```
%HOMEWORK 8 - MECH 105
%Abdulla Al Ameri
%Date Due: 7, Feb, 2018
```

Problem: 1

```
clear
clc

%Intilaizing variables

syms x; %Defining Variable x

f(x) = (25*(x^3)) + (-6*(x^2)) + (7*x)- 88; %Creating the same
 function twice
g(x) = (25*(x^3)) + (-6*(x^2)) + (7*x)- 88;


xfin = 3; % X value we're interested of
xin = 1; % X value we will start with (base point)

stepsize = abs(xfin - xin ); %Difference between xfin and xin

Actual = g(xfin); %The actual value for the function so we can use it
 in evaluation.


Order = 0; %This will tell us which Taylor degree polynomial we're at.
 Starting at 0.


Int = f(xin) * (stepsize^Order); % The zeroth term of Taylor
 polynomial numerator

Estim = Int/factorial(Order); % The zeroth term of Taylor polynomial
 denominator


R_Err = ones(xin,xfin); % creating an empty vector array to store the
 relative
% error in array called R_Err


    for n = ones(1,3); %The loop will run 3 times. (i.e. 3rd order
 taylor series.)

        Order = Order + 1; %Will increase the term order by 1

        f(x) = diff(f,x,n); %Will differentiate f(x)

        Estim = Estim + ((f(xin)/factorial(Order))*(stepsize^Order));
```

```
        %Calc the taylor estm. next degree polynmial and add the
 results.

        Et = (abs((Estim - g(3)))/(g(3)))*100;
        %Cacluate the relative error.

        R_Err(1,Order) = Et;
        %Store the relative error percent in a new vector

    end

    display(R_Err)

    %To show the relative error percent.
```

```
R_Err =

  85.920577617328519  36.101083032490976                   0
```

Problem: 2

```
clear
clc

%Intilaizing variables

syms x %Defining variable x


f(x) = (25*(x^3)) + (-6*(x^2)) + (7*x)- 88; %Defining the function
fprime(x) = diff(f,x,1); %Defining the derivative of the function.

step = 0.25; %Step size

xvalue = 2; %Middle Value (x)

xminus1 = xvalue - step; %Lower Value (x-step)

xplus1 = xvalue + step; %Upper Value (x+step)


%Note: vpa function will just show the number in decimal rather than
%fraction.


forwEst = vpa((abs(((f(xplus1) - f(xvalue)) / (step)))));
%Forwoard estimation


DiffForw = vpa((abs(forwEst - fprime(2))));
%Difference between forwoard estimation and actual value
```

```
BackwEst = vpa((abs(((f(xvalue) - f(xminus1)) / (step)))));
%Backward estimation

DiffBackw = vpa((abs(BackwEst - fprime(2))));
%Difference between backward estimation and actual value

MiddleEst = vpa((abs(((f(xplus1) - f(xminus1)) / (2*step)))));
%Middle estimation

DiffMiddle = vpa((abs(MiddleEst - fprime(2))));
%Difference between middle estimation and actual value


%all the three estimates have remainders based on taylor reminder
 therom.
```

Published with MATLAB® R2017b