
Problem 1

```
clc
clear
syms x
% defining the original function
f0= @(x) 25*x^3-6*x^2+7*x-88;
N=3; % Taylor series to the order of N
xi=1; % base point
xg=3; % x value to be approximated
h=xg-xi; % step size
actual=f0(xg); %finding actual value
% for loop runs through each order of Taylor series
for n=0:N
    df=(subs(diff(f0,x,n),x,xi)); % finding nth derivative, eval at
    base point
    y=(df*h^n)/factorial(n); % Taylor theorem
    tay(n+1)=y; %creating a array with each value
    approx=sum(tay) % summing the array to get approximation
    error=(actual-approx)/actual; % error calculations
    error=double(error)*100
end
```

approx =

-62

error =

111.1913

approx =

78

error =

85.9206

approx =

354

error =

36.1011

```
approx =
```

```
554
```

```
error =
```

```
0
```

Problem 2

```
clear
clc
syms x
% function to be evaluated
f0= @(x) 25*x^3-6*x^2+7*x-88;
h=.25; % step size
xi=2; % what we want the derivative to be evaluated at
xf=xi+h; % x_i+1
xl=xi-h; % x_i-1
actual=(subs(diff(f0,x,1),x,xi)); %finding actual value of f'(2)
actual=double(actual); %making actual a floating point number
%calculating the approximations and the errors
forward= (f0(xf)-f0(xi))/h
ferror=abs((actual-forward)/actual)*100

backward= (f0(xi)-f0(xl))/h
berror=abs((actual-backward)/actual)*100

center= (f0(xf)-f0(xl))/h
cerror=abs((actual-center)/actual)*100

% the backwards approx gives the closest answer to the actual, it
% would
% need a smaller taylor series order to achieve the correct answer, it
% also
% means the average change of f(x) at 1.75<x<2 is the closest to the
% change of f(x)
% at x=2

forward =

320.5625

ferror =

13.2730
```

backward =

248.5625

berror =

12.1687

center =

569.1250

cerror =

101.1042

Published with MATLAB® R2017b