Instructions:

- Please comment your code
- FOUR significant figures is sufficient.

- MATLAB Script containing your commands ideally formatted in cell-mode; and
- an HTML file via the PUBLISH command; and
- any function files and image files.

PART A: Functions of Two Parameters

This part is worth 20 marks. You must answer all questions for this part.

QUESTION 1 **[12 Total Marks]**

Create a function to solve a generic 3 x 3 system of equations with a column vector representing 3 different constants.

For example:

$$\begin{bmatrix} a1 & a4 & a7 \\ a2 & a5 & a8 \\ a3 & a6 & a9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix}$$
$$Symbolically\ AX = E$$

function [X] = simul(inputCoefficients, rhsConstants)

a) Find the determinant of A to establish if the input coefficient matrix is invertible. If the determinant is zero then the matrix is not invertible. [2 Marks]

b) Use an IF statement based on the result of a) to decide whether or not to proceed with a solution based on ONE of four methods. [2 Marks]

c) Using the AX = E equation and ONE of four of the following methods, solve the 3 x 3 system: [4 marks]

   i. using Gauss-Jordan reduction procedure (use RREF)

   ii. using $A^{-1}B$ % find A^{-1}, the inverse matrix, using INV or A^-1

   iii. using A \ B % left hand Gaussian Elimination division

   iv. using [B'/A']' % right hand division

d) Create code in your function to return a 3 x 1 vector called X. [4 Marks]

QUESTION 2 **[8 Total Marks]**

Solve the following problem with the function created in question one:

$$\begin{bmatrix} 0.7922 & 0.0357 & 0.6787 \\ 0.9595 & 0.8491 & 0.7577 \\ 0.6557 & 0.9340 & 0.7431 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0.0344 \\ 0.4387 \\ 0.3816 \end{bmatrix}$$

A X = E

a) using your function from a) find the solution (matrix X which is a 3 x 1 matrix).

[4 Marks]

b) Run your function at command-line level and check the solution matrix, X, with AX = E. [2 Marks]

c) Evaluation of Solution: [2 Marks]

Please use an inequality (a Boolean expression) to evaluate the accuracy of your solution. For example,

```
>> norm(A*X - E) < eps
```

If the Boolean expression (see box above) returns one (TRUE) then NORM(…) is less than EPS and you get 2 marks otherwise if it returns zero (FALSE) then NORM(…) is greater than or equal to EPS and you get 0 marks.

Explanation of the NORM and EPS Functions

When passed a difference vector (A*X – E), NORM gives the 2-norm or Euclidean Distance (or length) from the origin to the difference vector.

```
>> M = [1 2 3];

>> norm(M,2)

ans =  3.7417

>> norm(M)

ans =  3.7417
```

EPS (Epsilon), a MATLAB built-in function, with no arguments, is the distance from 1.0 to the next larger double precision number, that is EPS with no arguments returns 2^(-52).

```
>> eps

ans =  2.2204e-16
```

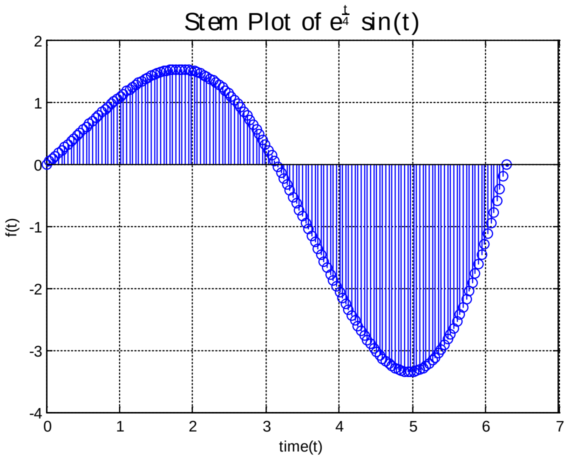
PART B: Graphing in MATLAB

This part is worth 40 marks. You must answer all questions for this part.

Complete the MATLAB code wherever required for the following 2D MATLAB graph functions:

QUESTION 1 **[8 Total Marks]**

STEM graph displaying the function $f(t) = e^{\frac{t}{4}} \sin t \;\; for\ the\ interval\ 0 \leq t \leq 2\pi$

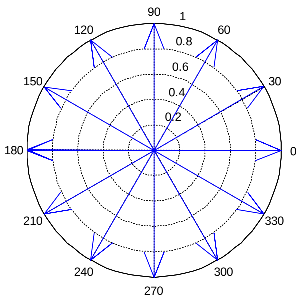



```
t = linspace(0,2*pi,150);

exponent = …;

a=exp(exponent*…).*(sin(t));

stem(…, …)

syms t

f = exp(exponent*…).*(sin(t));
grid on

xlabel '…'
ylabel('…');

title_handle = title(['Stem Plot of ' '$' latex(f) '$'],...
    'interpreter','latex');

set(…,'fontsize',16)
```

QUESTION 2 **[6 Total Marks]**

COMPASS graph displaying the function

$$z = \cos\theta + j\sin\theta \ for\ the\ interval - \pi \le \theta \le \pi$$




```
theta=-pi:pi/…:pi

rx=cos(…)

ry=sin(…)

z= … + …

compass(…)
```

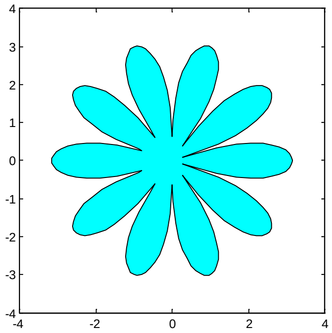
QUESTION 3 **[10 Total Marks]**

a) FILL graph displaying the function $r^2 = 10\cos\mathbf{5}t \ for\ the\ interval\ 0 \le t \le 2\pi$

[9 Marks]

HINT: make r the subject and use the trigonometric identities to find x and y:

$$x = \mathrm{r}\cos t, y = r\sin t$$

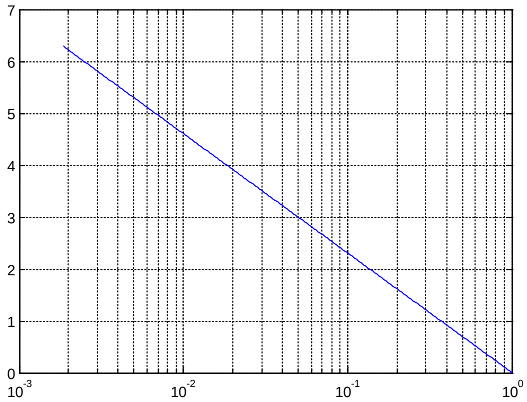



```
t = linspace(0,2*pi,…);

r=sqrt(…(…));

x=…

y=…
fill(x,y,'c')
axis('square')
```

b) Try altering the function, r^2, to increase the number of “petals” from 10 to 20. Save the MATLAB figure as a PNG file. [1 Mark]

QUESTION 4 **[3 Total Marks]**

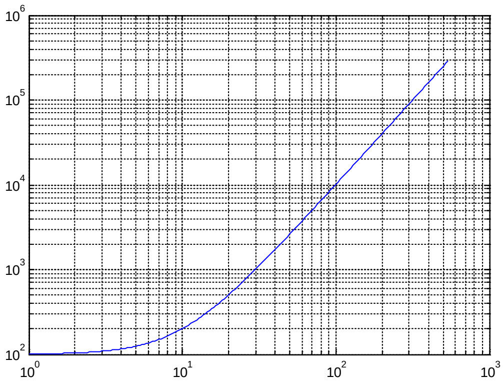
SEMILOGX graph displaying the function $x = e^{-t}, y = t, for\ the\ interval\ 0 \leq\ t \leq 2\pi$
HINT: fill out the ellipses with some MATLAB Code.




```
t = linspace(0,2*pi,200);

x = exp(-t);

y1 = …;

semilogx(…,…), grid on
```

QUESTION 5 **[3 Total Marks]**

LOGLOG graph displaying the function $x = e^{t}, y = 100 + e^{2*t},\ for\ the\ interval\ 0 \leq\ t \leq 2\pi$



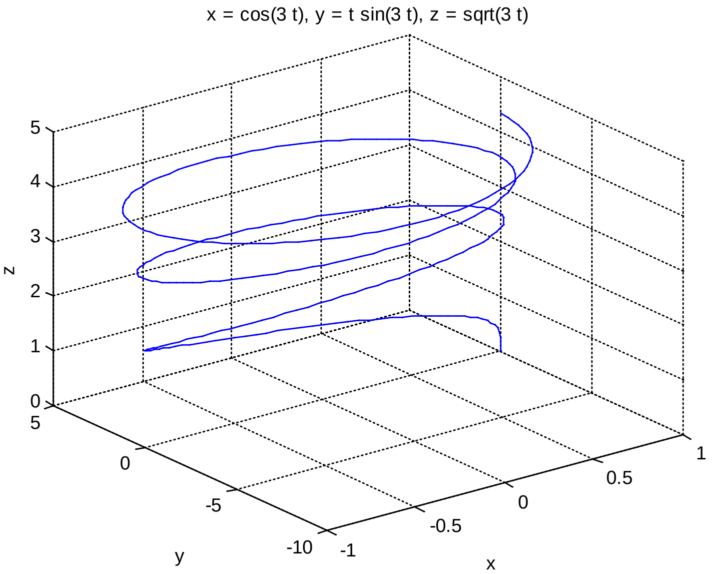

```
t = linspace(0,2*pi,200);

x = exp(t);

y = 100+exp(…);

loglog(…,…), grid
```

Complete the MATLAB code wherever required for the following special MATLAB graph functions:

QUESTION 6 **[5 Total Marks]**

EZPLOT graph for the following functions:

$$x = \cos(3t)\,, \qquad y = t\sin(3t), \qquad z = \sqrt{3t}$$



a) `ezplot3(’…’,’…’,’…’)` [3 Marks]

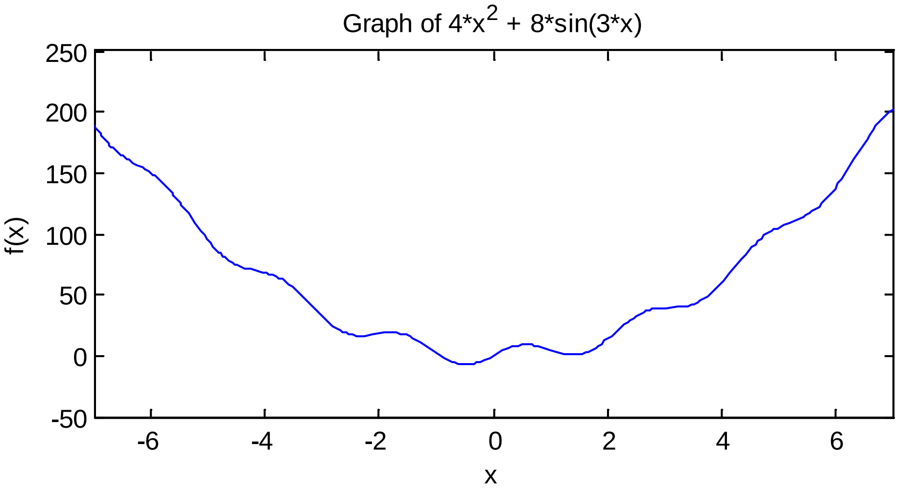
Increase the number of loops from 3 to 6:

b) `ezplot3(’…’,’…’,’…’)` [2 Marks]

QUESTION 7 **[5 Total Marks]**

FPLOT graph to display the following function

$$y = 4x^2 + 8\sin(3x)\, for\ the\ interval - 7\ to\ 7$$




```
fplot('…',[...,...])

xlabel x

ylabel …

title '…'
```

PART C: Flight Path of a Model Rocket [30 TOTAL MARKS]

Scenario

The flight path of a model rocket, of mass **0.05** kilograms, can be modelled in MATLAB using simple equations and graphs. During the first 0.25 seconds (s) the rocket is propelled upward by the rocket engine with a force of **25 N**. However, the rocket runs out of fuel before reaching its peak hence, the engine stops.

The rocket then flies up carried by its own momentum while slowing down under the gravity force (g = **9.81metres/s/s)** at a constant deceleration.

Once it reaches the peak of its flight path the rocket starts to fall down. At a downward velocity of 20 m/s the parachute opens instantly so the velocity stays constant (that is 20 m/s) with an acceleration of zero until it hits the ground.

The rocket is assumed to be a particle that moves along a straight line in the vertical plane. For motion with constant acceleration (a) along a straight line, the velocity and position as a function of time (t) are given by the equations:

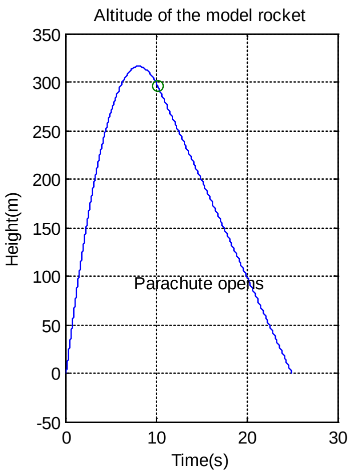
1. $v(t) = v_0 + at$
2. $h(t) = s_0 + v_0 t + \frac{1}{2}at^2$

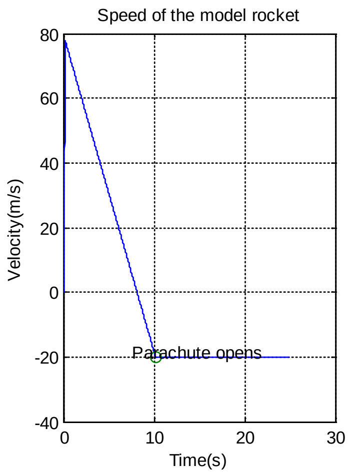
$$\textit{where } v_0 \textit{ and } s_0 \textit{ are the initial velocity and position (height) respectively}$$

MATLAB Task

Write a MATLAB script to calculate the height (metres) and the speed (metres/s) of the rocket over flight-time.

Plot these results against time using two graphs.
NOTE: the speed metric should include the direction (+up / -down) as well as metres/s information.

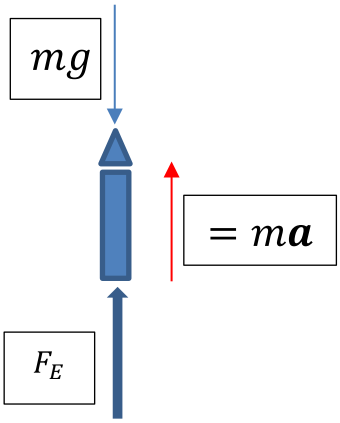




Instructions to create this MATLAB script:

- Flight path of the rocket is divided into ***three*** **segments (see below for detailed description)**
- Each segment is calculated with either WHILE or FOR-END loops
- In every iteration, of a loop, time is incremented by **Dt = 0.01**
- Separately graph the results of height (m) and velocity (m/s)
- Annotate your plots inside your script or with the Plot Editor
- The accuracy of the h(t) and v(t) results depends on the size of the time increment, Dt
- Set n to be no bigger than 50,000 to avoid infinite loops
- Make use of the MATLAB Code frame on page 15 to complete the script

Detailed Description of the Three Segments of the Rocket's Flight-time



$$+\uparrow \sum F = \cdots + (\ldots) = m\boldsymbol{a}$$

a) Segment One [10 Marks]

The first 25 seconds when the rocket engine is firing and the rocket is moving upward with constant acceleration. Determine the acceleration by referring to the mass acceleration diagram and noting the forces acting on the rocket. Then fill out the ellipses of the summative force equation with the appropriate forces, then solve it for **a**.

Next work out v(t) and h(t) from the previous equations, 1 and 2, on page 9.

$$v_0 \text{ and } s_0 \text{ both equal zero in this case}$$

In the MATLAB program this segment starts at t = 0 and loops while t < 0.25s. At t = 0.25 s time, velocity and height become t_1, v_1 and h_1.

b) Segment Two [10 Marks]

The motion of the rocket from when the rocket engine stops firing, or runs out of fuel, until the parachute opens. The rocket moves with a constant deceleration of g. Now the equations for v(t) and h(t) alter to become:

$$v(t) = v_{1-} g(t - t_1) \; and \; h(t) = h_1 + v_1(t - t_1) - \frac{1}{2}g(t - t_1)^2$$

The iteration continues until the velocity is 20 m/s but note that as we are plotting speed against time we really need to consider direction too so that this is really negative (-)20 m/s (v_2) since the rocket is falling downwards, that is, in a negative direction. At the end of this segment two, the time, velocity and height become t_2, v_2 and h_2.

c) Segment Three [10 Marks]

The motion of the rocket from when the parachute opens until the rocket hits the ground. The rocket now moves with a constant velocity of -20m/s but zero acceleration. Now the equations for v(t) and h(t) alter to become:

$$v(t) = -20 \; and \; h(t) = h_1 - v_{chute}(t - t_2)$$

The loop continues to iterate so long as height is > zero.

PART D: Einstein's Special Relativity **[10 TOTAL MARKS]**

According to special relativity, a metallic rod of initial length $L, when\ at\ rest,$ will when travelling at a velocity v through outer space, shrink by an amount $\delta\ (metres)$ according to the following formula:

$$\delta = L\left(1 - \sqrt{1 - \frac{v^2}{c^2}}\right)$$

$$Where\ c = 300 * \frac{10^6 m}{s}\ (the\ speed\ of\ light)$$

$$L = initial\ length\ of\ object\ in\ metres\ when\ at\ rest$$

$$v = velocity\ of\ object\ in\ metres/second$$

Imagine a rod with an initial length of 30 metres travels through outer space and reaches 90% of the speed of light, c. At this time how long is the rod?

a) $Define\ in\ MATLAB: c = 300 * 10^6, L = 30\ and\ v = 0.90 * c$ [1 Mark]

b) With the above formula calculate $\delta\ (metres)$. [8 Marks]

c) Using δ find the overall length of the rod in metres and hence, answer the question: "At this time how long is the rod?" [1 Mark]

PART E: Moment Magnitude Scale **[10 TOTAL MARKS]**

The Moment Magnitude Scale (MMS), M_W, measures the total energy released by an earthquake and is represented by the following formula:

$$M_W = \frac{2\log_{10} M_0}{3} - 10.7$$

$$Where\ M_0 = size\ or\ magnitude\ of\ the\ seismic\ moment\ in\ N \cdot m\ (10^7 dyne - cm)$$
$$and\ the\ subscript\ W = mechanical\ work\ accomplished$$

In this question you need to determine how many times more energy was released from the Great Chilean earthquake on 22 May 1960, which was the largest earthquake ever recorded in the world with a 9.5 rating on the MMS and the famous Canterbury earthquake which occurred in NZ on 22 February 2011 with a MMS rating of 6.3.

a) $Define$ in MATLAB : $M_{W_{Canterbury}} = 6.3\ and\ M_{W_{Chile}} = 9.5$ using variable names of your own choice for the two quakes.

[1 Mark]

b) $Make\ M_0$ the subject of the formula above and write a simple script or function to calculate the energy released from one earthquake. Use a reasonable value for $M_W\ to\ test\ your\ script$, for example, a value between 3 and 11.

If you write a function then make M_W the input parameter and M_0 the return variable in your function definition line. Then call your function at command-line level to test it with a value for M_W.

[8 Marks]

c) Using your script or function from sub-part (b) work out the ratio of the energy released from the two earthquakes from part (a) and hence, answer the question: “How many times more energy was released from the Great Chilean earthquake than the famous Canterbury earthquake?”

[1 Mark]

PART F: Damped Harmonic Motion **[10 TOTAL MARKS]**

Plotting the Amplitude of motion of Damped Harmonic Motion of a Spring-weighted Object

A mass hung below a wall-mounted spring is released and then drops before rising again in an oscillating or repetitive manner at an ordinary frequency, $\boldsymbol{f}$ $(\boldsymbol{Hz})$. The distance or gap between the drops and rises reduces with time with a constant of proportionality of $\boldsymbol{k}$. Therefore, the amplitude reduces reducing from the maximum amplitude $\boldsymbol{A}$ at $\boldsymbol{t_0}$ to zero at time $\boldsymbol{t}$. This process is called damped harmonic motion, an oscillatory or sinusoidal motion that diminishes over time. Note that small values of k cause a longer decay over a given time and larger values cause a shorter decay over a smaller time.

For damped harmonic motion the amplitude of motion is given by:

$$y = Ae^{-kt}\,sin(\omega t)$$

$$where\ A\ is\ the\ maximum\ amplitude = 3$$
$$e^x\ is\ the\ exponential\ function$$
$$k\ is\ a\ constant\ of\ proportionality = 0.4$$
$$t = time\ in\ seconds\ (0..15)$$
$$sin(\ldots)\ is\ the\ \ trigonometric\ sine\ function\ with\ argument\ in\ radians$$
$$f\ is\ the\ ordinary\ frequency = 2\ Hz$$
$$\omega\ is\ the\ angular\ frequency\ (\frac{rad}{s}) = 2\pi f$$
$$\pi = 3.142$$

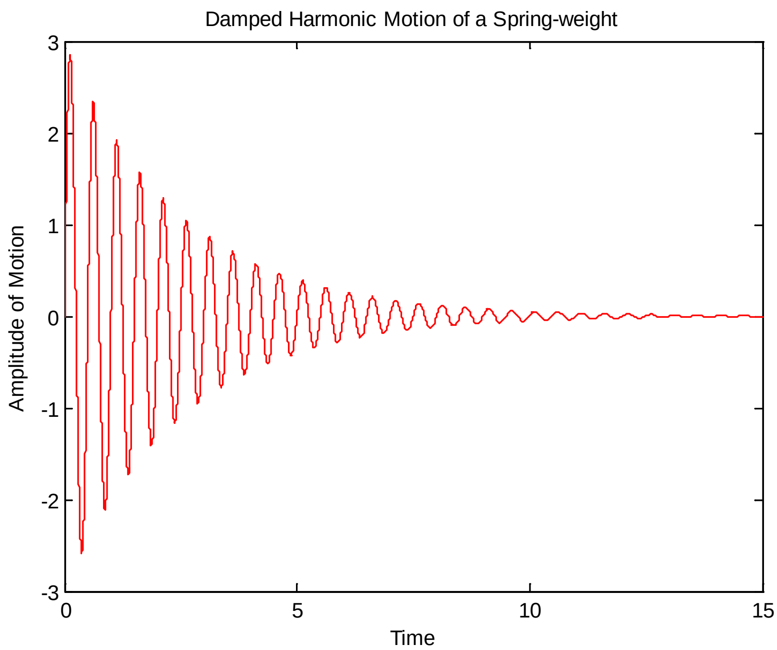


Figure 1: Damped Harmonic Motion

Using the above formula for amplitude of motion create a plot similar to Figure 1. Make sure that you label your graph and colour the plot in red using a solid line.

APPENDIX

MATLAB CODE FRAME

You may use this code frame to develop the program. You are free to code the loops with FOR-END loops but you will need to pre-calculate the times. Alternatively, you may use a vectorised approach and use element-wise calculations instead of loops.

```
% flight of a model rocket
m=0.05; g=9.81; tEngine =0.25; Force=25; vChute = -20; Dt =0.01;
clear t v h;
n = 1;
t(n) = 0; v(n) = 0; h(n) = 0;

% segment 1 - engine is on for tEngine seconds
a1=... % acceleration
while ... < ... && n < 5000
   n=n+1; % vector index
   t(n) = ...; % increment time
   v(n) = ...; % velocity
   h(n) = ...; % height
end
v1=v(n); h1=h(n); t1=t(n);

%segment 2 - motion from when the engine stops
while ... >= ... && n < 50000
    n=n+1; % vector index
    t(n) = ...; % increment time
    v(n) = ...; % velocity
    h(n) = ...; % height
end
v2=v(n); h2=h(n); t2=t(n);

% segment 3 - parachute opens
while ... > ... && n < 50000
   n=n+1; % vector index
   t(n) = ...; % increment time
   v(n) = ...; % velocity
   h(n) = ...; % height

end

% do the graphs
subplot(1,2,1)
plot(...)
title 'Altitude of the model rocket'
text(...,...,'Parachute opens');
xlabel Time(s)
ylabel Height(m)
grid
subplot(1,2,2)
plot(...)
title 'Speed of the model rocket'
text(...,...,'Parachute opens');
xlabel Time(s)
ylabel Velocity(m/s)
grid
```