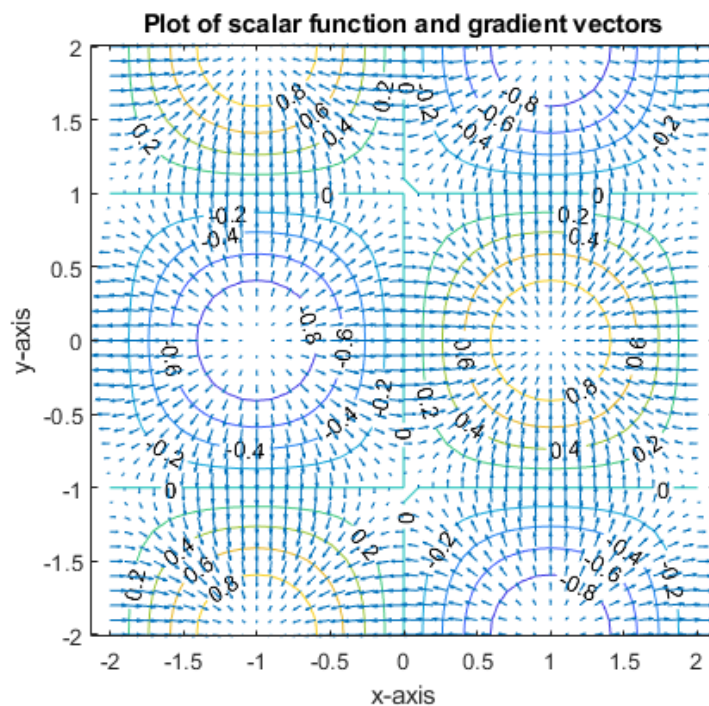


Figure 3:

```

1  clc
2  clear
3  close all
4
5  %mesh

```

```

6  m = -2:0.1:2;
7  [x,y] = meshgrid(m);
8
9  %function
10 z = (sin((pi/2).*x)).*(cos((pi/2).*y));
11
12 %gradient function
13 [gx,gy] = gradient(z,0.2,0.2);
14 contour(m,m,z,'showtext','on')
15 hold on
16 quiver(m,m,gx,gy)
17 hold off
18
19 %formatting
20 axis('image');
21 xlabel('x-axis')
22 ylabel('y-axis')
23 title('Plot of scalar function and gradient vectors')

```

1.7 g

1.7.1 i

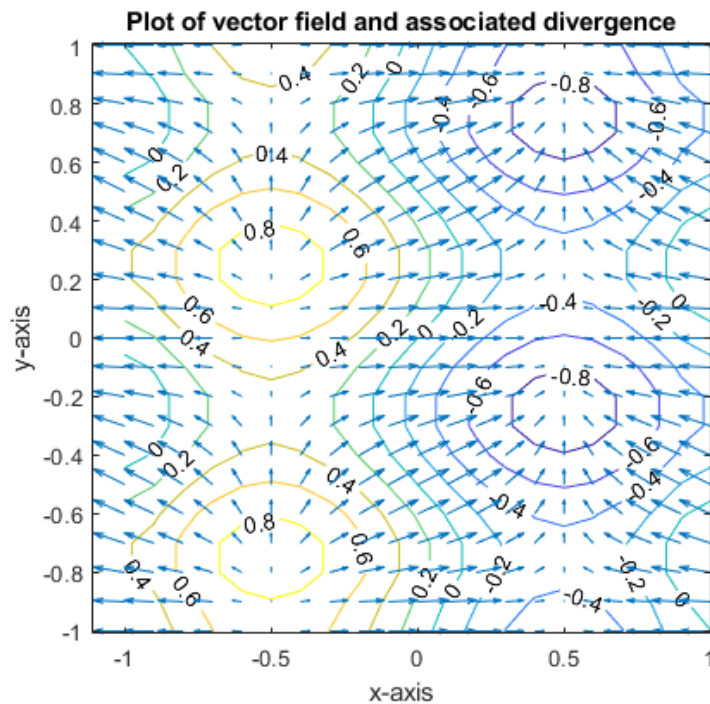


Figure 4:

```

1  clc
2  clear
3  close all
4
5  %mesh
6  m = -1:0.1:1;

```



```

7  [x,y] = meshgrid(m);
8
9  %function
10 ui = 2.*cos(pi.*x);
11 uj = (sin(pi.*y)).^2;
12
13 %divergence
14 d = divergence(ui,uj);
15 contour(m,m,d, 'showtext', 'on')
16 hold on
17 quiver(m,m, ui , uj)
18 hold off
19
20 %formatting
21 axis('image');
22 xlabel('x-axis')
23 ylabel('y-axis')
24 title('Plot of vector field and associated divergence')

```

1.7.2 ii

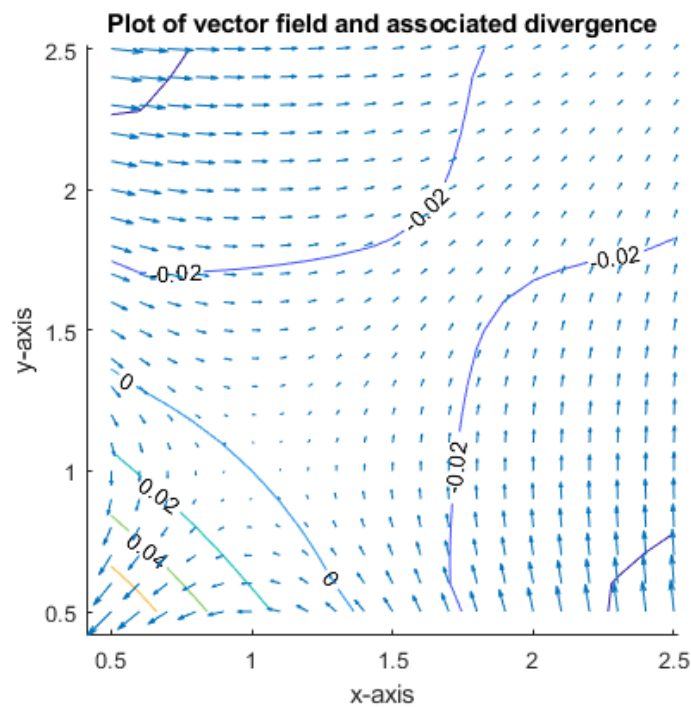


Figure 5:

```

1  clc
2  clear
3  close all
4
5  %mesh
6  m = 0.5:0.1:2.5;
7  [x,y] = meshgrid(m);
8
9  %function

```

```

10 ui = log(y).*exp(-x);
11 uj = log(x).*exp(-y);
12
13 %divergence
14 d = divergence(ui,uj);
15 hold on
16 contour(m,m,d,'showtext','on')
17 quiver(m,m,ui,uj)
18 hold off
19
20 %formatting
21 axis('image');
22 xlabel('x-axis')
23 ylabel('y-axis')
24 title('Plot of vector field and associated divergence')

```

1.8 h

1.8.1 i

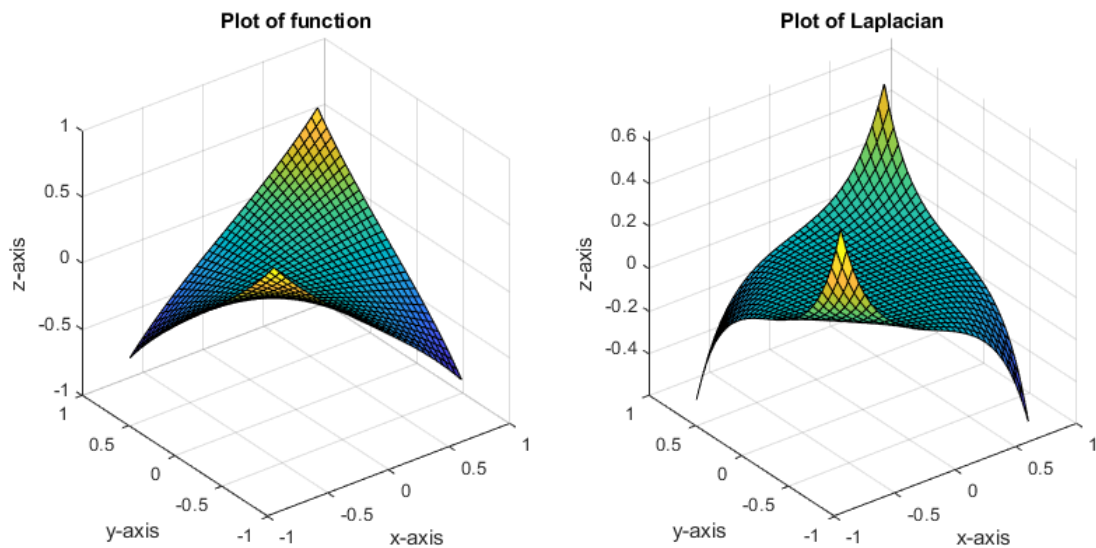


Figure 6:

```

1 clc
2 clear
3 close all
4
5 %mesh
6 m = -pi/4:0.05:pi/4;
7 [x,y] = meshgrid(m);
8
9 %function
10 z = tan(x.*y);
11
12 %laplacian

```

```

13 L = del2(z,0.05);
14
15 %plotting
16 subplot(1,2,1)
17 surf(x,y,z)
18 axis('square');
19 xlabel('x-axis')
20 ylabel('y-axis')
21 zlabel('z-axis')
22 title('Plot of function')
23 subplot(1,2,2)
24 surf(x,y,L)
25 axis('square');
26 xlabel('x-axis')
27 ylabel('y-axis')
28 zlabel('z-axis')
29 title('Plot of Laplacian')

```

1.8.2 ii

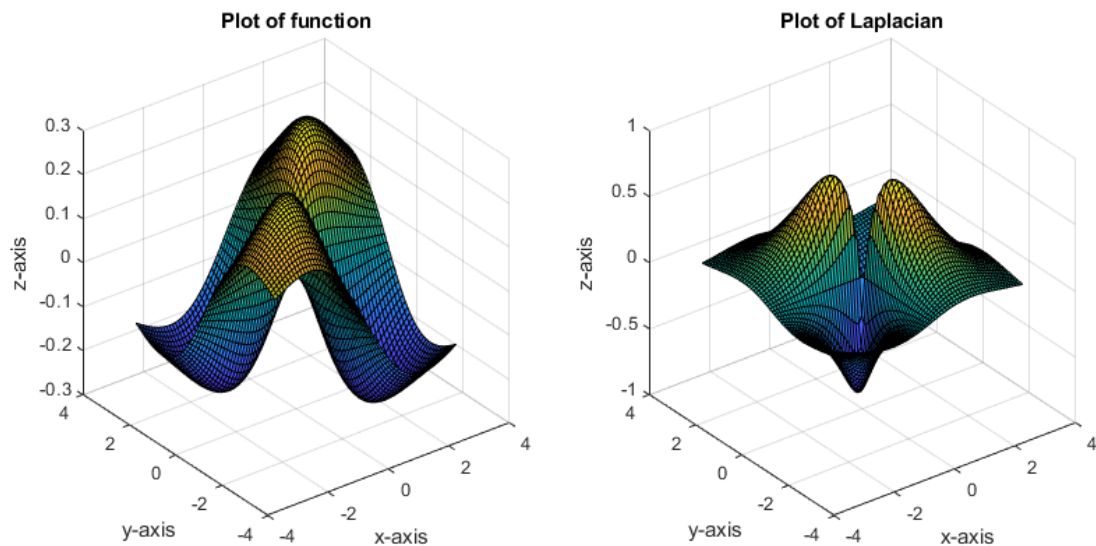


Figure 7:

```

1 clc
2 clear
3 close all
4
5 %mesh
6 m = -3:0.1:3;
7 [x,y] = meshgrid(m);
8
9 %function
10 z = x.*y.*exp(-sqrt(x.^2 + y.^2));
11
12 %laplacian
13 L = del2(z,0.05);

```

```

14
15 %plotting
16 subplot(1,2,1)
17 surf(x,y,z)
18 axis('square');
19 xlabel('x-axis')
20 ylabel('y-axis')
21 zlabel('z-axis')
22 title('Plot of function')
23 subplot(1,2,2)
24 surf(x,y,L)
25 axis('square');
26 xlabel('x-axis')
27 ylabel('y-axis')
28 zlabel('z-axis')
29 title('Plot of Laplacian')

```

2 Question 2

2.1 a

In our series of equations, there are three unknown internal bar forces N_{12} , N_{23} , N_{13} , and three unknown reaction forces, R_{2x} , R_{2y} , R_{3y} . We also have two unknown angles, α and β , and the force F . Given that there are six unknowns that we would like to find and six equations with those variables, the conditions are fulfilled to solve this using matrices. Our answer would be in terms of the variables α , β and F . Values may be assumed for these or we can calculate them based on the geometry of the beams.

2.2 b

$$\begin{bmatrix} -\cos \alpha & \cos \beta & 0 & 0 & 0 & 0 \\ -\sin \alpha & -\sin \beta & 0 & 0 & 0 & 0 \\ \cos \alpha & 1 & 0 & 1 & 0 & 0 \\ \sin \alpha & 0 & 0 & 0 & 1 & 0 \\ 0 & -\cos \beta & -1 & 0 & 0 & 0 \\ 0 & \sin \beta & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} N_{12} \\ N_{13} \\ N_{23} \\ R_{2x} \\ R_{2y} \\ R_{3y} \end{bmatrix} = \begin{bmatrix} 0 \\ F \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.1)$$

2.3 c

```

1 clc
2 clear
3 close all
4
5 alpha = 0.927295;
6 beta = 0.643501;
7 F = 1000;
8

```

```

9  A = [-cos(alpha) cos(beta) 0 0 0 0;
10      -sin(alpha) -sin(beta) 0 0 0 0;
11      cos(alpha) 0 1 1 0 0;
12      sin(alpha) 0 0 0 1 0;
13      0 -cos(beta) -1 0 0 0;
14      0 sin(beta) 0 0 0 1];
15  B = [0; F; 0; 0; 0; 0];
16
17  sol = A\B;

```

This returned the following:

$$\begin{bmatrix} N_{12} \\ N_{13} \\ N_{23} \\ R_{2x} \\ R_{2y} \\ R_{3y} \end{bmatrix} = \begin{bmatrix} -800 \\ -600 \\ 480 \\ 0 \\ 640 \\ 360 \end{bmatrix} \quad (2.2)$$

2.4 d

```

1  clc
2  clear
3  close all
4
5  alpha = 0.927295;
6  beta = 0.643501;
7  F = 1000;
8
9  A = [-cos(alpha) cos(beta) 0 0 0 0;
10      -sin(alpha) -sin(beta) 0 0 0 0;
11      cos(alpha) 0 1 1 0 0;
12      sin(alpha) 0 0 0 1 0;
13      0 -cos(beta) -1 0 0 0;
14      0 sin(beta) 0 0 0 1];
15  B = [0; F; 0; 0; 0; 0];
16
17  [L,U] = lu(A); %splits matrix A such that A = L*U, L is lower triangular,
                  U is upper triangular
18  y = L\B;
19  sol = U\y;

```

This returned the following:

$$\begin{bmatrix} N_{12} \\ N_{13} \\ N_{23} \\ R_{2x} \\ R_{2y} \\ R_{3y} \end{bmatrix} = \begin{bmatrix} -800 \\ -600 \\ 480 \\ 0 \\ 640 \\ 360 \end{bmatrix} \quad (2.3)$$

2.5 e

Matlab App Developer was utilised to create a user friendly interface for inputting the Force F , the lengths of each member (as shown in the diagram) and the coefficient matrix (where any mathematical expression can be inputted). The GUI displays the angles α and β as well as a table of values for each of the internal bar and reaction forces. The code is shown below.

```

1  classdef q2eApp_exported < matlab.apps.AppBase
2
3      % Properties that correspond to app components
4      properties (Access = public)
5          UIFigure          matlab.ui.Figure
6          ForceNLabel        matlab.ui.control.Label
7          Force              matlab.ui.control.NumericEditField
8          L12mLabel          matlab.ui.control.Label
9          L12                matlab.ui.control.NumericEditField
10         L23mLabel          matlab.ui.control.Label
11         L23                matlab.ui.control.NumericEditField
12         L13mLabel          matlab.ui.control.Label
13         L13                matlab.ui.control.NumericEditField
14         UITable             matlab.ui.control.Table
15         FindForcesButton    matlab.ui.control.Button
16         betaGaugeLabel      matlab.ui.control.Label
17         betaGauge           matlab.ui.control.NinetyDegreeGauge
18         alphaGaugeLabel     matlab.ui.control.Label
19         alphaGauge          matlab.ui.control.NinetyDegreeGauge
20         ProgrammetocalculateforcesLabel  matlab.ui.control.Label
21         UITable2            matlab.ui.control.Table
22         Label               matlab.ui.control.Label
23     end
24
25     % Callbacks that handle component events
26     methods (Access = private)
27
28         % Code that executes after component creation
29         function startupFcn(app)
30             %initialise table
31             ATable = ["-cos(alpha)" "cos(beta)" "0" "0" "0" "0";
32                     "-sin(alpha)" "-sin(beta)" "0" "0" "0" "0";
33                     "cos(alpha)" "0" "1" "1" "0" "0";
34                     "sin(alpha)" "0" "0" "0" "1" "0";
35                     "0" "-cos(beta)" "-1" "0" "0" "0";
36                     "0" "sin(beta)" "0" "0" "0" "1"];
37             %display table and assign table properties
38             set(app.UITable2, 'Visible', 'on');
39             set(app.UITable2, 'Data', ATable, 'ColumnFormat', {'char'});
40             set(app.UITable2, 'ColumnEditable', true(1,6))
41         end
42
43         % Button pushed function: FindForcesButton
44         function FindForcesButtonPushed(app, event)
45             %calculate alpha and beta
46             alpha = acos((app.L12.Value^2 + app.L23.Value^2 - app.L13.
                     Value^2)/(2*app.L12.Value*app.L23.Value));

```

```

47         beta = acos((app.L13.Value^2 + app.L23.Value^2 - app.L12.
           Value^2)/(2*app.L13.Value*app.L23.Value));
48
49     %conversion for display gauges
50     app.alphaGauge.Value = rad2deg(alpha);
51     app.betaGauge.Value = rad2deg(beta);
52
53     %matrix maths
54     A = get(app.UITable2, 'Data');
55     %convert user inputs into expressions and evaluate
56     c = size(A);
57     c = c(1)*c(2);
58     for i = 1:c
59         A(i) = eval(A(i));
60     end
61     A = str2double(A);
62     B = [0; app.Force.Value; 0; 0; 0; 0];
63     sol = A\B;
64     for i = 1:length(B)
65         if sol(i) < 0.01 && sol(i) > -0.01
66             sol(i) = 0;
67         end
68     end
69     namesForces = [" L12";" L13";" L23";" R2x";" R2y";" R3y"];
70     vars = [namesForces sol];
71
72     %output to table
73     set(app.UITable, 'Visible', 'on');
74     set(app.UITable, 'Data', vars, 'ColumnFormat',{ 'numeric' });
75
76     end
77 end
78
79 % Component initialization
80 methods (Access = private)
81
82     % Create UIFigure and components
83     function createComponents(app)
84
85         % Create UIFigure and hide until all components are created
86         app.UIFigure = uifigure('Visible', 'off');
87         app.UIFigure.Position = [100 100 762 598];
88         app.UIFigure.Name = 'MATLAB App';
89
90         % Create ForceNLabel
91         app.ForceNLabel = uilabel(app.UIFigure);
92         app.ForceNLabel.HorizontalAlignment = 'right';
93         app.ForceNLabel.Position = [32 403 56 22];
94         app.ForceNLabel.Text = 'Force (N)';
95
96         % Create Force
97         app.Force = uieditfield(app.UIFigure, 'numeric');
98         app.Force.Position = [103 403 100 22];

```

```

99
100 % Create L12mLabel
101 app.L12mLabel = uilabel(app.UIFigure);
102 app.L12mLabel.HorizontalAlignment = 'right';
103 app.L12mLabel.Position = [41 370 47 22];
104 app.L12mLabel.Text = 'L12 (m)';
105
106 % Create L12
107 app.L12 = uieditfield(app.UIFigure, 'numeric');
108 app.L12.Position = [103 370 100 22];
109
110 % Create L23mLabel
111 app.L23mLabel = uilabel(app.UIFigure);
112 app.L23mLabel.HorizontalAlignment = 'right';
113 app.L23mLabel.Position = [41 349 47 22];
114 app.L23mLabel.Text = 'L23 (m)';
115
116 % Create L23
117 app.L23 = uieditfield(app.UIFigure, 'numeric');
118 app.L23.Position = [103 349 100 22];
119
120 % Create L13mLabel
121 app.L13mLabel = uilabel(app.UIFigure);
122 app.L13mLabel.HorizontalAlignment = 'right';
123 app.L13mLabel.Position = [41 328 47 22];
124 app.L13mLabel.Text = 'L13 (m)';
125
126 % Create L13
127 app.L13 = uieditfield(app.UIFigure, 'numeric');
128 app.L13.Position = [103 328 100 22];
129
130 % Create UITable
131 app.UITable = uitable(app.UIFigure);
132 app.UITable.ColumnName = {'Force'; 'Value (N)'};
133 app.UITable.RowName = {};
134 app.UITable.Position = [272 59 479 185];
135
136 % Create FindForcesButton
137 app.FindForcesButton = uibutton(app.UIFigure, 'push');
138 app.FindForcesButton.ButtonPushedFcn = createCallbackFcn(app,
    @FindForcesButtonPushed, true);
139 app.FindForcesButton.Position = [103 292 100 22];
140 app.FindForcesButton.Text = 'Find Forces';
141
142 % Create betaGaugeLabel
143 app.betaGaugeLabel = uilabel(app.UIFigure);
144 app.betaGaugeLabel.HorizontalAlignment = 'center';
145 app.betaGaugeLabel.Position = [186 117 29 22];
146 app.betaGaugeLabel.Text = 'beta';
147
148 % Create betaGauge
149 app.betaGauge = uigauge(app.UIFigure, 'ninetydegree');
150 app.betaGauge.Limits = [0 90];

```



```

151     app.betaGauge.Position = [154 154 90 90];
152
153     % Create alphaGaugeLabel
154     app.alphaGaugeLabel = uilabel(app.UIFigure);
155     app.alphaGaugeLabel.HorizontalAlignment = 'center';
156     app.alphaGaugeLabel.Position = [62 117 35 22];
157     app.alphaGaugeLabel.Text = 'alpha';
158
159     % Create alphaGauge
160     app.alphaGauge = uigauge(app.UIFigure, 'ninetydegree');
161     app.alphaGauge.Limits = [0 90];
162     app.alphaGauge.Orientation = 'northeast';
163     app.alphaGauge.ScaleDirection = 'counterclockwise';
164     app.alphaGauge.Position = [34 154 90 90];
165
166     % Create ProgrammetocalculateforcesLabel
167     app.ProgrammetocalculateforcesLabel = uilabel(app.UIFigure);
168     app.ProgrammetocalculateforcesLabel.HorizontalAlignment = '
        right';
169     app.ProgrammetocalculateforcesLabel.FontSize = 20;
170     app.ProgrammetocalculateforcesLabel.FontWeight = 'bold';
171     app.ProgrammetocalculateforcesLabel.Position = [441 534 306
        56];
172     app.ProgrammetocalculateforcesLabel.Text = 'Programme to
        calculate forces';
173
174     % Create UITable2
175     app.UITable2 = uitable(app.UIFigure);
176     app.UITable2.ColumnName = {'L12'; 'L13'; 'L23'; 'R2x'; 'R2y';
        'R3y'};
177     app.UITable2.RowName = {};
178     app.UITable2.ColumnEditable = true;
179     app.UITable2.Position = [272 264 479 193];
180
181     % Create Label
182     app.Label = uilabel(app.UIFigure);
183     app.Label.HorizontalAlignment = 'right';
184     app.Label.Position = [202 479 545 56];
185     app.Label.Text = {'Please input the force F, the lengths of
        the members L12, L23 and L13.'; 'The programme will then
        calculate the values of alpha and beta and display them to
        you.'; 'If you would like to change the coeffcient matrix
        , look to the table on the right and adjust as you like.';
        'Click "Find Forces" to calculate the values of the
        internal bar forces and the reaction forces.'};
186
187     % Show the figure after all components are created
188     app.UIFigure.Visible = 'on';
189
190     end
191
192     % App creation and deletion
193     methods (Access = public)

```

```

194
195 % Construct app
196 function app = q2eApp-exported
197
198 % Create UIFigure and components
199 createComponents(app)
200
201 % Register the app with App Designer
202 registerApp(app, app.UIFigure)
203
204 % Execute the startup function
205 runStartupFcn(app, @startupFcn)
206
207 if nargin == 0
208     clear app
209 end
210 end
211
212 % Code that executes before app deletion
213 function delete(app)
214
215     % Delete UIFigure when app is deleted
216     delete(app.UIFigure)
217 end
218 end
219 end

```

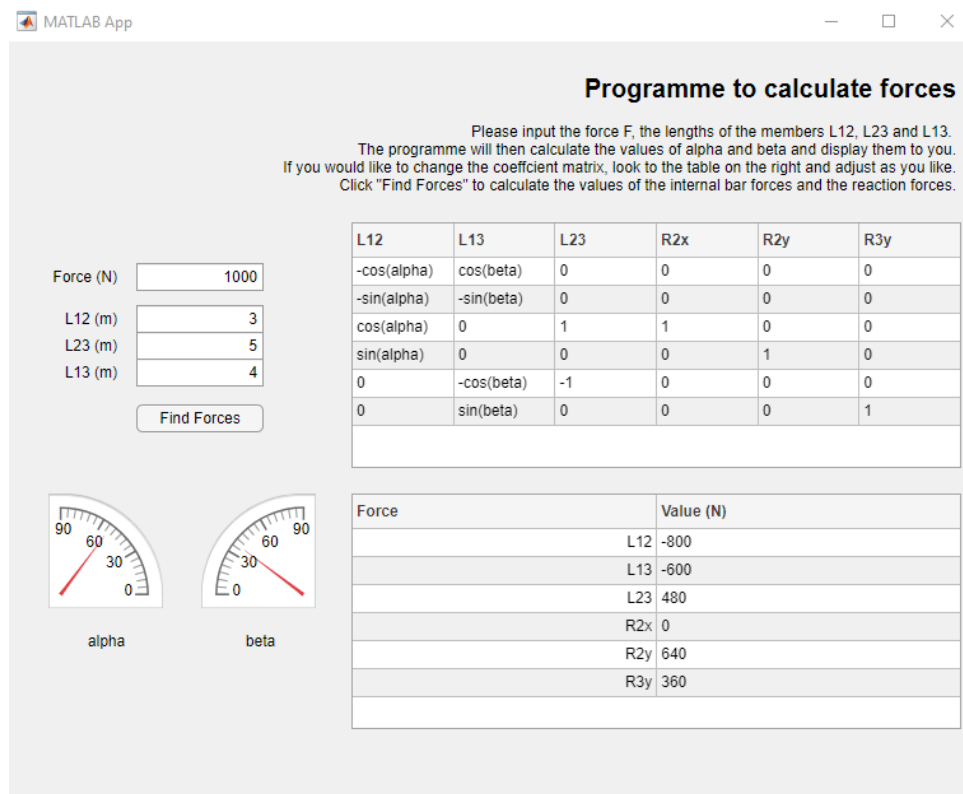


Figure 8: Screenshot from Matlab App, showcasing GUI, input and output parameters.

2.6 f

Code was written to generate a table of data:

```
1  clc
2  clear
3  close all
4  %forces
5  F = [1000 3000 4500];
6
7  %lengths of members
8  L12 = [6 8 5];
9  L23 = [10 12 8];
10 L13 = [9 7 4];
11
12 %initilise matrix
13 sol = zeros(9,10);
14
15 %initilise counter
16 counter = 0;
17
18 %nested loops, iterates between F and then between A1, A2, A3 and stores
    in sol matrix
19 for i = 1:3
20     for j = 1:3
21         %calculate alpha and beta
22         alpha = acos((L12(j)^2 + L23(j)^2 - L13(j)^2)/(2*L12(j)*L23(j)));
23         beta = acos((L13(j)^2 + L23(j)^2 - L12(j)^2)/(2*L13(j)*L23(j)));
24
25         %calculate A and B matrices
26         A = [-cos(alpha) cos(beta) 0 0 0 0;
27             -sin(alpha) -sin(beta) 0 0 0 0;
28             cos(alpha) 0 1 1 0 0;
29             sin(alpha) 0 0 0 1 0;
30             0 -cos(beta) -1 0 0 0;
31             0 sin(beta) 0 0 0 1];
32         B = [0; F(i); 0; 0; 0; 0];
33
34         %generate result
35         temp = (A\B)';
36
37         %increment counter
38         counter = counter + 1;
39
40         %store result
41         sol(counter, 5:10) = temp;
42     end
43 end
44
45 %table formatting
46 sol(:,1) = repelem(F',3,1);
47 sol(:,2) = repmat(L12',3,1);
48 sol(:,3) = repmat(L13',3,1);
49 sol(:,4) = repmat(L23',3,1);
```

```

50
51 %swap L13 and L23 columns
52 v = sol(:, 7);
53 sol(:, 7) = sol(:, 6);
54 sol(:, 6) = v;
55
56 %clean up values
57 for i=1: numel(sol)
58     if sol(i) < 0.01 && sol(i) > -0.01
59         sol(i) = 0;
60     end
61 end
62
63 %table generation
64 T = array2table(sol);
65 T.Properties.VariableNames = { 'Force', 'L12', 'L23', 'L13', 'N12', 'N13',
    'N23', 'R2x', 'R2y', 'R3y' };

```

	1 Force	2 L12	3 L23	4 L13	5 N12	6 N13	7 N23	8 R2x	9 R2y	10 R3y
1	1000	6	9	10	-815.7246	373.8738	-464.1192	0	725	275.0000
2	1000	8	7	12	-799.0757	661.7346	-861.7938	0	447.9167	552.0833
3	1000	5	4	8	-1.0504e+03	958.4751	-1.1153e+03	0	429.6875	570.3125
4	3000	6	9	10	-2.4472e+03	1.1216e+03	-1.3924e+03	0	2175	825.0000
5	3000	8	7	12	-2.3972e+03	1.9852e+03	-2.5854e+03	0	1.3438e+03	1.6562e+03
6	3000	5	4	8	-3.1512e+03	2.8754e+03	-3.3459e+03	0	1.2891e+03	1.7109e+03
7	4500	6	9	10	-3.6708e+03	1.6824e+03	-2.0885e+03	0	3.2625e+03	1.2375e+03
8	4500	8	7	12	-3.5958e+03	2.9778e+03	-3.8781e+03	0	2.0156e+03	2.4844e+03
9	4500	5	4	8	-4.7267e+03	4.3131e+03	-5.0189e+03	0	1.9336e+03	2.5664e+03

Table 1: Table of data generated from MATALB, showing forces in three configuration with three different loads.

Force (N)	L12 (m)	L13	L23	N13 (N)	N23	N13	R2x	R2y	R3y
1000	6	9	10	-815.7	373.9	-464.1	0	725.0	275.0
1000	8	7	12	-799.1	661.7	-861.8	0	447.9	552.1
1000	5	4	8	-1050.4	958.5	-1115.3	0	429.7	570.3
3000	6	9	10	-2447.2	1121.6	-1392.4	0	2175.0	825.0
3000	8	7	12	-2397.2	1985.2	-2585.4	0	1343.8	1656.3
3000	5	4	8	-3151.2	2875.4	-3346.0	0	1289.1	1710.9
4500	6	9	10	-3670.8	1682.4	-2088.5	0	3262.5	1237.5
4500	8	7	12	-3595.8	2977.8	-3878.1	0	2015.6	2484.4
4500	5	4	8	-4726.7	4313.1	-5018.9	0	1933.6	2566.4

Table 2: Table to show values of internal bar forces and reaction forces.

3 Question 3

```

1  clc
2  clear
3  close all
4
5  %import data
6  T = readmatrix('q3Data.xlsx');
7
8  %auto calcs
9  meanA = mean(T(:,4)); %mean
10 meanB = mean(T(:,8));
11
12 stdA = std(T(:,4)); %standard deviation
13 stdB = std(T(:,8));
14
15 %manual calcs
16 muA = sum(T(:,4))/numel(T(:,4)); %find mean
17 muB = sum(T(:,8))/numel(T(:,8));
18
19 meanDiffA = T(:,4) - muA; %find difference between value and mean
20 meanDiffB = T(:,8) - muB;
21
22 squareSumDiffA = sum(meanDiffA.^2); %square and sum
23 squareSumDiffB = sum(meanDiffB.^2);
24
25 stanDevA = sqrt(squareSumDiffA/(numel(T(:,4))-1)); %square root and
    divide by n-1 (sample)
26 stanDevB = sqrt(squareSumDiffB/(numel(T(:,8))-1));
27
28 %check
29 if meanA == muA && meanB == muB && stdA == stanDevA && stdB == stanDevB
30     disp('correct')
31 else
32     disp('try again')
33 end

```

	A	B
Mean	50991.90	50328.27
Standard Deviation	53.77	863.99

Table 3: Table to show values of means and standard deviations of weekly output of vaccines for manufacturer A and B.

3.1 b

Hypothesis test for efficiency, utilising a two tailed z-test as sample size is large. Null hypothesis: efficiencies are not significantly different, alternate hypothesis: efficiencies are significantly different.

$$H_0 : P_a - P_b = 0 \quad (3.1)$$

$$H_1 : P_a - P_b \neq 0 \quad (3.2)$$

Hypothesis test for factory output, utilising a two tailed t-test as sample size is 30. Null hypothesis: factory outputs are not significantly different, alternate hypothesis: factory outputs are significantly

different.

$$H_0 : \mu_a - \mu_b = 0 \quad (3.3)$$

$$H_1 : \mu_a - \mu_b \neq 0 \quad (3.4)$$

3.2 c

4 Question 4

4.1 a

4.1.1 i

Let X be number of trials until the first head appears. Coin is unbiased, thus $X \sim Geo\left(\frac{1}{2}\right)$.

$$P(X = 1) = \frac{1}{2} \quad (4.1)$$

$$P(X = 2) = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \quad (4.2)$$

$$P(X = 3) = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{8} \quad (4.3)$$

$$P(X = n) = \frac{1}{2^n} \quad (4.4)$$

4.1.2 ii

$$\sum_{n=1}^{\infty} P(X = n) = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots + \frac{1}{2^n} \quad (4.5)$$

Geometric series, hence:

$$a = \frac{1}{2}, r = \frac{1}{2} \quad (4.6)$$

$$\sum_{n=1}^{\infty} P(X = n) = \frac{a}{1 - r} = \frac{\frac{1}{2}}{1 - \frac{1}{2}} = 1 \quad (4.7)$$

4.2 b

4.2.1 i

$$f(x) = \begin{cases} \alpha(1-x^2) & -1 \leq x \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.8)$$

$$F(x) = 1 = \int_{-1}^1 (\alpha - \alpha x^2) dx \quad (4.9)$$

$$= \left[\alpha x - \frac{\alpha x^3}{3} \right]_{-1}^1 \quad (4.10)$$

$$= \left(\alpha - \frac{\alpha}{3} + \alpha - \frac{\alpha}{3} \right) \quad (4.11)$$

$$\frac{4\alpha}{3} = 1 \quad (4.12)$$

$$\alpha = \frac{3}{4} \quad (4.13)$$

$$P\left(-\frac{1}{2} \leq x \leq \frac{1}{2}\right)$$

$$P\left(-\frac{1}{2} \leq x \leq \frac{1}{2}\right) = \int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\frac{3}{4} - \frac{3x^2}{4} \right) dx \quad (4.14)$$

$$= \left[\frac{3x}{4} - \frac{x^3}{4} \right]_{-\frac{1}{2}}^{\frac{1}{2}} \quad (4.15)$$

$$= \left(\frac{3}{8} - \frac{1}{32} + \frac{3}{8} - \frac{1}{32} \right) \quad (4.16)$$

$$= \frac{11}{16} \quad (4.17)$$

$$= 68.75\% \quad (4.18)$$

$$P\left(\frac{1}{4} \leq x \leq 2\right)$$

$$P\left(\frac{1}{4} \leq x \leq 2\right) = \int_{\frac{1}{4}}^1 \left(\frac{3}{4} - \frac{3x^2}{4} \right) dx \quad (4.19)$$

$$= \left[\frac{3x}{4} - \frac{x^3}{4} \right]_{\frac{1}{4}}^1 \quad (4.20)$$

$$= \left(\frac{3}{4} - \frac{1}{4} - \frac{3}{16} + \frac{1}{256} \right) \quad (4.21)$$

$$= \frac{81}{256} \quad (4.22)$$

$$= 31.64\% \quad (4.23)$$

4.2.2 ii

$$P(X \leq x) = 0.95$$

$$P(X \leq x) = 0.95 = \int_{-1}^x \left(\frac{3}{4} - \frac{3x^2}{4} \right) dx \quad (4.24)$$

$$\left[\frac{3x}{4} - \frac{x^3}{4} \right]_{-1}^x = 0.95 \quad (4.25)$$

$$\left(\frac{3x}{4} - \frac{x^3}{4} + \frac{3}{4} - \frac{1}{4} \right) = 0.95 \quad (4.26)$$

$$\frac{x^3}{4} - \frac{3x}{4} + \frac{9}{20} = 0 \quad (4.27)$$

Solving via calculator:

$$x_1 \neq 1.2481 \quad (4.28)$$

$$x_2 \neq -1.9777 \quad (4.29)$$

$$x_3 = 0.7293 \quad (4.30)$$

4.3 c

4.3.1 i

$$f(x, y) = \begin{cases} \alpha e^{-0.1(x+y)} & x > 0 \text{ \& } y > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4.31)$$

$$f_1(x) = \lim_{t \rightarrow \infty} \int_0^t \left(\alpha e^{-0.1(x+y)} \right) dy \quad (4.32)$$

$$= \lim_{t \rightarrow \infty} \left[-\frac{\alpha}{0.1} e^{-0.1(x+y)} \right]_0^t \quad (4.33)$$

$$= \lim_{t \rightarrow \infty} \left[-\frac{\alpha}{0.1} e^{-0.1t} + \frac{\alpha}{0.1} e^{-0.1x} \right] \quad (4.34)$$

$$= \left[0 + \frac{\alpha}{0.1} e^{-0.1x} \right] \quad (4.35)$$

$$f_1(x) = \frac{\alpha}{0.1} e^{-0.1x} \quad (4.36)$$

$$f_2(y) = \lim_{t \rightarrow \infty} \int_0^t \left(\alpha e^{-0.1(x+y)} \right) dx \quad (4.37)$$

$$= \lim_{t \rightarrow \infty} \left[-\frac{\alpha}{0.1} e^{-0.1(x+y)} \right]_0^t \quad (4.38)$$

$$= \lim_{t \rightarrow \infty} \left[-\frac{\alpha}{0.1} e^{-0.1t} + \frac{\alpha}{0.1} e^{-0.1y} \right] \quad (4.39)$$

$$= \left[0 + \frac{\alpha}{0.1} e^{-0.1y} \right] \quad (4.40)$$

$$f_2(y) = \frac{\alpha}{0.1} e^{-0.1y} \quad (4.41)$$

$f(x, y) \neq f_1(x)f_2(y)$, condition not fulfilled for independence.

4.3.2 ii

$$F(x, y) = 1 = \lim_{t \rightarrow \infty} \int_{y=0}^t \int_{x=0}^t \left(\alpha e^{-0.1(x+y)} \right) dx dy \quad (4.42)$$

$$= \lim_{t \rightarrow \infty} \int_{y=0}^t \left[-\frac{\alpha}{0.1} e^{-0.1(x+y)} \right]_{x=0}^t dy \quad (4.43)$$

$$= \lim_{t \rightarrow \infty} \int_{y=0}^t \left(\frac{\alpha}{0.1} e^{-0.1y} \right) dy \quad (4.44)$$

$$= \lim_{t \rightarrow \infty} \left[-\frac{\alpha}{0.01} e^{-0.1y} \right]_{y=0}^t \quad (4.45)$$

$$= 0 + \frac{\alpha}{0.01} e^0 \quad (4.46)$$

$$\frac{\alpha}{0.01} = 1 \quad (4.47)$$

$$\alpha = 0.01 \quad (4.48)$$

4.3.3 iii

$$P(X \geq 10)$$

$$P(X \geq 10) = \lim_{t \rightarrow \infty} \int_{x=10}^t \int_{y=0}^t \left(0.01 e^{-0.1(x+y)} \right) dy dx \quad (4.49)$$

$$= \lim_{t \rightarrow \infty} \int_{x=10}^t \left[-0.1 e^{-0.1(x+y)} \right]_{y=0}^t dx \quad (4.50)$$

$$= \lim_{t \rightarrow \infty} \int_{x=10}^t \left[0 + 0.1 e^{-0.1x} \right] dx \quad (4.51)$$

$$= \lim_{t \rightarrow \infty} \left[-e^{-0.1x} \right]_{10}^t \quad (4.52)$$

$$= \left[0 + e^{-1} \right] \quad (4.53)$$

$$= 36.79\% \quad (4.54)$$

4.3.4 iv

$$P(Y < X)$$

$$P(Y < X) = \lim_{t \rightarrow \infty} \int_{x=0}^t \int_{y=0}^x \left(0.01 e^{-0.1(x+y)} \right) dy dx \quad (4.55)$$

$$= \lim_{t \rightarrow \infty} \int_{x=0}^t \left[-0.1 e^{-0.1(x+y)} \right]_{y=0}^x dx \quad (4.56)$$

$$= \lim_{t \rightarrow \infty} \int_{x=0}^t \left(-0.1 e^{-0.2x} + 0.1 e^{-0.1x} \right) dx \quad (4.57)$$

$$= \lim_{t \rightarrow \infty} \left[0.5 e^{-0.2x} - e^{-0.1x} \right]_0^t \quad (4.58)$$

$$= [0 - 0 - 0.5 + 1] \quad (4.59)$$

$$= 50\% \quad (4.60)$$