**ENEL102, fall term 2018**

**Assignment 5**

**Function Integration and differential equations**

**(sections 9.3 and 9.4 )**

**Due date Nov 19**

This assignment is based on the material in section 9.3 and 9.4. Suggest you read through these sections first before attempting the assignment questions. As usual, cut&paste the questions into a word document and fill in the answers. Then submit your Word document on D2L.

………………………………………………………………………………………………………………………………..

**Q1** Numerically integrate the function  from the lower limit of x=0 to the upper limit of x=2 based on using integral().

**(Matlab input)**

f = @(x) (sin(x))./(x.^4 +1)

q = integral(f,0,2)

**(Matlab Response)**

q =

0.5554

………………………………………………………………………………………………………………………………..

**Q2** Numerically integrate the function  from the lower limit of x=-10 to the upper limit of x=10 based on using integral(). A conservative approach is to split your integration into three regions given as



where  is a small positive value. Then integrate over the three regions separately and then add up the results.

Note that in this case, the integral() routine does find the point where f(x) is indeterminate and handles it correctly. Show this by integrating directly from x=-10 to x=10. Sometimes this detection fails and Matlab would give the wrong answer. Hence if you know where the indeterminate values of x are then it is best to enter this information by partitioning up the integral.

**(Matlab input)**

func = @(x) (sin(x).^2)./(x.^2)

region1 = integral(func,-10,-0.005)

region2 = integral(func,-0.005,0.005);

region3 = integral(func,0.005, 10)

sum = region1+region2+region3

Integrating directly from x = -10 to x = 10

q = integral(func,-10,10)

**(Matlab Response)**

sum =

3.0373

q =

3.0373

……………………………………………………………………………………………………………………………

**Q3** Consider the curve parameterized by the equations

x(t) = sin(2t), y(t) = cos(t), z(t) = t,

and assume a range in the parameter t of . Create a three-dimensional plot of this curve using plot3().

**(Matlab input)**

t = linspace(0,10)

x = sin(2\*t)

y = cos(t)

z = t

plot3(x,y,z)

hold on

grid on

xlabel('x');ylabel('y');zlabel('z')

**(Matlab Response)**

****

……………………………………………………………………………………………………………………………

**Q4** In this question compute the arc length of the curve in **Q3**. Recall from calculus that the arc length is given as



**(Matlab input)**

f = @(t) sqrt(4\*cos(2\*t).^2 + sin(t).^2 + 1);

L = integral(f,0,10)

**(Matlab Response)**

L =

18.3257

……………………………………………………………………………………………………………………………

**Q5** A covering surface has the space described as follows:





where the units are in meters. Generate a mesh plot of this surface and label the axis.

**(Matlab input)**

x = -2:0.1:2;

y = -4:0.1:4;

[X, Y] = meshgrid(x,y);

Z = exp(-X.^2 +0.1\*X.\*Y -Y.^2);

mesh(X,Y,Z)

xlabel('x');ylabel('y');zlabel('z')

**(Matlab Response)**



**Q6** Determine the volume of the space that is bounded by the cover surface and the z=0 plane for the extent of



**(Matlab input)**

func = @(x,y) exp(-x.^2 +0.1\*x.\*y -y.^2);

q = integral2(func,-2,2,-4,4)

**(Matlab Response)**

q =

3.1306

**Q7** Consider a three dimensional scalar function in a Cartesian space given by the value of



Integrate the three dimensional scalar function within a unit sphere that is centered at the origin. The solution to this integration involves the careful consideration of the integration boundaries as set by the sphere. Use integral3() and note that you can write the integration limits as functions of x, y and z.

**(Matlab input)**

g = @(x,y,z) (x+1).\*(y.^2).\*(cos(z));

q = integral3(g,0,1,0,2\*pi,0,pi)

**(Matlab Response)**

q =

1.5596e-14

………………………………………………………………………………………………………………………………..

**Q8** Consider the simplest differential equation with initial condition given as



Solution to this is simply x=t. Set this up with Matlabs ode45() to verify this. Solve the DEQ for the range of  and plot the solution of x(t) based on ode45() output.

**(Matlab input)**

[t x] = ode45(@(t,x) 1, [0:0.25:1], 0)

plot(t,x)

xlabel('t'); ylabel('x')

**(Matlab Response)**

****

……………………………………………………………………………………………………………………………

**Q9** Consider the next simplest differential equation given as



Solution to this is . As before set this up with Matlabs ode45() to verify this. Solve the DEQ for the range of  and plot the solution of x(t) based on ode45() output.

**(Matlab input)**

[t x] = ode45(@(t,x) t, [0 1], 0)

plot(t,x)

xlabel('t'); ylabel('x')

**(Matlab Response)**

****

……………………………………………………………………………………………………………………………

**Q10** Consider another linear differential equation given as



Solution to this is



As before set this up with Matlabs ode45() to verify this. Solve the DEQ for the range of  and plot the solution of x(t) based on ode45() output.

**(Matlab input)**

[t x] = ode45(@(t,x) t, [0 1], 1)

plot(t,x); xlabel('t'); ylabel('x')

**(Matlab Response)**

****

……………………………………………………………………………………………………………………………

**Q11** Now consider a series circuit consisting of a voltage source, v(t), that switches from 0 to 1 volts at t=0 which is connected in series with an inductor L and a resistor R. Assume that the current is to be determined. The DEQ for this circuit is given as



Assume that L=R=1. Solve for the current for the range in time from 0<t<10 and plot the solution of x(t) based on ode45() output.

**(Matlab input)**

[t i] = ode45(@(i,x) i, [0:0.5:10], 1)

plot(t,i); xlabel('t'); ylabel('i')

**(Matlab Response)**



**Q12** Next consider a mass of M=1Kg that is suspended on a spring with an stiffness constant of k=100n/m. The weight is initially held such that the tension through the spring is zero. Then at time t=0 the weight is released. Find the displacement of the weight over the time interval of 0<t<10. Note that the DEQ for this example is given as



To solve a second order DEQ you have to reduce it to a first order format. Hence you set up a state space formulation as



Then input this first order state equation into ode45(). Solve for the state variables and plot these over the range of 0<t<10. Superimpose the two plots as this is easier to use for comparison than separate sub plots.

**(Matlab input)**

function second\_order

t = 0:0.01:10;

[t, x] = ode45(@rhs, t, [0 0]);

plot(t, x);

xlabel('t'); ylabel('x');

function dx = rhs(~,x)

dx1 = x(2);

dx2 = 9.8 - 100\*x(1);

dx = [dx1;dx2];

end

end

**(Matlab Response)**

****