# *Experiment 3*

# *Maxima and Minima of Single Variable Function*

## *I. Aim*

To find the local extremum, global maximum and minimum of a single variable function and plot them.

## *II. Mathematical Background:*

In mathematics, the maxima and minima of a function, known collectively as extrema, are the largest and smallest value of the function, either within a given range or on the entire domain of a function. In many applications, we are interested in finding the maximum and minimum value of a function over some specified interval. We will be using the below two methods:

1. Suppose f(x) is defined on some interval (a, b) and f has a local minimum or a local maximum at c in (a, b). Then derivative of f(x) at c = 0.
2. For a continuous function f on a bounded interval [a, b], suppose c is a global extremum. Then c must satisfy one of the following:
   a) f(c) = 0.
   b) Derivative of f at c does not exist.
   c) c is an endpoint.

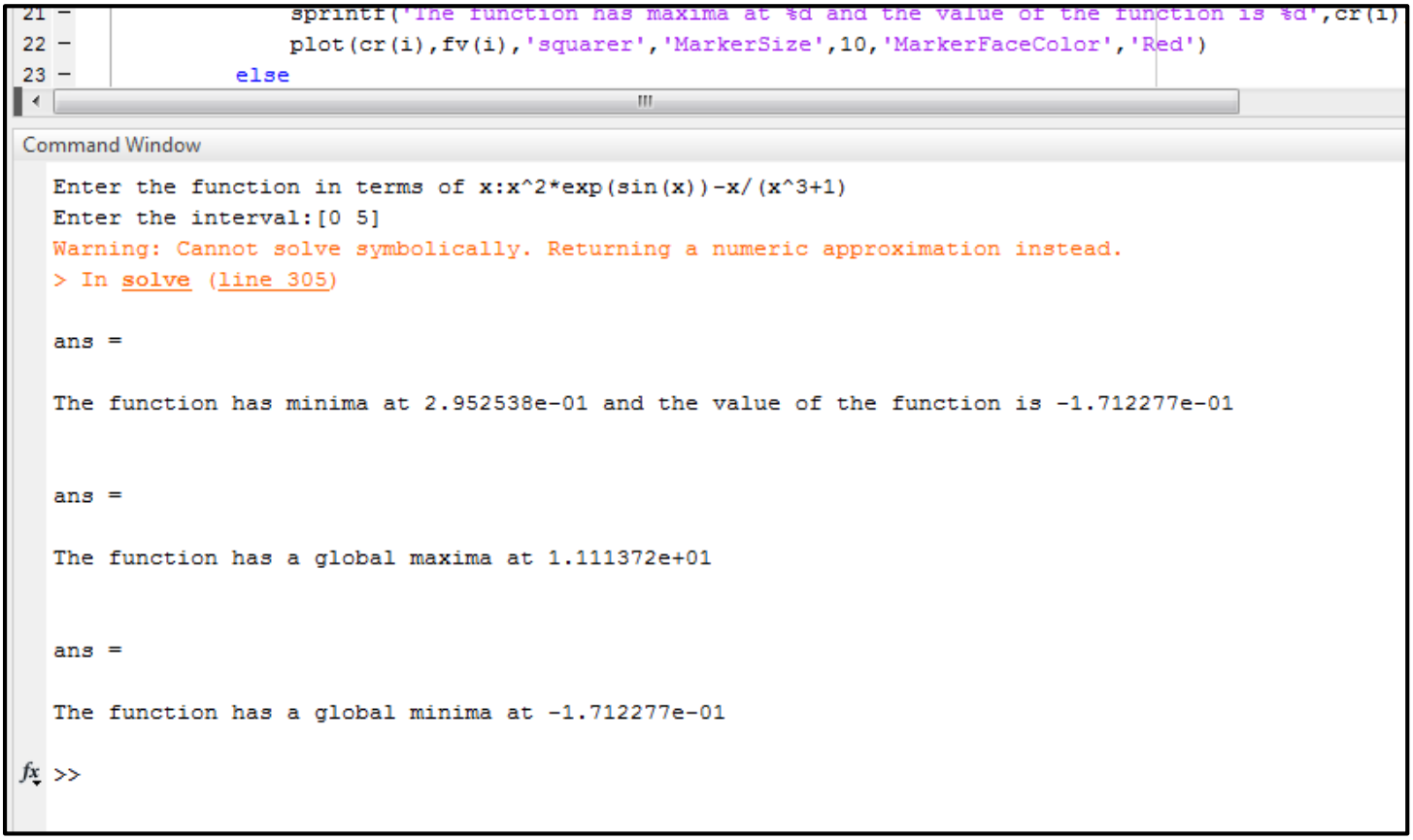
At last we check the sign of double derivative to see if the point is maxima or minima or a point of inflection using for loop.

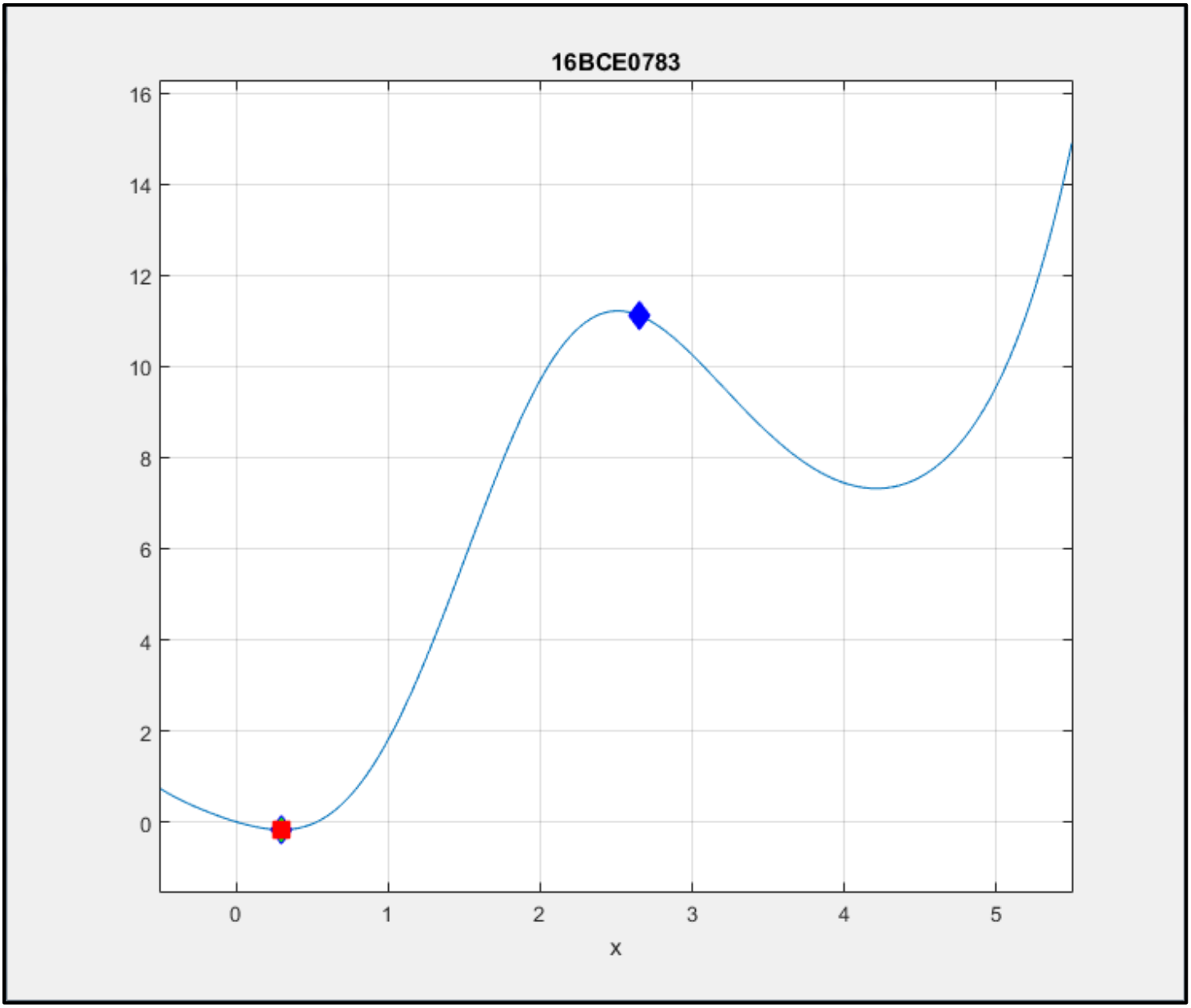
## *III. MATLAB Code:*

```
clc
clearvars
syms x    %symbolic function - x is a symbol
f = input('Enter the function in terms of x:');
I = input('Enter the interval:');
L = I(2) - I(1); %Length of the interval
h = ezplot(f,I+L/10*[-1 1]); hold on; grid on;
df=diff(f,x);
cr=double(solve(df,x,'Real',true));
%getting critical values of x, double converts symbolic to numeric, solve for only real
values of x
index = find(cr>I(1) & cr<I(2));
cr = unique(cr(index)); %Removing repeted values
fv = double(subs(f,x,cr));   %substituting values of cr in f in place of x
if (numel(cr)~=0)              % numel(c) -> number of elemants in the vector cr.
```

```
    for i=1:numel(cr)
        d2f= double(subs(diff(df,x),x,cr(i)));
        if (d2f>0)
            sprintf('The function has minima at %d and the value of the function is
%d',cr(i),fv(i))
            plot(cr(i),fv(i),'diamondb','MarkerSize',10,'MarkerFaceColor','Green')
        elseif (d2f<0)
            sprintf('The function has maxima at %d and the value of the function is
%d',cr(i),fv(i))
            plot(cr(i),fv(i),'squarer','MarkerSize',10,'MarkerFaceColor','Red')
        else
            sprintf('The function has inflection point at %d and the value of the
function is %d',cr(i),fv(i))
            plot(cr(i),fv(i),'diamondb','MarkerSize',10,'MarkerFaceColor','Blue')
        end
    end
end
cr=[I(1):cr:I(2)];  %including end values also
fvb=double(subs(f,x,cr));
[gmax,mx]=max(fvb);
[gmin,mn]=min(fvb);
sprintf('The function has a global maxima at %d ',gmax)   %(putting data in string)
plot(cr(mx),gmax,'diamondb','MarkerSize',10,'MarkerFaceColor','Blue')
sprintf('The function has a global minima at %d',gmin)
plot(cr(mn),gmin,'squarer','MarkerSize',10,'MarkerFaceColor','Red')
title('16BCE0783')
```

# *IV. MATLAB I/O:*






## *V. Question - Answers:*

The code written in the lab record above will not show correct results and all points marked when we use a function like, which contains a trigonometric function in exponential form.

Actually MATLAB cannot solve function of a function for x. Hence, to Correct that, instead of using solve command, we have to vpasolve() command. What vpasolve does that it solves the equation numerically whereas solve command solves the equation symbolically. For the equations which are not polynomial, MATLAB returns the first numerically solution found.

# Corrected Code:

```
clc
clearvars
syms x    %symbolic function - x is a symbol
f = input('Enter the function in terms of x:');
I = input('Enter the interval:');
L = I(2) - I(1); %Length of the interval
h = ezplot(f,I+L/10*[-1 1]); hold on; grid on;
df=diff(f,x);
cr=double(vpasolve(df,x,'Real',true));
%getting critical values of x, double converts symbolic to numeric, solve for only real
values of x
index = find(cr>I(1) & cr<I(2));
cr = unique(cr(index)); %Removing repeted values
fv = double(subs(f,x,cr));   %substituting values of cr in f in place of x
if (numel(cr)~=0)               % numel(c) -> number of elemants in the vector cr.
    for i=1:numel(cr)
        d2f= double(subs(diff(df,x),x,cr(i)));
        if (d2f>0)
            sprintf('The function has minima at %d and the value of the function is
%d',cr(i),fv(i))
            plot(cr(i),fv(i),'diamondb','MarkerSize',10,'MarkerFaceColor','Green')
        elseif (d2f<0)
            sprintf('The function has maxima at %d and the value of the function is
%d',cr(i),fv(i))
            plot(cr(i),fv(i),'squarer','MarkerSize',10,'MarkerFaceColor','Red')
        else
            sprintf('The function has inflection point at %d and the value of the
function is %d',cr(i),fv(i))
            plot(cr(i),fv(i),'diamondb','MarkerSize',10,'MarkerFaceColor','Blue')
        end
    end
end
cr=[I(1):cr:I(2)];  %including end values also
fvb=double(subs(f,x,cr));
[gmax,mx]=max(fvb);
[gmin,mn]=min(fvb);
sprintf('The function has a global maxima at %d ',gmax)   %(putting data in string)
plot(cr(mx),gmax,'diamondb','MarkerSize',10,'MarkerFaceColor','Blue')
sprintf('The function has a global minima at %d',gmin)
plot(cr(mn),gmin,'squarer','MarkerSize',10,'MarkerFaceColor','Red')
title('16BCE0783')
```