

```

+++++++                               Lab 003                               ++++++
+++++++                               2154638                               ++++++

```

CSCM70

CODE:

1)

```

% CSCM 70 ----- LAB 03 -----
% CSCM 70 ----- 2154638 -----
% =====
All parts in one also copied the diary in later command window Attached Screenshot
as well.
- Graphs listed below.

```

```

% first_der.m file
% CSCM 70 ----- LAB 03 -----
% CSCM 70 ----- 2154638 -----
% =====

% =====

function res = first_der(f,x,tolerance,h)
err = 1000;
% res = (f(x+h)-f(x-h))/(2*h);
res = (f(x+h) - f(x))/h;
while err > tolerance
    h = h/10;
    end_res = res;
%     res = (f(x+h)-f(x-h))/(2*h);
    res = (f(x+h) - f(x))/h;
    err = abs(res - end_res);
end
end

```

2.)

---Continued after 1st question is over

```

% =====

```

OUTPUT : COMMAND WINDOW

1.)

>>

diary on

f1 = @(x) 5

f1 =

[function_handle](matlab:helpPopup function_handle) with value:

@(x)5

first_der(f1,0,0.0001,0.01)

ans =

0

f2 = @(x) x^2 -x;

first_der(f2,0,0.0001,0.01)

ans =

-1

f3 = @(x) exp(2*x);

first_der(f3,0,0.0001,0.01)

ans =

2.0000

```
f4 = @(x) abs(x)
```

```
f4 =
```

[function_handle](matlab:helpPopup function_handle) with value:

```
@(x)abs(x)
```

```
first_der(f4,0,0.0001,0.01)
```

```
ans =
```

```
1
```

```
diary off
```

Running the 1.4 in the commented formula(`% res = (f(x+h)-f(x-h))/(2*h);`) will result in :

```
f4 = @(x) abs(x)
```

```
f4 =
```

[function_handle](matlab:helpPopup function_handle) with value:

```
@(x)abs(x)
```

```
first_der(f4,0,0.0001,0.01)
```

```
ans =
```

1

>>



The output on online editor was as below:

The screenshot shows the MATLAB online editor interface. At the top, there are two tabs: 'first_der.m' and 'untitled *'. The 'first_der.m' tab is active, displaying a function definition for 'first_der'. The function takes four arguments: 'f', 'x', 'tolerance', and 'h'. It initializes 'err' to 1000, calculates 'res' as $(f(x+h) - f(x-h)) / (2*h)$, and enters a 'while' loop that continues as long as 'err' is greater than 'tolerance'. Inside the loop, 'h' is updated to 'h/10'. Below the code editor is the 'Command Window'. It contains a yellow banner with the text 'New to MATLAB? See resources for Getting Started.' followed by MATLAB internal event manager messages. Below these, there are four lines of output, each starting with 'ans =' and followed by a value: 0, -1, 2.0000, and 0. At the bottom of the Command Window, there is a warning message: 'Warning: Error occurred while executing the listener'.

```
function res = first_der(f,x,tolerance,h)
    err = 1000;
    res = (f(x+h)-f(x-h))/(2*h);
    while err > tolerance
        h = h/10;
```

Command Window

New to MATLAB? See resources for Getting Started.

In matlab.internal.mvm.eventmgr.MVMEvent.invokeListen
In matlab.internal.mvm.eventmgr.MVMEvent>@(eventTags,

ans =
0

ans =
-1

ans =
2.0000

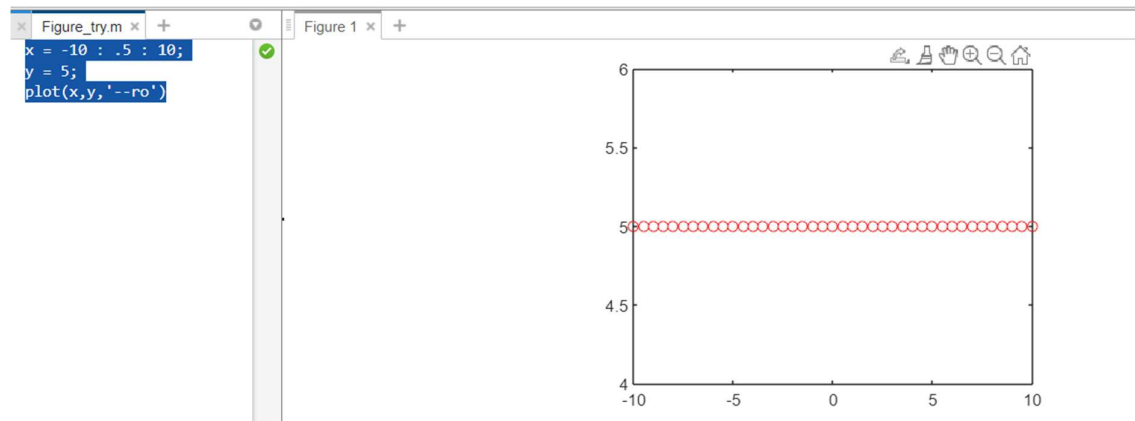
ans =
0

Warning: Error occurred while executing the listener

GRAPHS: First try

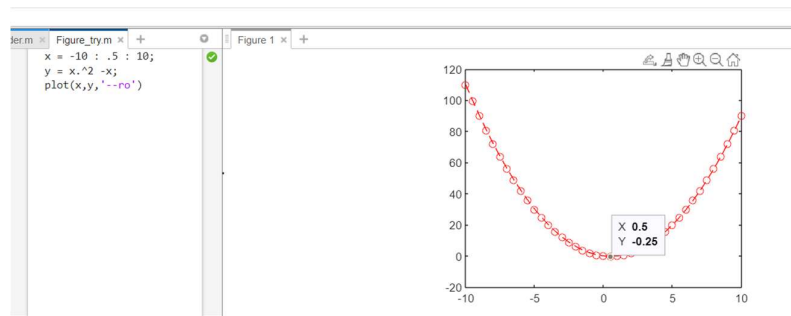
Which is expected as per the below graphs of each of the above 4 equations:

1.1



1.2

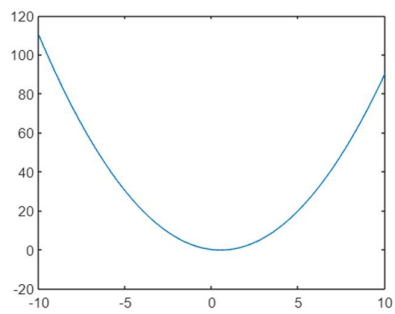
```
x = -10 : .25 : 10;  
y = x.^2 -x;  
plot(x,y,'-ro')
```



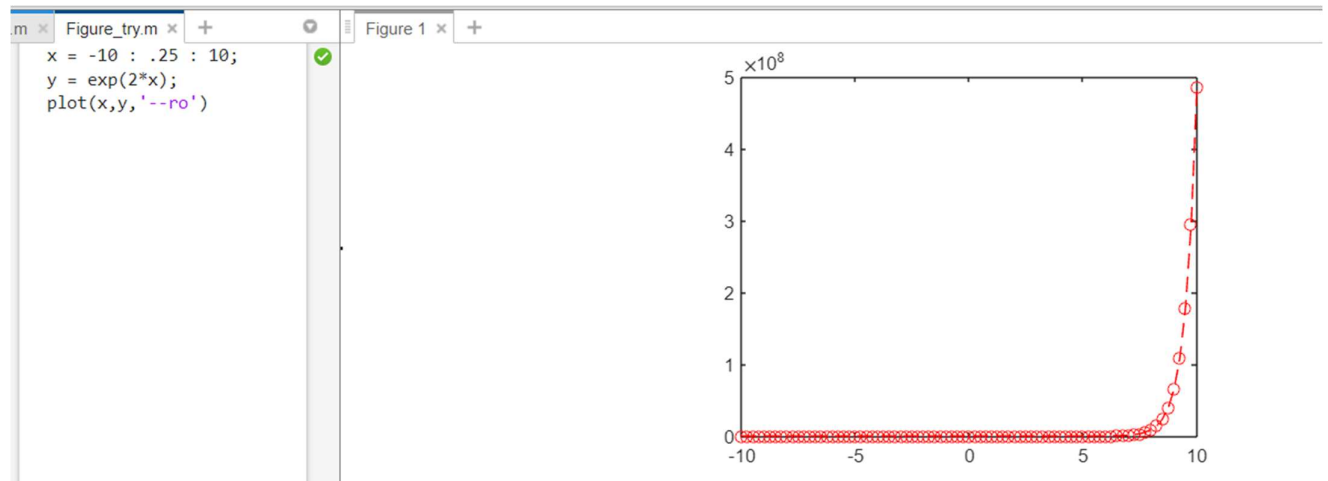
and Window

MATLAB? See resources for Getting Started.

[MATLAB? See resources for Getting Started.](#)

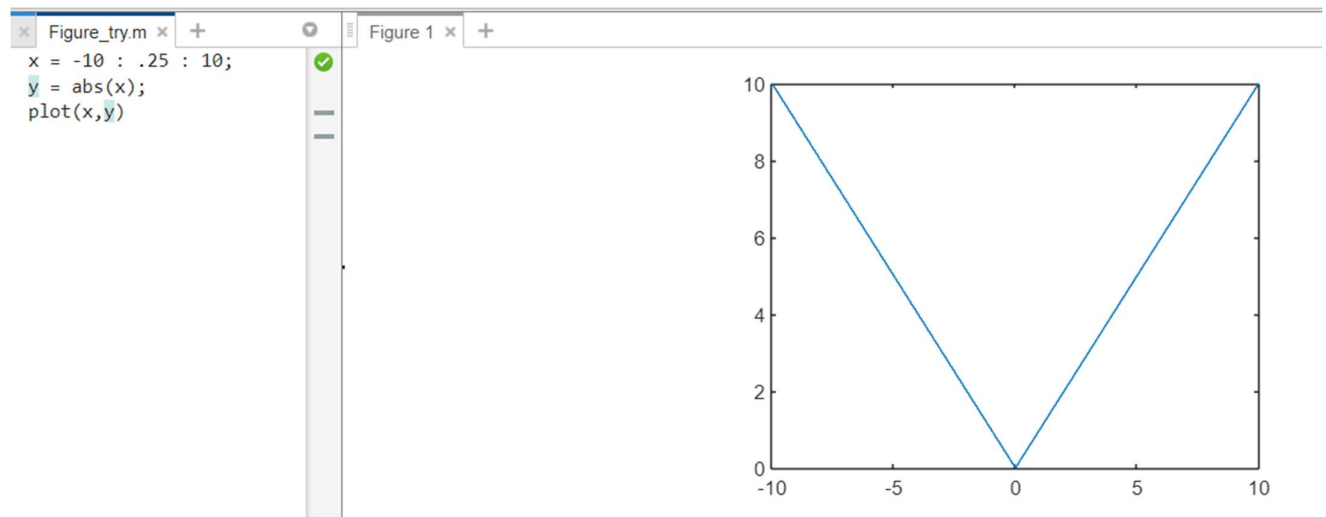


1.3



d Window

1.4



ANOTHER TRY FOR 1.4)

```
→  
→ % CSCM 70 ----- LAB 03 -----  
→ % CSCM 70 ----- 2154638 -----  
→ % =====  
→  
→ % =====  
→  
→ % function res = first_diff(f,x,tolerance,h)  
→ function res = first_diff()  
→ f = @(x) abs(x) ;  
→ x = 0;  
→ tolerance = 0.0001;  
→ h = 0.01;  
→ err_rhs = 1000;  
→ err_lhs = 1000;  
→  
→  
→ % res = (f(x+h)-f(x-h))/(2*h);  
→ res_rhs = (f(x+h) - f(x))/h;  
→ res_lhs = (f(x) - f(x-h))/h;  
→  
→ tolerance = 0.0001;  
→ h = 0.01;  
→ % RHS  
→ while err_rhs > tolerance  
→ main_res_rhs = res_rhs;  
→ res_rhs = (f(x+h) - f(x))/h;  
→ err_rhs = (res_rhs - main_res_rhs);  
→ res_err_rhs = err_rhs;  
→
```

```

→ end
→
→ tolerance = 0.0001;
→ h = 0.01;
→ % LHS
→ while err_lhs > tolerance
→   main_res_lhs = res_lhs;
→   res_lhs = (f(x) - f(x-h))/h;
→   err_lhs = (res_lhs - main_res_lhs);
→   res_err_lhs = err_lhs;
→
→ end
→ % res = res_err_rhs - res_err_lhs;
→ disp(res_err_rhs)
→ disp(res_err_lhs)
→ end
→

```

COMMAND WINDOW O/P

```
>> f = @(x) abs(x);
```

```
>>
```

```
>> first_diff()
```

```
0
```

```
0
```

>> So basically, it should be 1 and -1 the left and right limit for both of them and So it should turn to NAN or not defined.

2.)

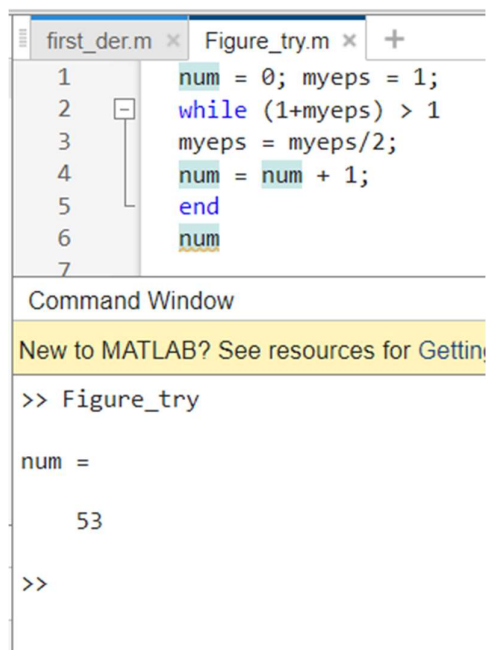
Answer 2:

That is as per the documentation at <https://in.mathworks.com/help/matlab/ref/eps.html> expression `d = eps` returns the distance from 1.0 to the next larger double-precision number, that is, 2^{-52} . Therefore the loop runs 52 times + 1(exit loop), hence num = 53.

```

num = 0; myeps = 1;
while (1+myeps) > 1
myeps = myeps/2;
num = num + 1;
end
num

```

The image shows a MATLAB interface with two windows. The top window, titled 'Figure_try.m', contains a script with the following code:

```
1 num = 0; myeps = 1;
2 while (1+myeps) > 1
3     myeps = myeps/2;
4     num = num + 1;
5 end
6 num
```

The bottom window is the 'Command Window'. It shows the command 'Figure_try' being executed, followed by the output 'num = 53'.

The output comes as expected as per the documentation.

Epsilon in simple sense means that after some values the computer will round up the decimals and because of that some errors are created.

In this above example we understood the epsilon error. This error is caused because of the storage problem. After certain number computer cannot visualize the errors.

Machine error = Machine epsilon

3.)

Answer 3:

Code:

```
% CSCM 70 ----- LAB 03 -----
% CSCM 70 ----- 2154638 -----
% =====

% =====

% function res =sign_der(f,x,h)
function res =sign_der()
f = @(x) sin(x);
x = .5;
h = 1;
% res = (f(x+h)-f(x-h))/(2*h);
res = (f(x+h) - f(x))/h;
i=1;
for i= 1 :20
    h = h/10;
    % disp(h)
    % res = (f(x+h)-f(x-h))/(2*h);
    res = (f(x+h) - f(x))/h;
    disp([i res]);
    i = i + 1;
    disp(res)
end
% disp(h)
end
```

Output:

```
>> diary '03_third_question_all_parts.txt'

>> diary on

>> sign_der()

    1.0000    0.8522

    0.8522

    2.0000    0.8752

    0.8752
```

3.0000 0.8773

0.8773

4.0000 0.8776

0.8776

5.0000 0.8776

0.8776

6.0000 0.8776

0.8776

7.0000 0.8776

0.8776

8.0000 0.8776

0.8776

9.0000 0.8776

0.8776

10.0000 0.8776

0.8776

11.0000 0.8776

0.8776

12.0000 0.8776

0.8776

13.0000 0.8776

0.8776

14.0000 0.8771

0.8771

15.0000 0.8882

0.8882

16.0000 1.1102

1.1102

17 0

```

0

18      0

0

19      0

0

20      0

0

ans =

0

>> diary off

>>

```

The value of derivative of $\sin(x)$ crosses comes out at zero. This is the case as the value exceeded even the maximum value of the function at 16 step itself , hence zero came as the answer.

And after the 16th step we will get the values as 0 itself. As the computer will keep on progressing till 20th iteration.

Answer 4

One:

```
1 % x → 1/x;  
2 f = @(x) 1/x;  
3 % pkg load symbolic;  
4 syms x;  
5 ff = f(x);  
6 ffd = diff(ff,x)  
7
```

Command Window

New to MATLAB? See resources for [Getting Started](#)

```
>> first_der
```

```
ffd =
```

```
-1/x^2
```

Two

```
1 f = @(x) (x^2)/cos(x)
2 % pkg load symbolic
3 syms x
4 ff = f(x);
5 ffd = diff(ff,x)
6
```

Command Window

New to MATLAB? See resources for [Getting Started](#).

```
>> first_der
```

```
ffd =
```

```
-1/x^2
```

```
>> first_der
```

```
f =
```

[function_handle](#) with value:

```
@(x)(x^2)/cos(x)
```

```
ffd =
```

```
(2*x)/cos(x) + (x^2*sin(x))/cos(x)^2
```

```
>>
```

Third

```
>> first_der
```

```
f =
```

[function_handle](#) with value:

```
@(x)cos(x)^2+sin(x)^2
```

```
ffd =
```

```
0
```

```
>>
```

+++++	Lab 003	END	+++++
+++++	2154638		+++++
CSCM70			
