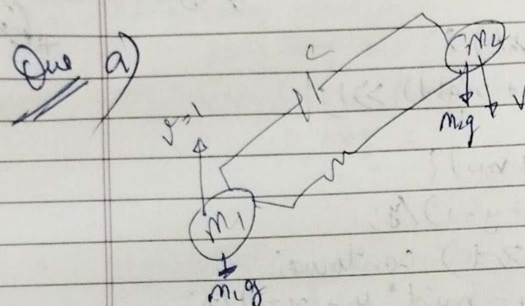
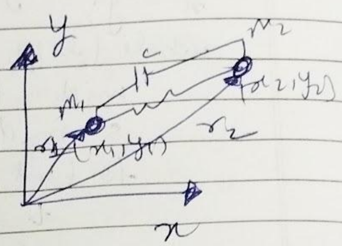
# ME3030 Assignment 2
# CO21BTECH11004

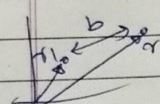
## Que 1) Part a

Name- Darpan Gaur
Roll No- CO21BTECH11004

Que a)




$\Rightarrow$ $b = r_2 - r_1$ , $\hat{b} = \dfrac{r_2 - r_1}{|r_2 - r_1|}$ direction

$|b| = |r_2 - r_1|$

Extenstion/Compression in spring $= |r_2 - r_1| - l$

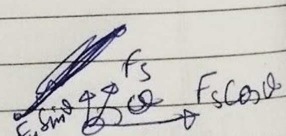
Spring Force $\Rightarrow k(|r_2 - r_1| - l)\hat{b}$

Damping Force $\Rightarrow c(\dot{r}_2 - \dot{r}_1)$

Gravitational Force $= -mg$

Eq $\Rightarrow$ $m\ddot{r} = k(|r_2 - r_1| - l)\hat{b} + c(\dot{r}_2 - \dot{r}_1) - mg$

In x-y plane
x → Take x component
y → Take y componen




$\cos\theta = \dfrac{x_2 - x_1}{\underbrace{\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}}_{|r_2 - r_1|}}$

$\sin\theta = \dfrac{y_2 - y_1}{\sqrt{(x_1-x_2)^2 + (y_1-y_2)^2}}$

## Equations

- $m_1 \begin{Bmatrix} \ddot{x}_1 \\ \ddot{y}_1 \end{Bmatrix} = \frac{k(|r_2 - r_1| - l)}{|r_2 - r_1|} \begin{Bmatrix} x_2 - x_1 \\ y_2 - y_1 \end{Bmatrix} + c \begin{Bmatrix} \dot{x}_2 - \dot{x}_1 \\ \dot{y}_2 - \dot{y}_1 \end{Bmatrix} - m_1 g \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$

- $m_2 \begin{Bmatrix} \ddot{x}_2 \\ \ddot{y}_2 \end{Bmatrix} = -m_2 g \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} - \frac{k(|r_2 - r_1| - l)}{|r_2 - r_1|} \begin{Bmatrix} x_2 - x_1 \\ y_2 - y_1 \end{Bmatrix} - c \begin{Bmatrix} \dot{x}_2 - \dot{x}_1 \\ \dot{y}_2 - \dot{y}_1 \end{Bmatrix}$

Solving Using $4^{th}$ order Runge Kutta (RK4)

Put, $k_1 = dt.f(t_n, y_n)$

$$k_2 = dt.f\left(t_n + \frac{dt}{2}, y_n + \frac{k_1}{2}\right)$$

$$k_3 = dt.f\left(t_n + \frac{dt}{2}, y_n + \frac{k_2}{2}\right)$$

$$k_4 = dt.f(t_n + dt, y_n + k_3)$$

$$y_{n+1} = y_n + \left(\frac{k_1 + 2k_2 + 2k_3 + k_4}{6}\right)$$

In this question, there are eight variables

$z_1 = x_1$, $z_2 = x_2$

$z_3 = y_1$, $z_4 = y_2$

$z_5 = \dot{x}_1$, $z_6 = \dot{x}_2$

$z_7 = \dot{y}_1$, $z_8 = \dot{y}_2$

Make → Array of variables of $z_1, \ldots z_8$

and make find_f → function for returing derivative

& update

$$\text{variables} = \text{variables} + \left(\frac{k_1 + 2k_2 + 2k_3 + k_4}{6}\right)$$

replace f by find f function which returns array of derivatives corresponding eight variables

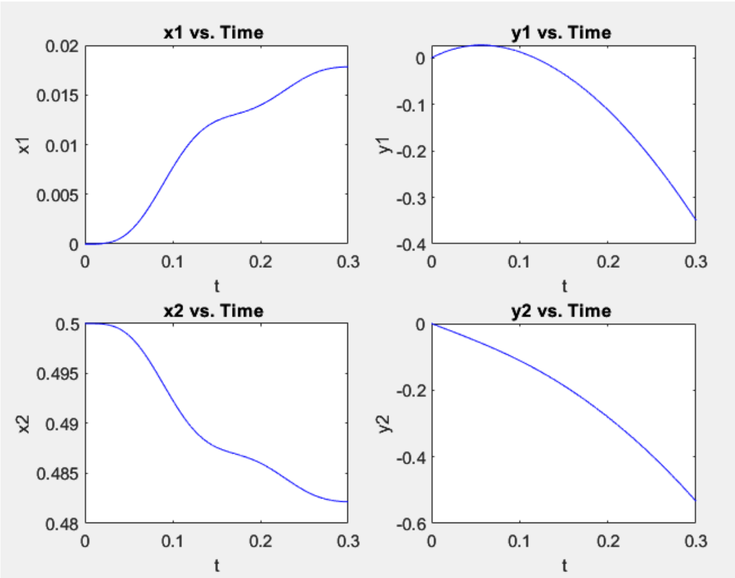
# Que 1) Part B

```
% Name          :- Darpan Gaur
% Roll Number   :- CO21BTECH11004

dt = 0.00001;
t = 0:dt:0.5;

% arrays to store positoin
x1 = zeros(1, length(t));
x2 = zeros(1, length(t));
y1 = zeros(1, length(t));
y2 = zeros(1, length(t));

% arrays to store velocity
x1_1 = zeros(1, length(t));
x2_1 = zeros(1, length(t));
y1_1 = zeros(1, length(t));
y2_1 = zeros(1, length(t));

% set boumdary conditions
x1(1) = 0.0;
x2(1) = 0.5;
y1(1) = 0.0;
y2(1) = 0.0;
x1_1(1) = 0.0;
x2_1(1) = 0.0;
y1_1(1) = 1.0;
y2_1(1) = -1.0;

% constants
l = 0.5;
m1 = 1.0;
m2 = 1.0;
c = 5.0;
k = 1000.0;
g = 0.0;

% array to store Energy
KE = zeros(1, length(t));
S_PE = zeros(1, length(t));
G_PE = zeros(1, length(t));
TE = zeros(1, length(t));

KE(1) = (m1*((x1_1(1))^2 + (y1_1(1))^2) + m2*((x2_1(1))^2 + (y2_1(1))^2))*0.5;
% b = |r2-r1|
b = sqrt((x2(1) - x1(1))^2 + (y2(1) - y1(1))^2);
```

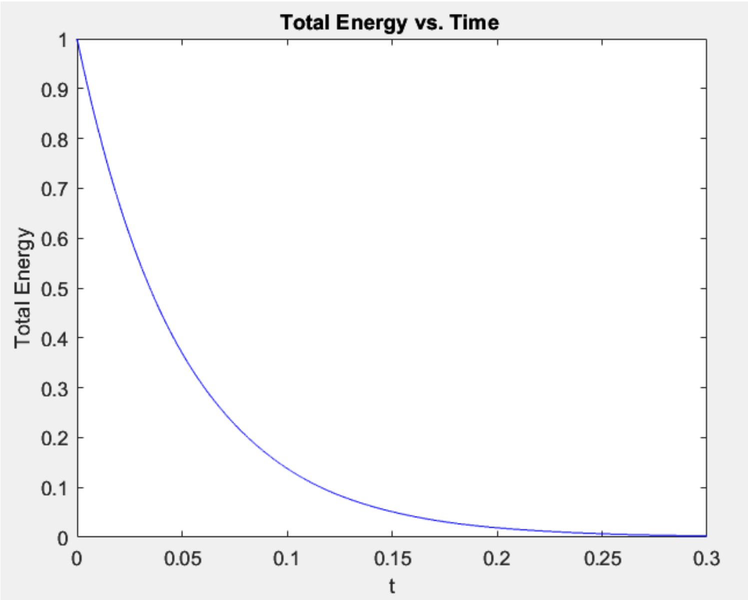
```
S_PE(1) = k*(b-l)*(b-l)*0.5;
G_PE(1) = (m1*y1(1) + m2*y2(1))*g;
TE(1) = KE(1) + S_PE(1) + G_PE(1);

variables = [x1(1); x2(1); y1(1); y2(1); x1_1(1); x2_1(1); y1_1(1); y2_1(1)];
for i=1:length(t)-1
   % b = |r2-r1|
   k1 = dt * find_f(t(i), variables, m1, m2, k, c, l, g);
   k2 = dt * find_f(t(i) + dt/2, variables + k1/2, m1, m2, k, c, l, g);
   k3 = dt * find_f(t(i) + dt/2, variables + k2/2, m1, m2, k, c, l, g);
   k4 = dt * find_f(t(i) + dt, variables + k3, m1, m2, k, c, l, g);

   variables = variables + (k1 + 2*k2 + 2*k3 + k4) / 6;

   x1(i+1) = variables(1);
   x2(i+1) = variables(2);
   y1(i+1) = variables(3);
   y2(i+1) = variables(4);
   x1_1(i+1) = variables(5);
   x2_1(i+1) = variables(6);
   y1_1(i+1) = variables(7);
   y2_1(i+1) = variables(8);
   b = sqrt((x2(i+1) - x1(i+1))^2 + (y2(i+1) - y1(i+1))^2);

   KE(i+1) = (m1*((x1_1(i+1))^2 + (y1_1(i+1))^2) + m2*((x2_1(i+1))^2 +
(y2_1(i+1))^2))*0.5;
   S_PE(i+1) = k*(b-l)*(b-l)*0.5;
   G_PE(i+1) = (m1*y1(i+1) + m2*y2(i+1))*g;
   TE(i+1) = KE(i+1) + S_PE(i+1) + G_PE(i+1);
   TE(i+1) = round(TE(i+1), 3);    % for correcting round off error
end

% plot
figure;

subplot(2,2,1);
plot(t, x1,'b');
xlabel('t');
ylabel('x1');
title('x1 vs. Time ');

subplot(2,2,2);
plot(t, y1, 'b');
xlabel('t');
ylabel('y1');
title('y1 vs. Time ');
```

```
subplot(2,2,3);
plot(t, x2, 'b');
xlabel('t');
ylabel('x2');
title('x2 vs. Time ');
subplot(2,2,4);
plot(t, y2, 'b');
xlabel('t');
ylabel('y2');
title('y2 vs. Time ');

figure;
plot(t, TE, 'b');
xlabel('t');
ylabel('Total Energy');
title('Total Energy vs. Time ');

function f = find_f(~, variables, m1, m2, k, c, l, g)
   x1   = variables(1);    x2   = variables(2);
   y1   = variables(3);    y2   = variables(4);
   x1_1 = variables(5);    x2_1 = variables(6);
   y1_1 = variables(7);    y2_1 = variables(8);

   b = sqrt((x2 - x1)^2 + (y2 - y1)^2);
   fSpring = k * (b - l);
   fD_x = c * (x2_1 - x1_1);
   fD_y = c * (y2_1 - y1_1);

   x1_2 =  (fSpring*(x2 - x1))/(m1*b) + fD_x / m1;
   x2_2 = -(fSpring*(x2 - x1))/(m2*b) - fD_x / m2;
   y1_2 = -g + (fSpring * (y2 - y1)) / (m1 * b) + fD_y / m1;
   y2_2 = -g - (fSpring * (y2 - y1)) / (m2 * b) - fD_y / m2;

   f = [x1_1; x2_1; y1_1; y2_1; x1_2; x2_2; y1_2; y2_2];
end
```

## Plots

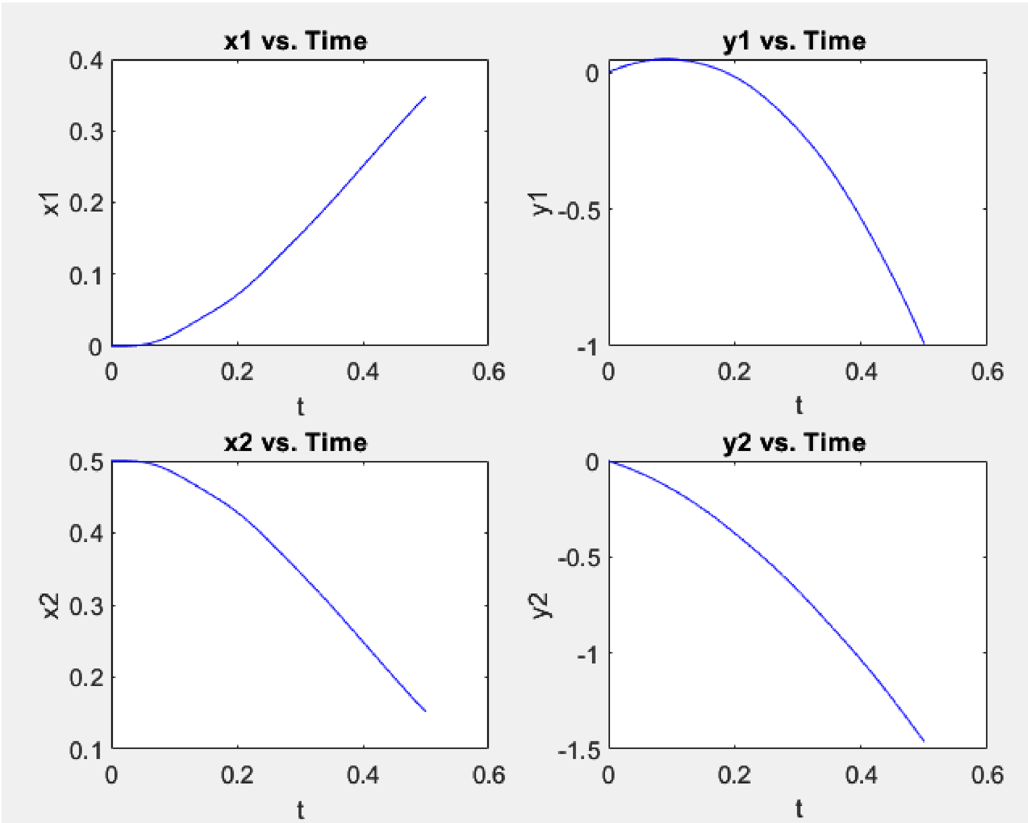


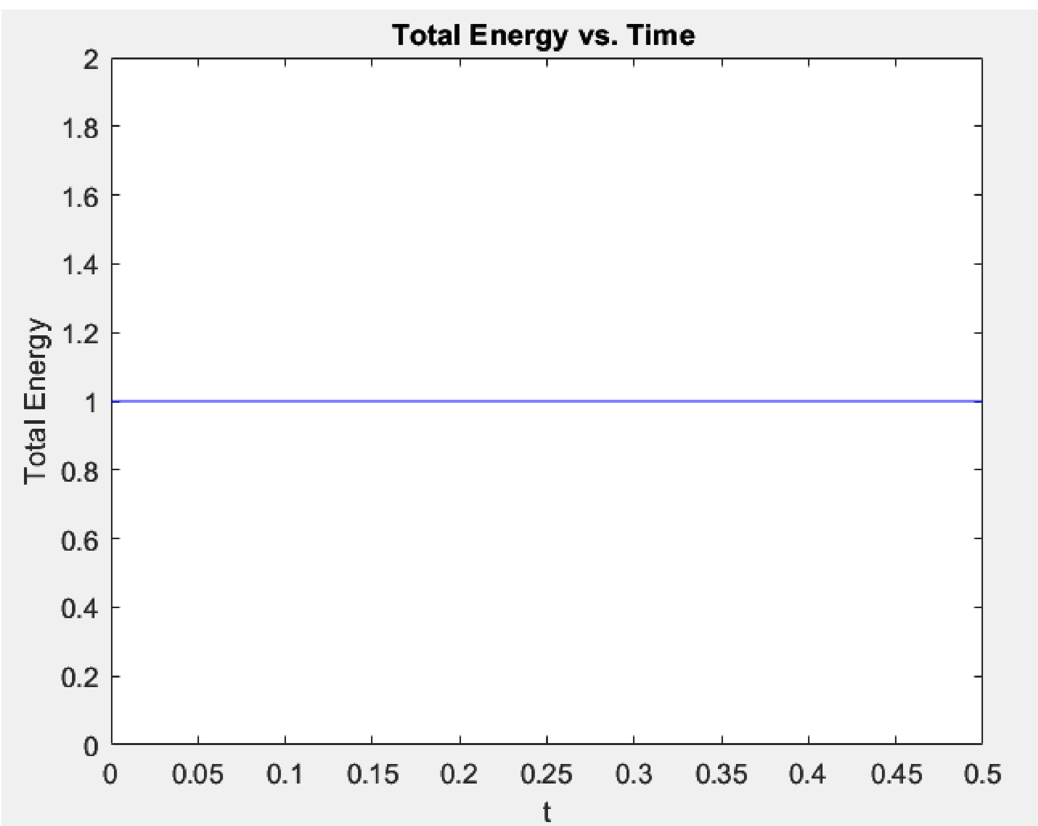

- Plots are prepared with following parameters:
  - c = 5
  - g = 9.81
  - t =0 to 0.3




- Energy decreases as damping is there. So, due to damping energy is lost.

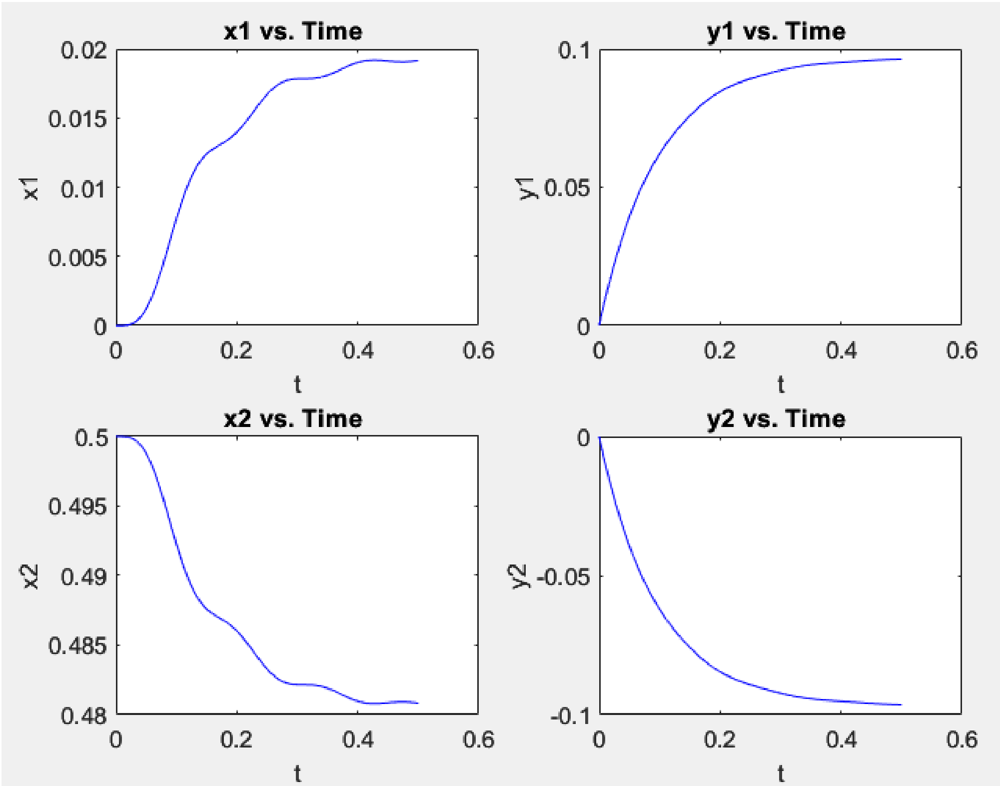
# Que 1) Part C

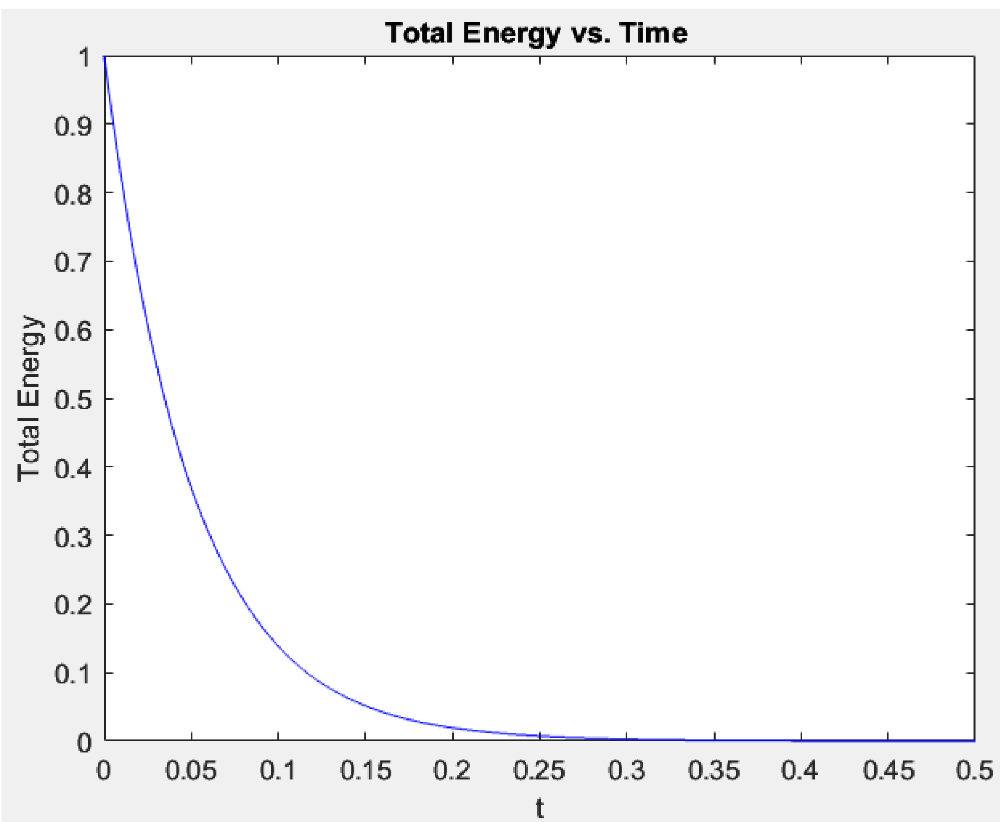



- Put c = 0.0,g=9.81 in the code for this part. Time :- t = 0 to 0.5
- Since, there is no damping so energy is almost constant.
- Smooth graph is formed, as no damping.

# Que 1) Part D




- Put g = 0.0, c=5.0 for this part of the problem in the code. Time :- t = 0 to 0.5
- When no gravity, and damping present, energy decreases.
- y1 reaches a new maximum value as compared to the case when gravity is present because it tends to decrease the velocity and hence distance converd in y direction.