SVKM'S NMIMS Deemed to be UNIVERSITY | NAVI MUMBAI

MATLAB

Unit 7-Lecture 28 and 29

Debugging M-files

BTech (CSBS) -Semester VII

18 October 2022, 09:35AM

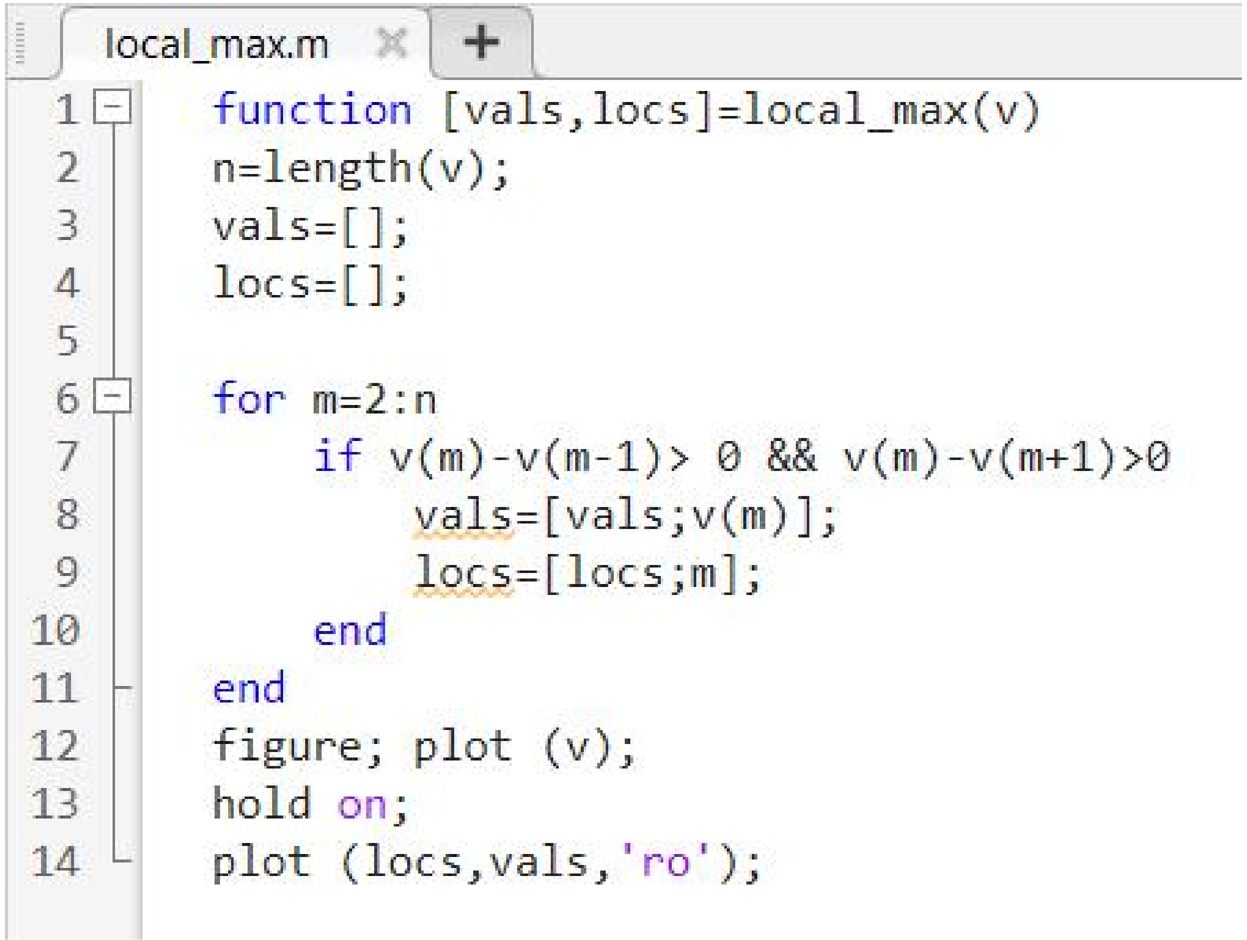
*few figures/content have been prepared/referred from internet source/books

Control Flow and Operators

1) preparing for debugging,
2) examining values,
3) Debugging process
4) setting breakpoints
5) running with breakpoints
6) correcting an M- file
7) correcting and ending debugging

Running with breakpoints

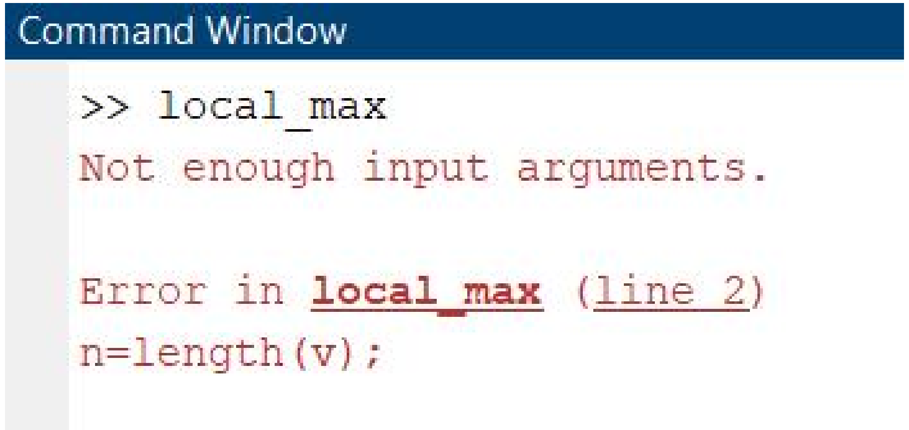
Code:



```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

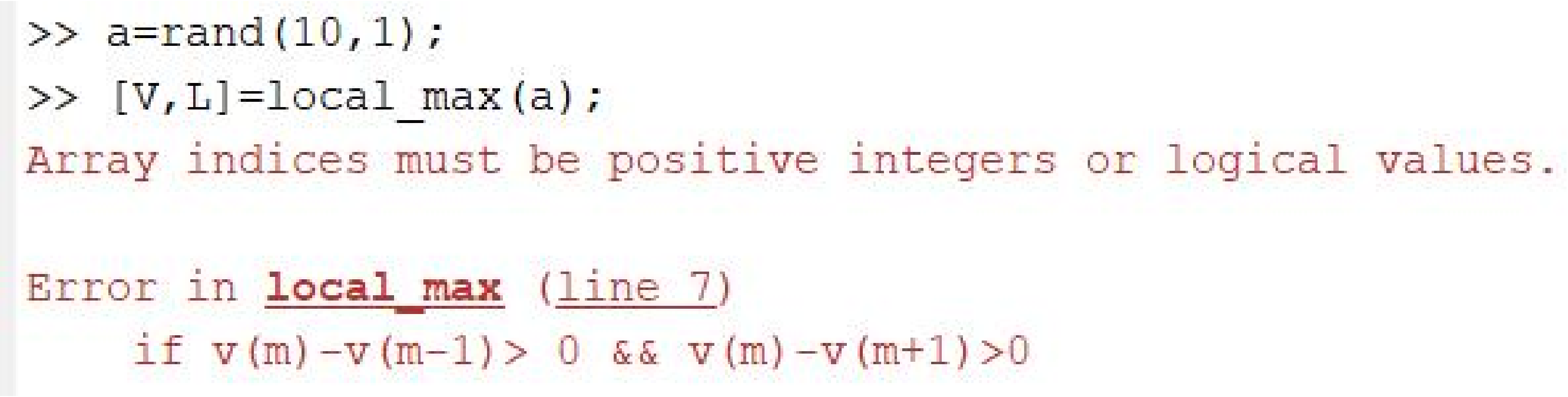
Running with breakpoints

Run:



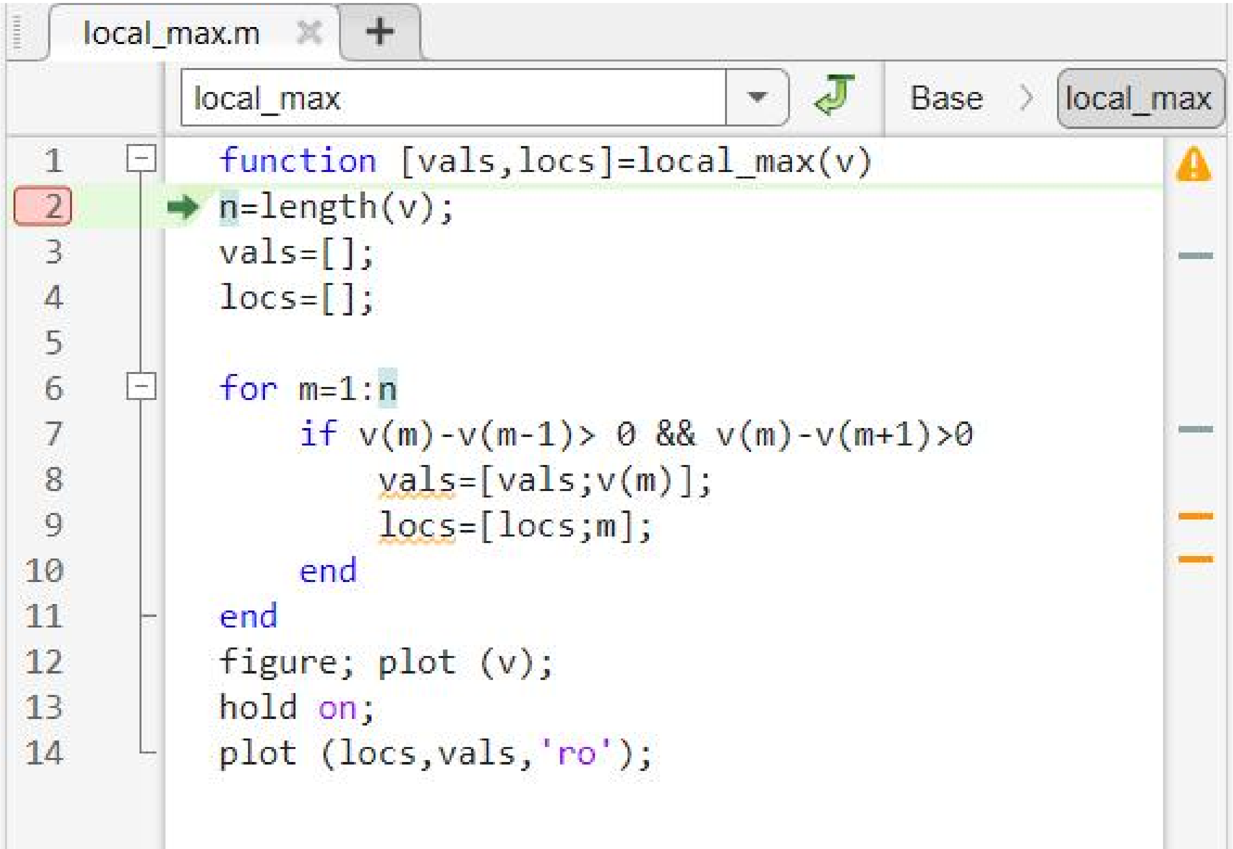
```
>> local_max
Not enough input arguments.

Error in local_max (line 2)
n=length(v);
```

Run some value:

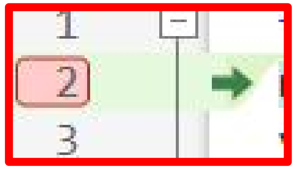


```
>> a=rand(10,1);
>> [V,L]=local_max(a);
Array indices must be positive integers or logical values.

Error in local_max (line 7)
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
```

Set the breakpoint

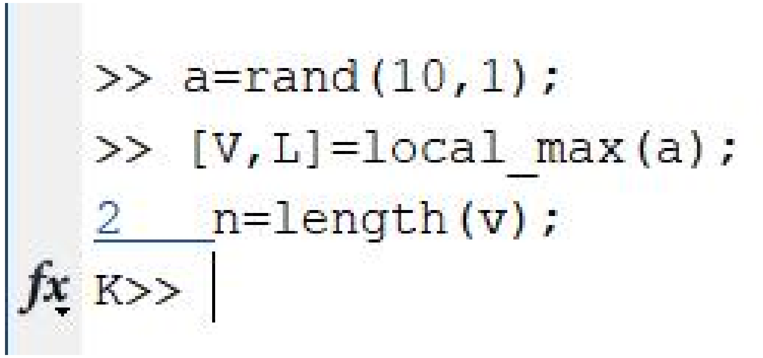


```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```



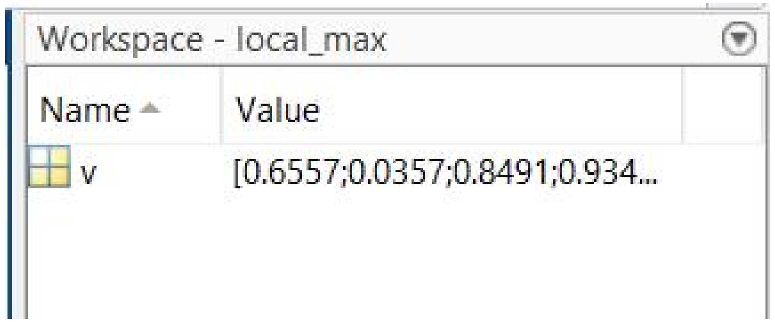
This means the code is not executed yet

Result:



```
>> a=rand(10,1);
>> [V,L]=local_max(a);
2    n=length(v);
K>>
```

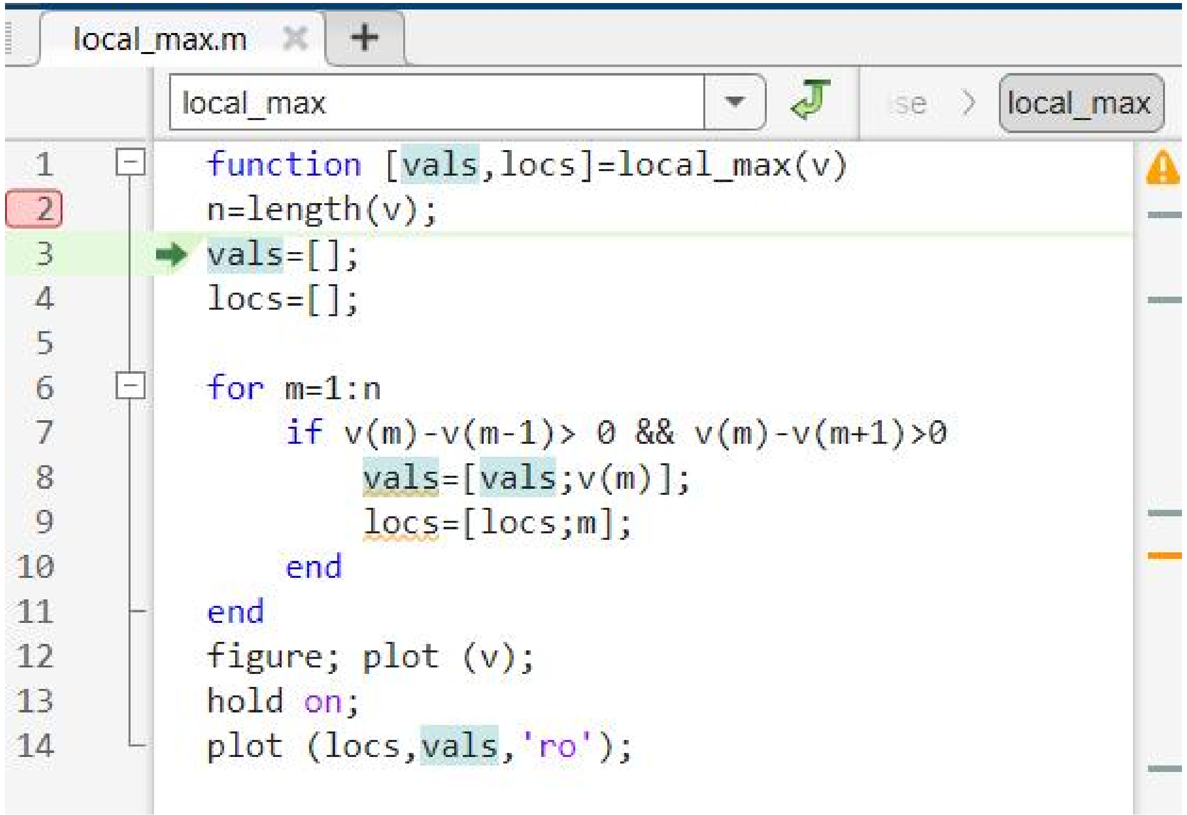
All workspace variable are gone:



Workspace - local_max

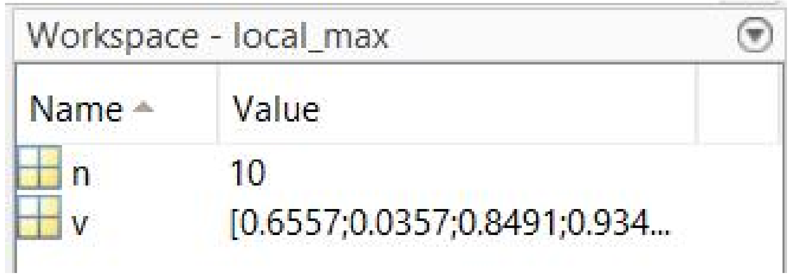
Name	Value
v	[0.6557;0.0357;0.8491;0.934...

Execute line: Press Step



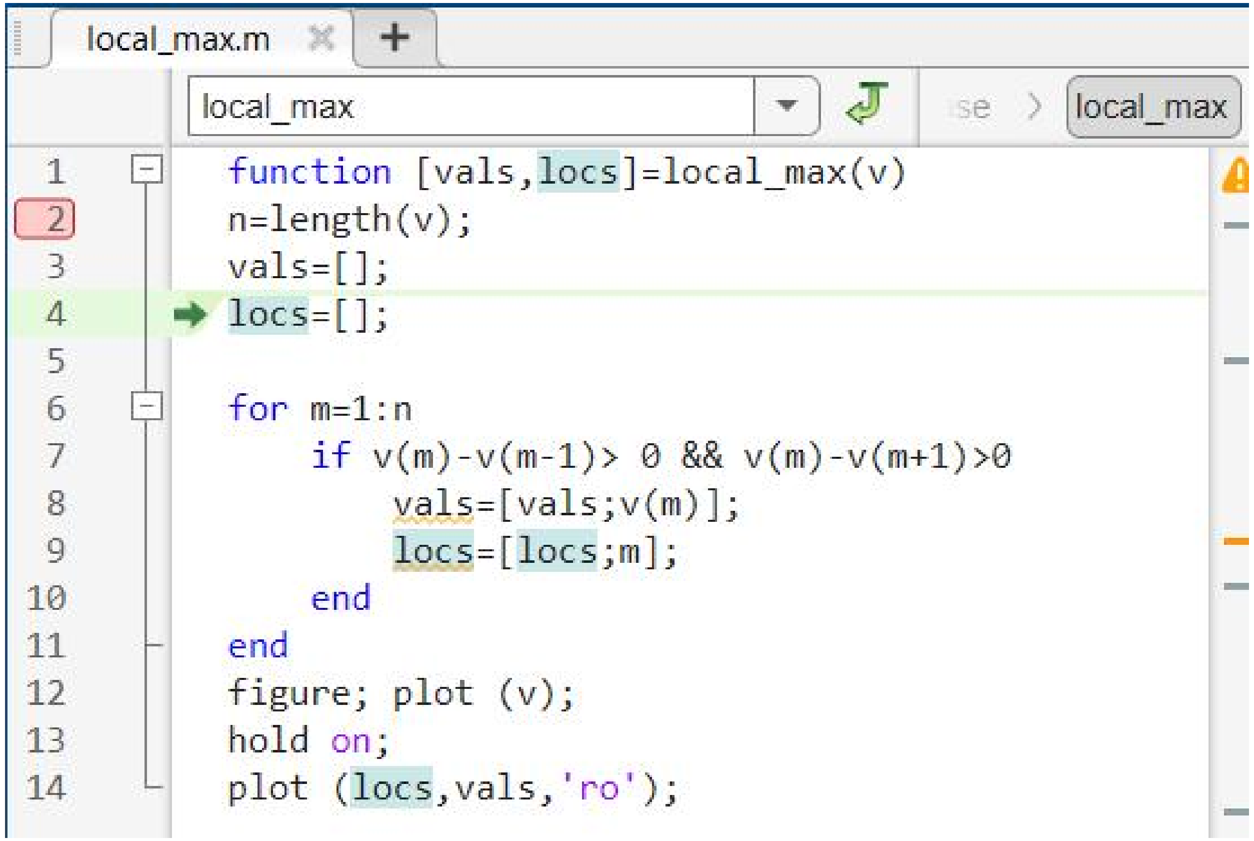
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Workspace is declared with n values



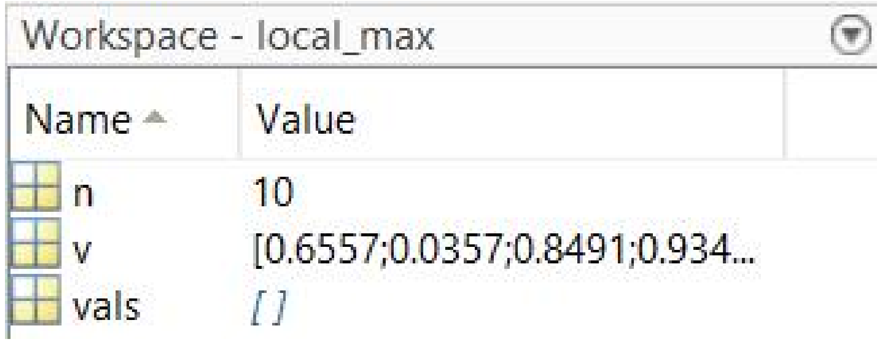
Name	Value
n	10
v	[0.6557;0.0357;0.8491;0.934...

Execute line: Press Step Step again anad again till error line



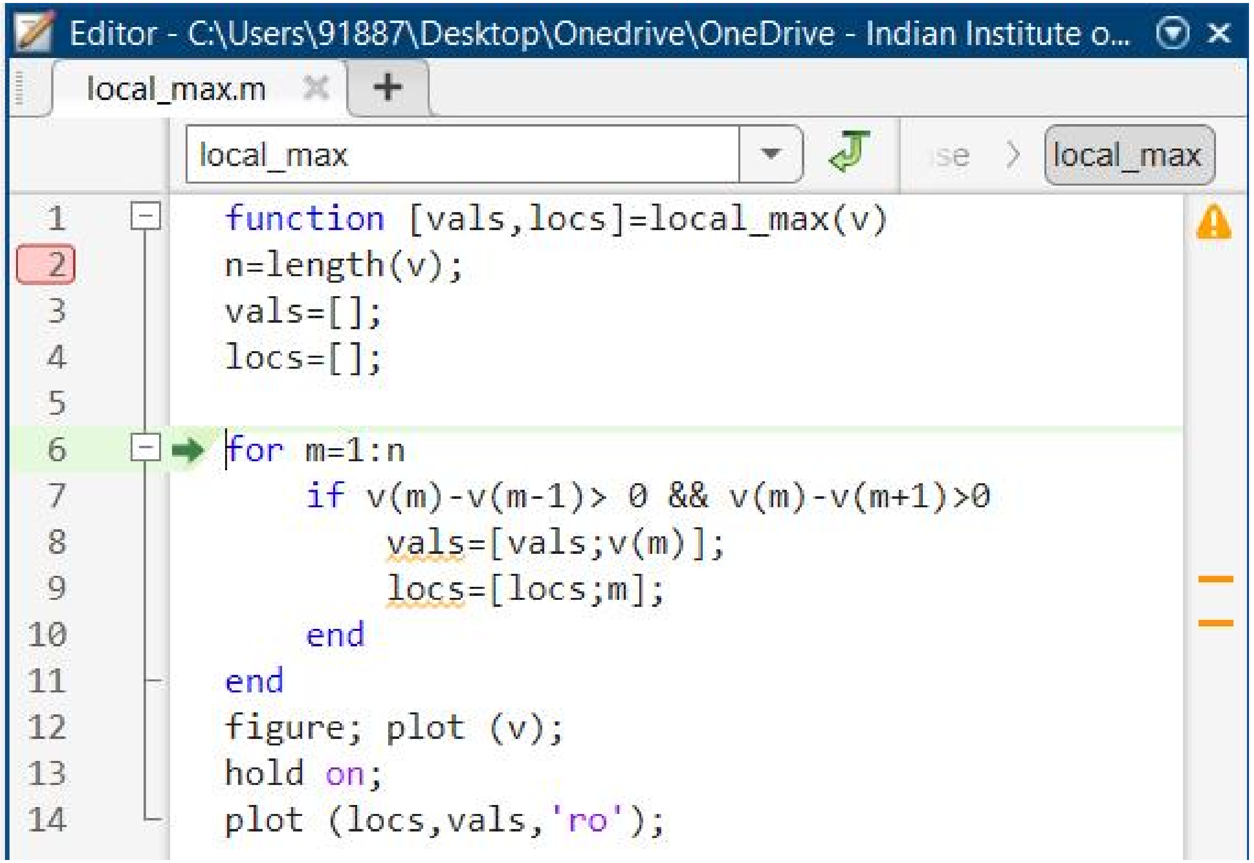
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Workspace is declared with n values



Name	Value
n	10
v	[0.6557;0.0357;0.8491;0.934...
vals	[ ]

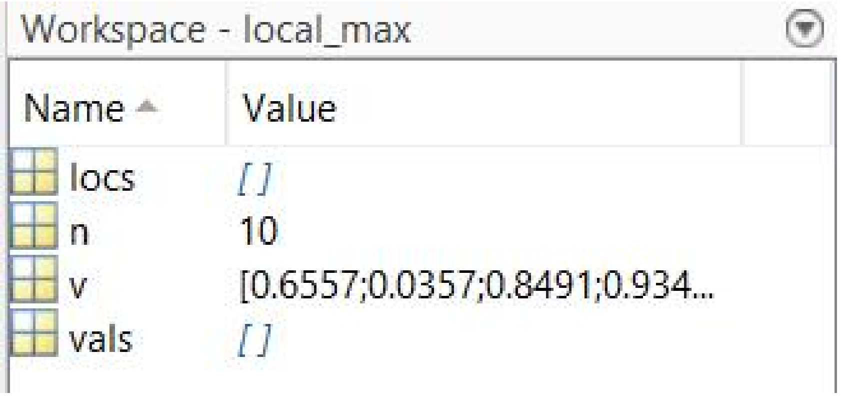
Execute line: Press Step Step again anad again till error line



Editor - C:\Users\91887\Desktop\Onedrive\OneDrive - Indian Institute o...
local_max.m
+
local_max
se
local_max

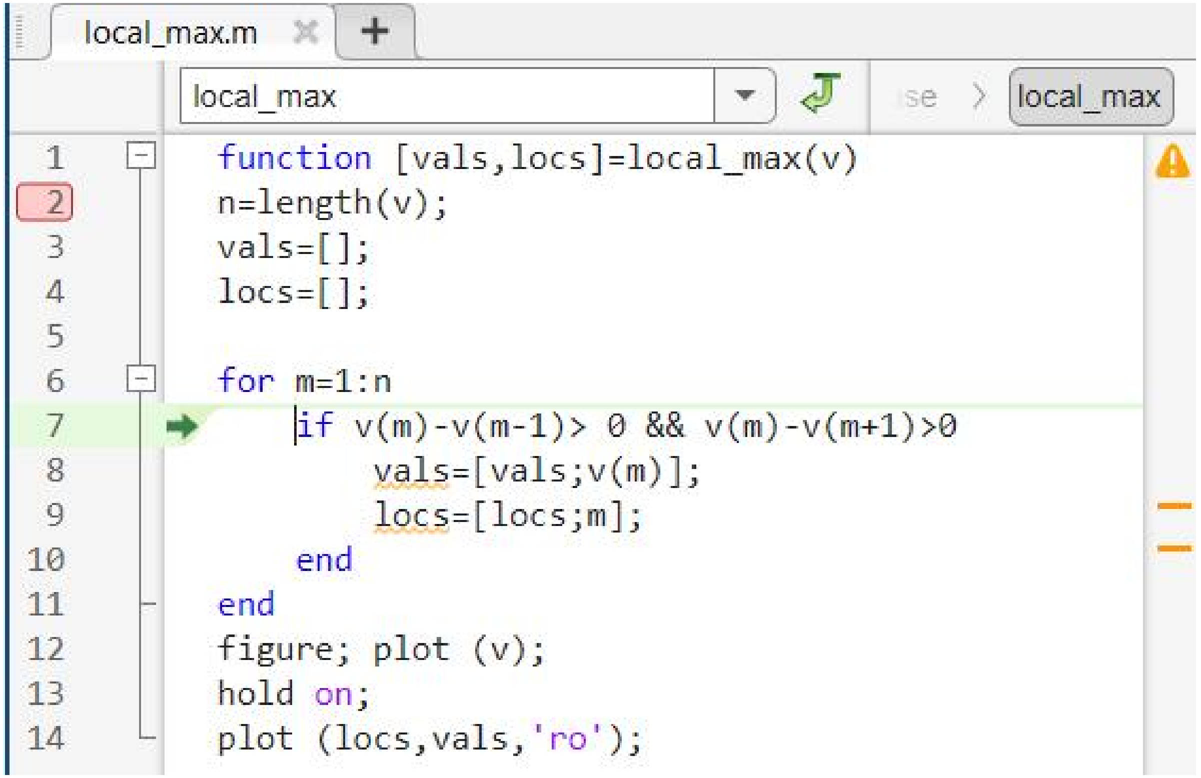
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Workspace is declared with n values



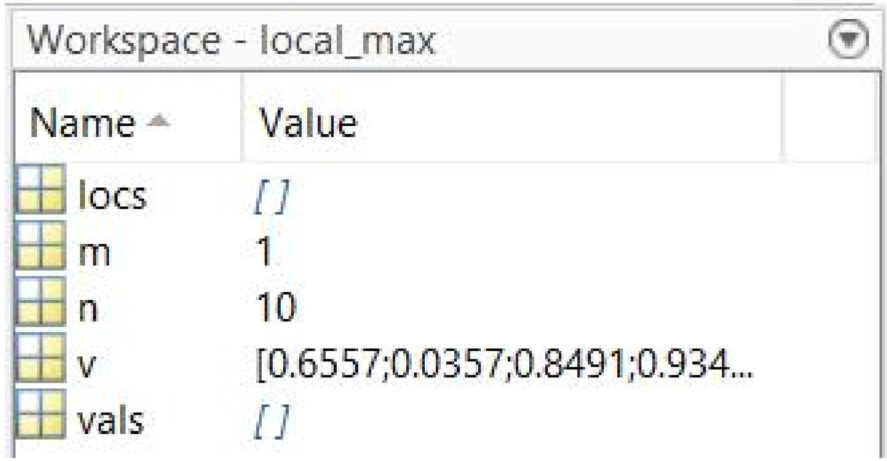
Name	Value
locs	[]
n	10
v	[0.6557;0.0357;0.8491;0.934...
vals	[]

Execute line: Press Step Step again anad again till error line



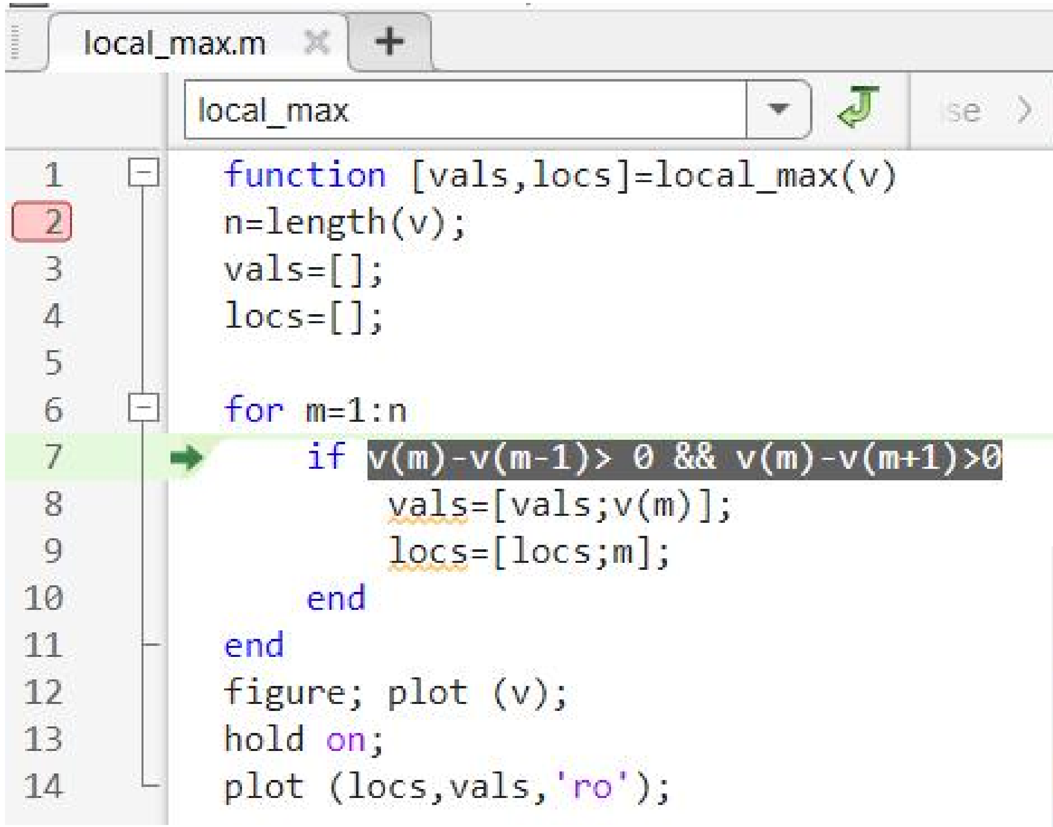
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Workspace is declared with n values



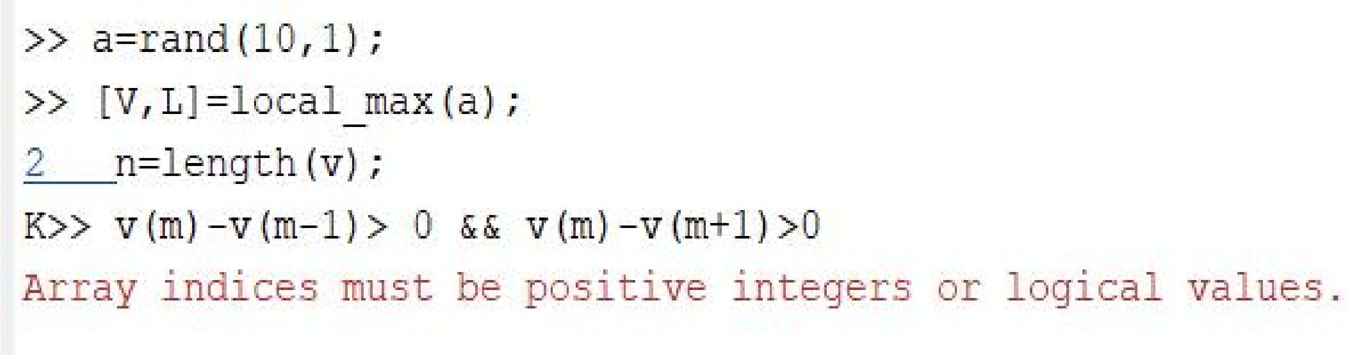
Name	Value
locs	[ ]
m	1
n	10
v	[0.6557;0.0357;0.8491;0.934...
vals	[ ]

Error line



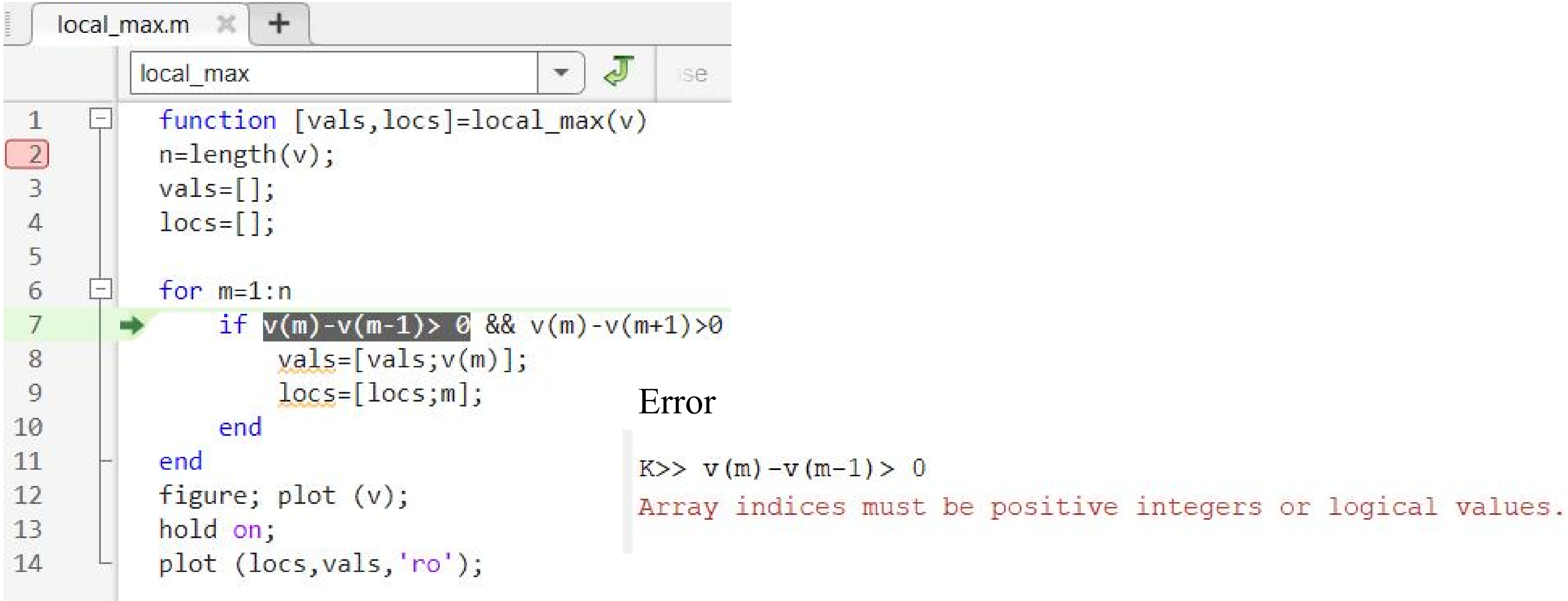
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Error



```
>> a=rand(10,1);
>> [V,L]=local_max(a);
2   n=length(v);
K>> v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
Array indices must be positive integers or logical values.
```

Breaking the Error line furthur

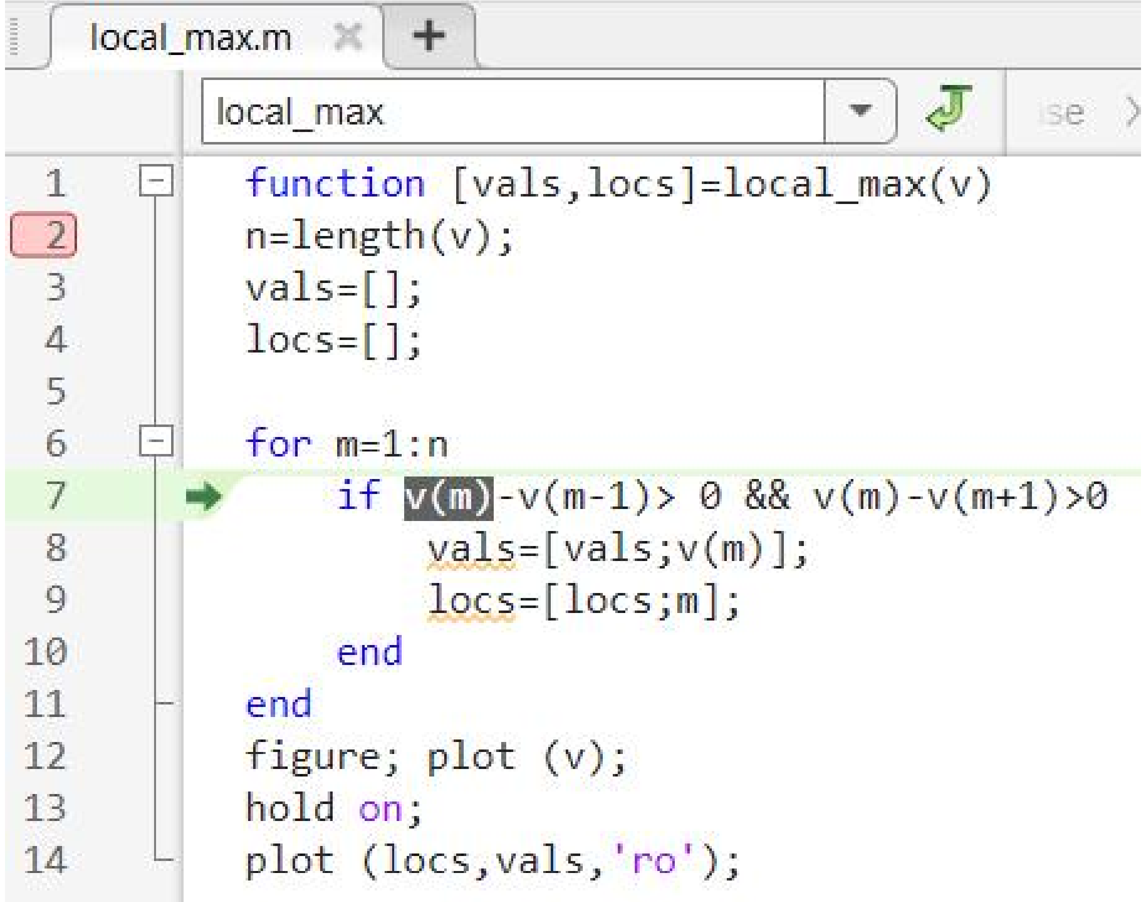


```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Error

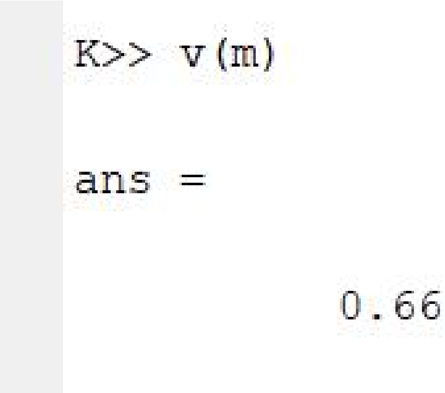
```
K>> v(m)-v(m-1)> 0
Array indices must be positive integers or logical values.
```

Checking individual terms



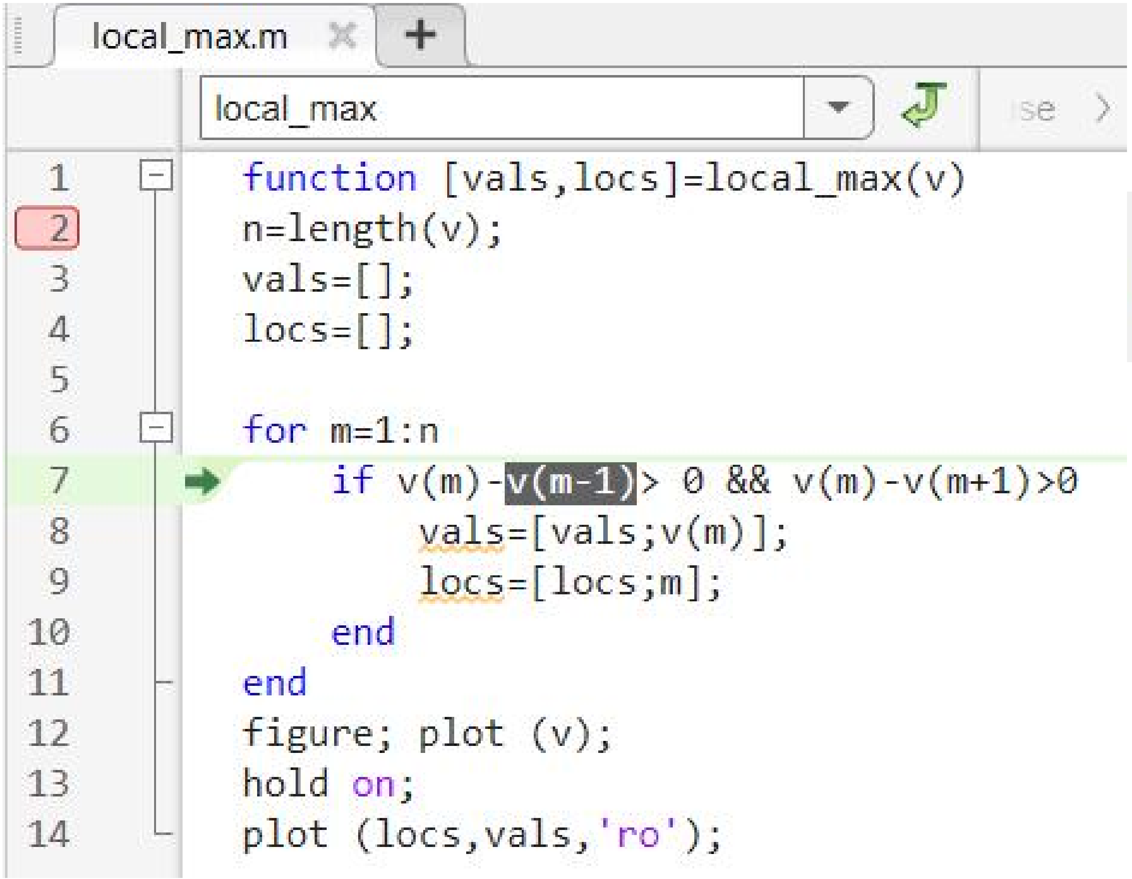
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=1:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

Run:



```
K>> v(m)

ans =

          0.66
```

Checking individual terms



```
1   function [vals,locs]=local_max(v)
2   n=length(v);
3   vals=[];
4   locs=[];
5
6   for m=1:n
7       if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
8           vals=[vals;v(m)];
9           locs=[locs;m];
10      end
11  end
12  figure; plot (v);
13  hold on;
14  plot (locs,vals,'ro');
```

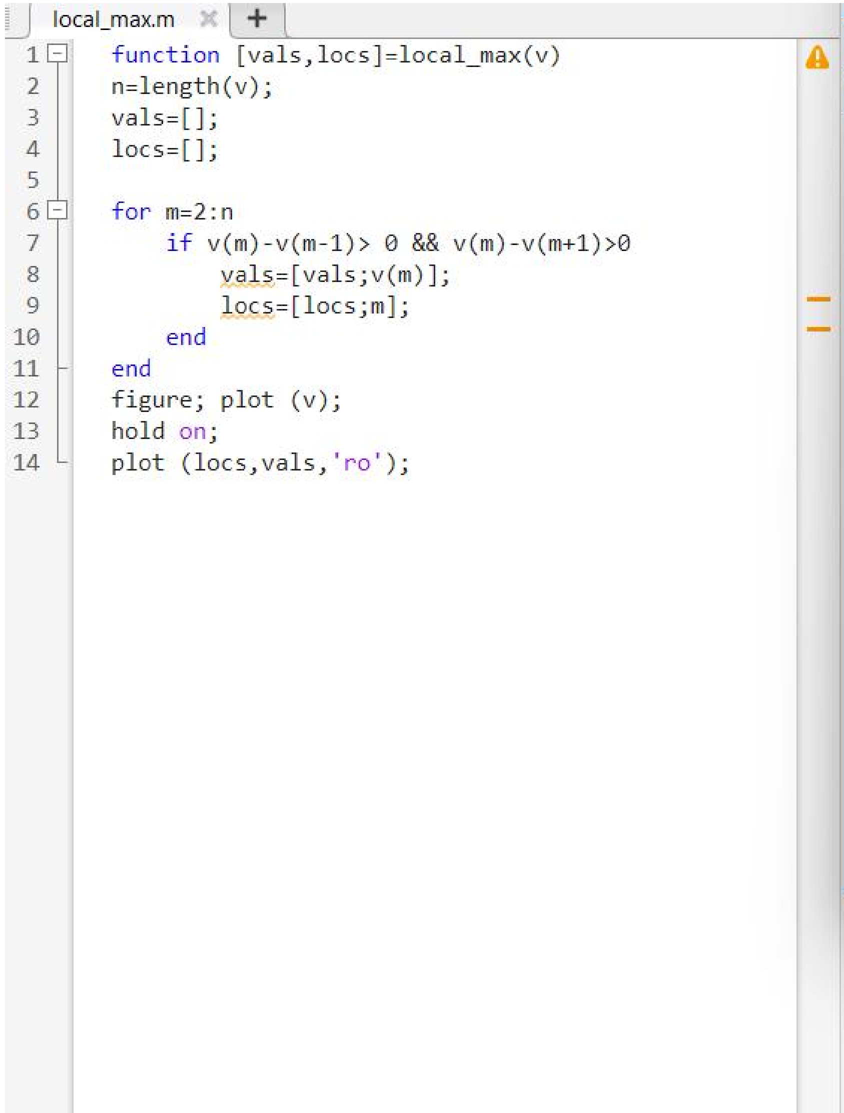
Run:

```
K>> v(m-1)
Array indices must be positive integers or logical values.
```

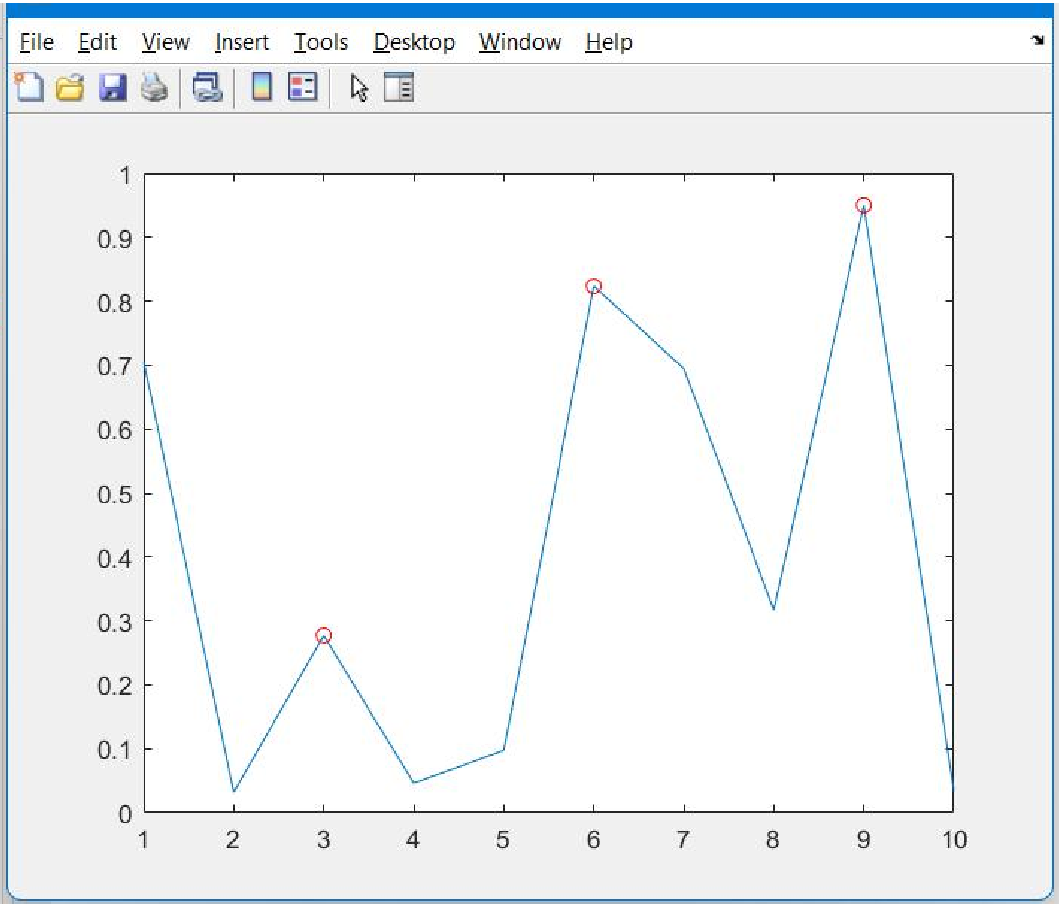
sice, value of m in line 6 starts from 1. so v(m-1) will be 1-1=0.
Thus there is an error and need to be corrected.

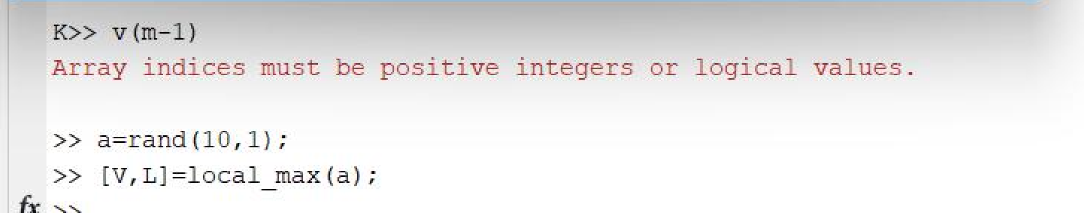
Correct the code

Quit debugger here, come to the normal .m file, remove breakpoint and run the code



```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

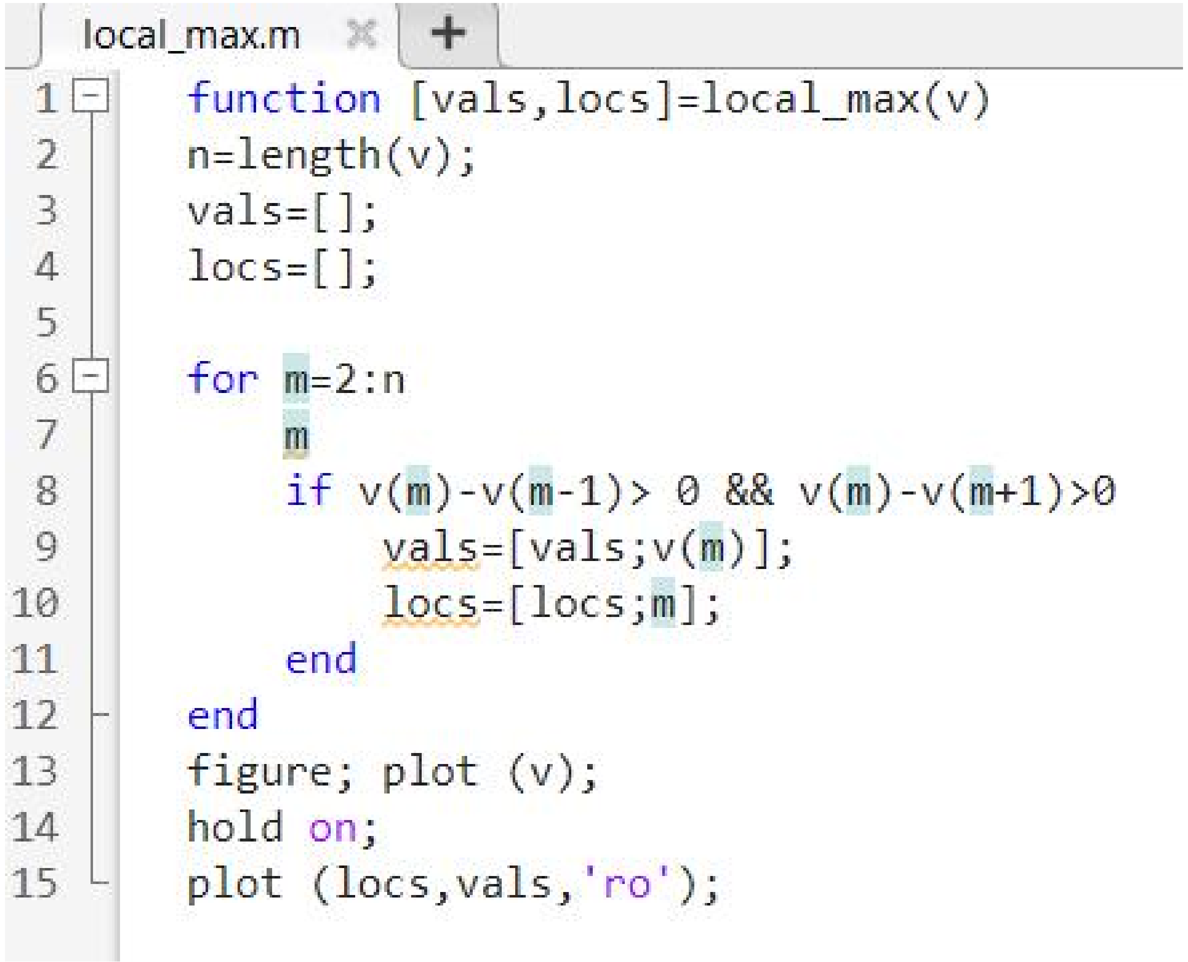




```
K>> v(m-1)
Array indices must be positive integers or logical values.

>> a=rand(10,1);
>> [V,L]=local_max(a);
fx >>
```

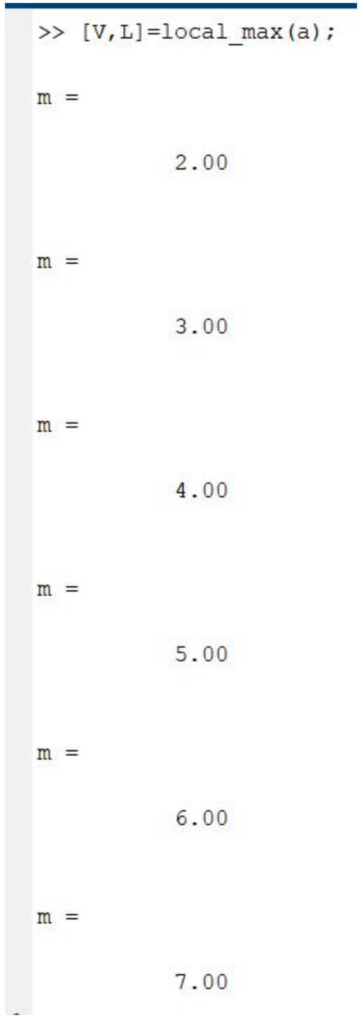
Check the values

- write m under for loop
- quit debugger
- stop breakpoint
- save the file
- execute the code

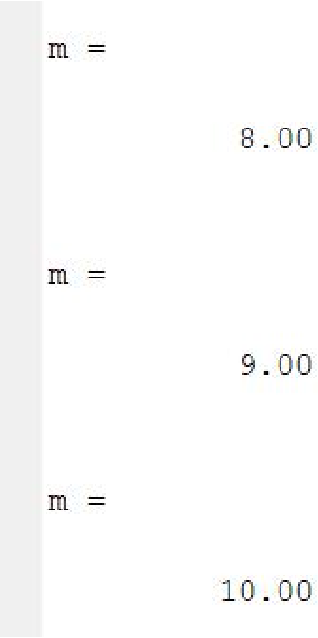


```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n
    m
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

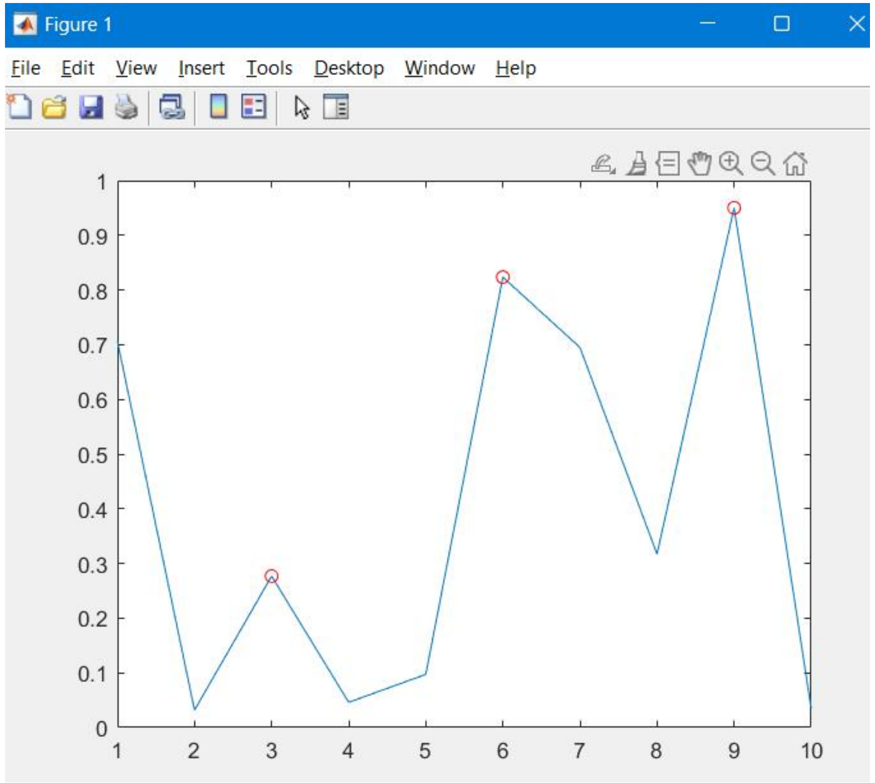
Execute the code



```
>> [V,L]=local_max(a);

m =

        2.00


m =

        3.00


m =

        4.00


m =

        5.00


m =

        6.00


m =

        7.00
```

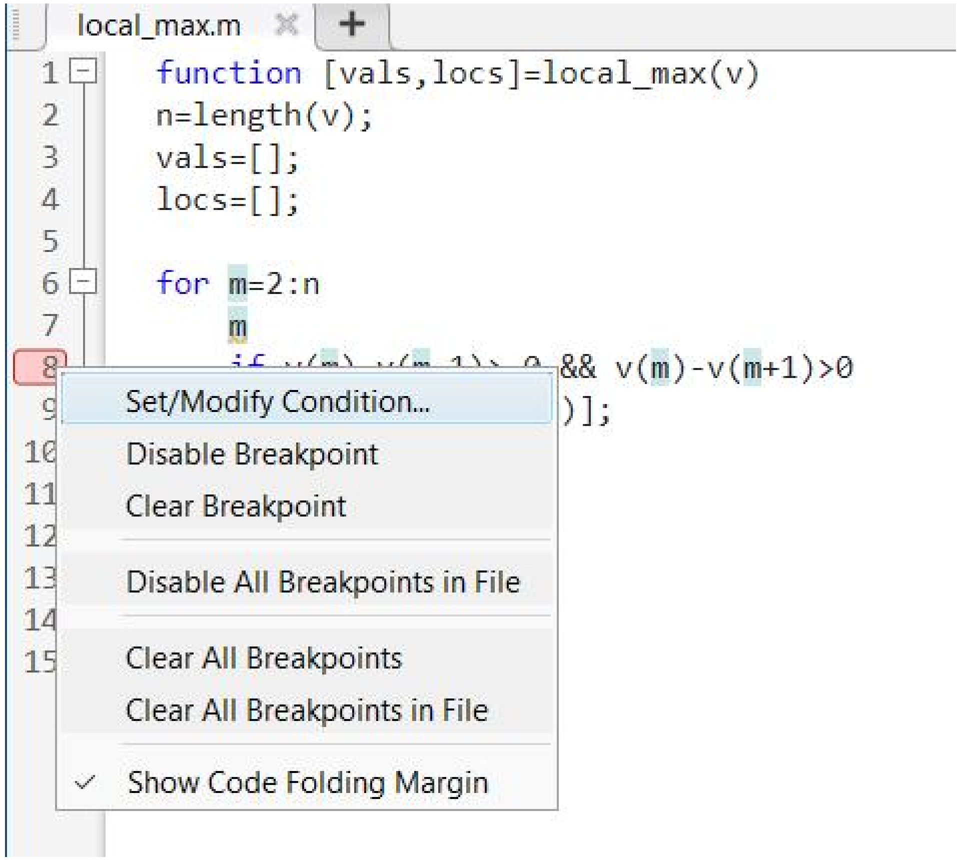


```
m =

        8.00


m =

        9.00


m =

       10.00
```

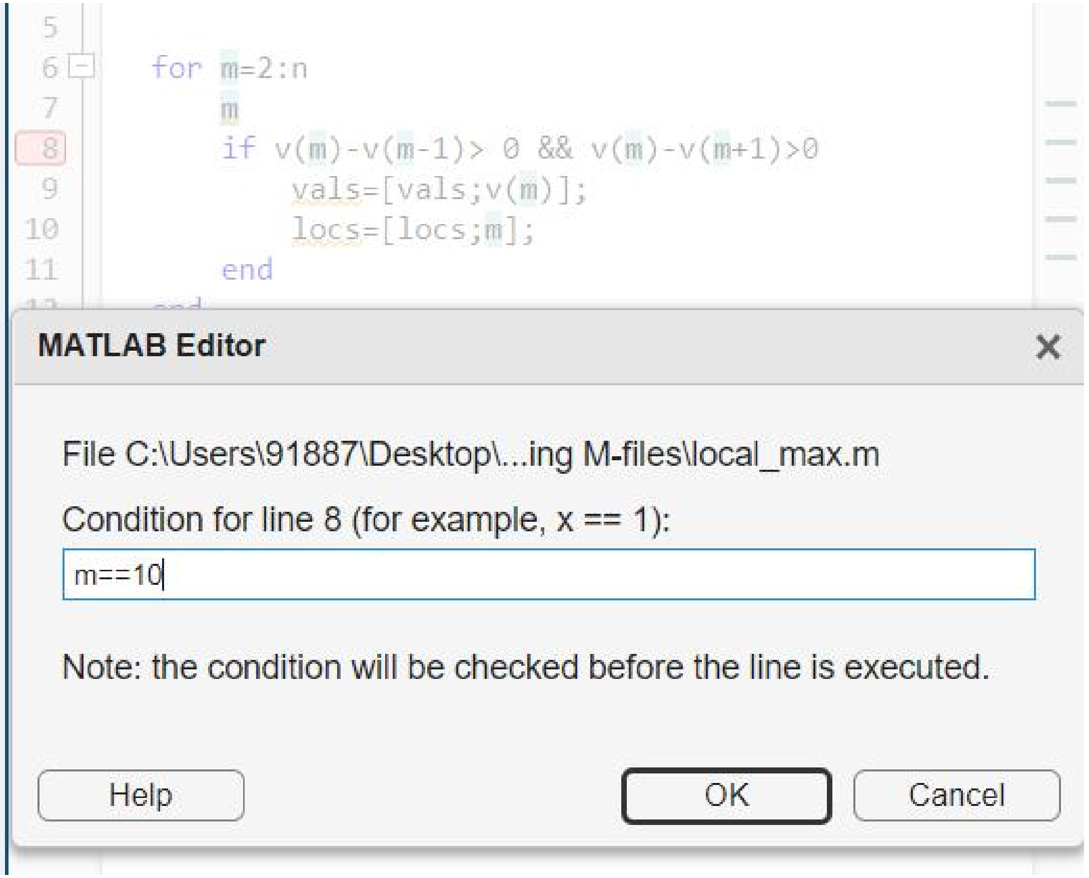


Code is executed properly, is there any problem?

Set a condition breakpoint



```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n
    m
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        )];
```



```
for m=2:n
    m
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
```

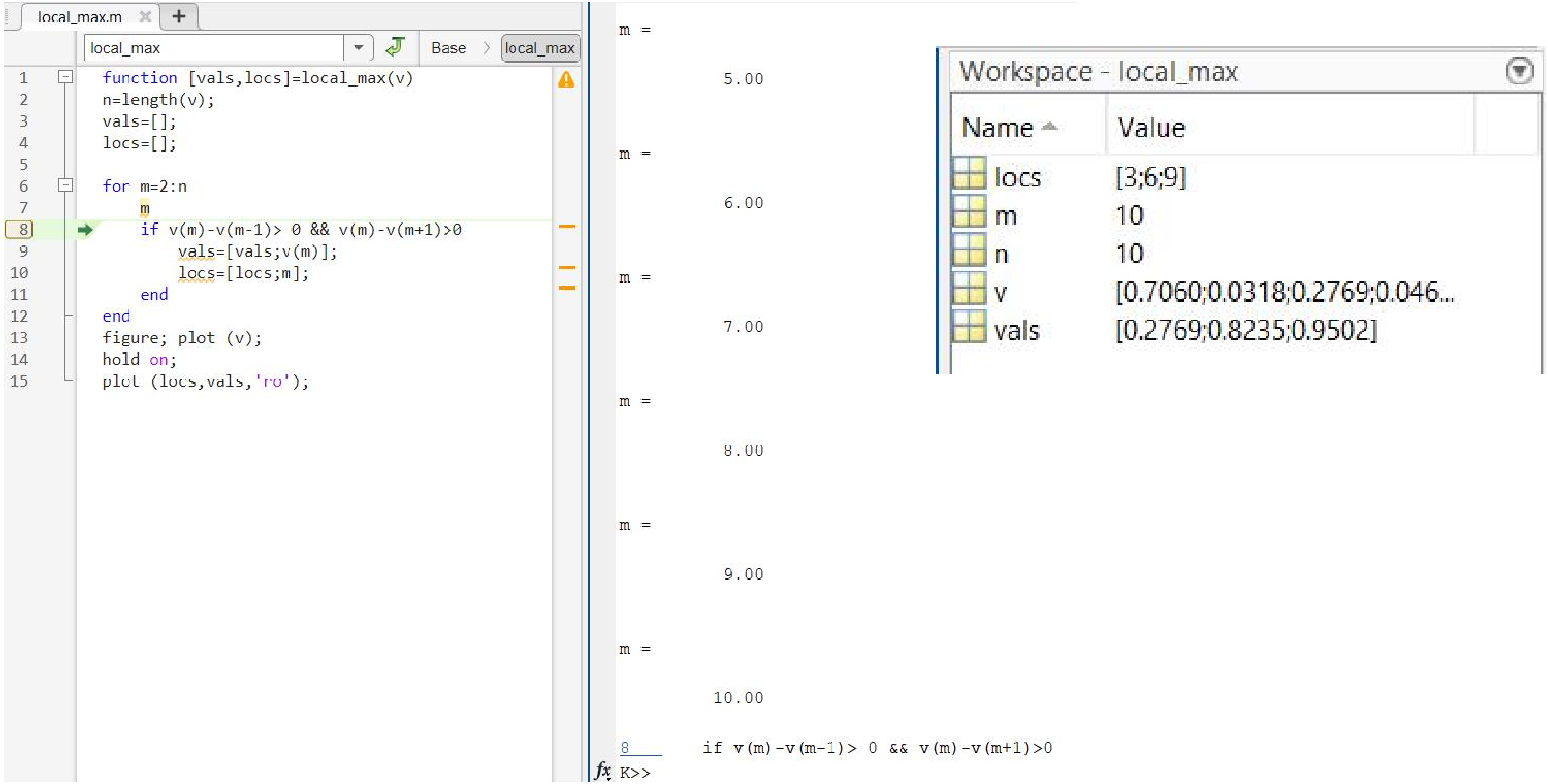
File C:\Users\91887\Desktop\...ing M-files\local_max.m

Condition for line 8 (for example, x == 1):

m==10

Note: the condition will be checked before the line is executed.

Execute file



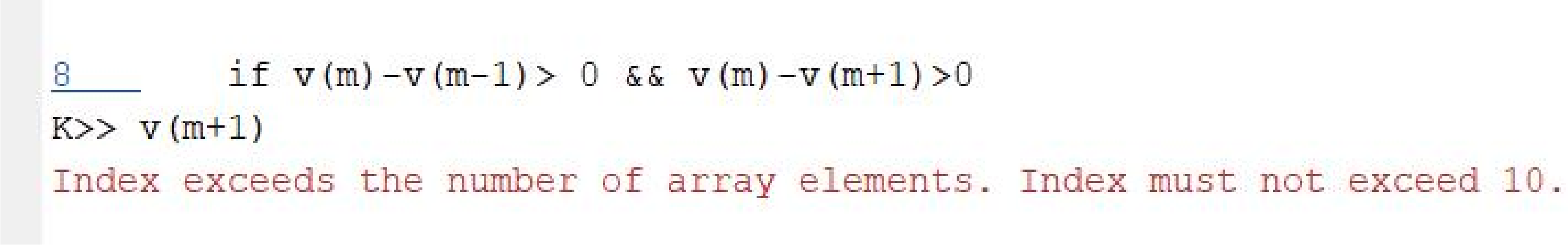
```matlab
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n
    m
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

```
m =

         5.00


m =

         6.00


m =

         7.00


m =

         8.00


m =

         9.00


m =

        10.00

8       if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
K>>
```

Workspace - local_max

Name	Value
locs	[3;6;9]
m	10
n	10
v	[0.7060;0.0318;0.2769;0.046...
vals	[0.2769;0.8235;0.9502]

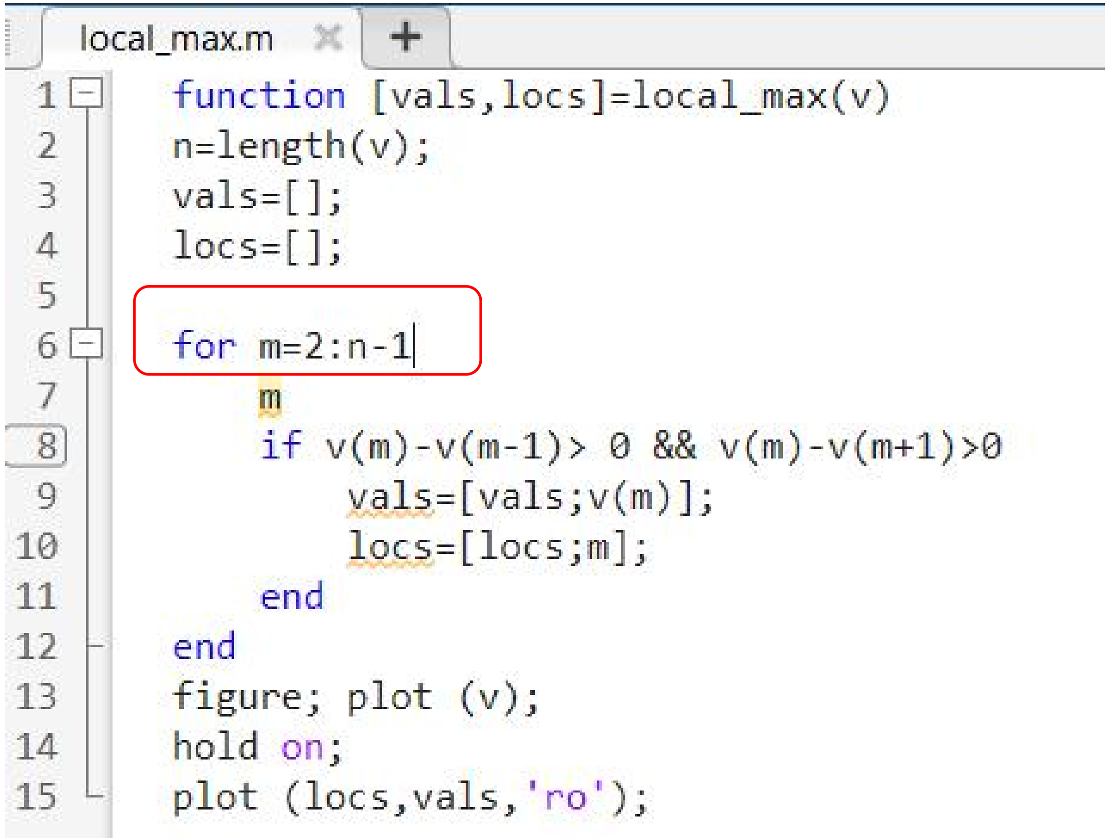
Error found in line 8 again



```
8          if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
K>> v(m+1)
Index exceeds the number of array elements. Index must not exceed 10.
```

Now correct the code at line 6, the new code will be

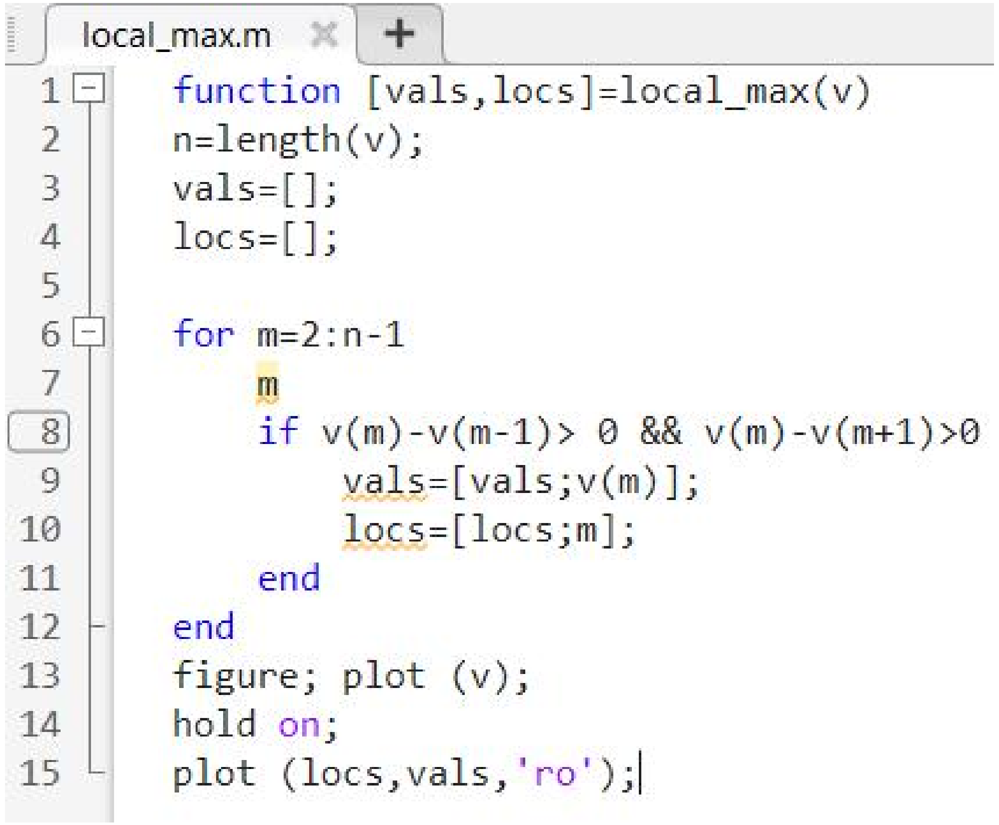
New code



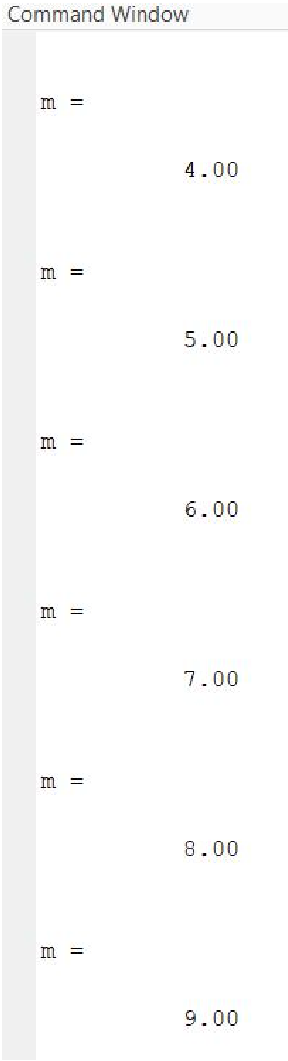
```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n-1
    m
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```

- quit debugger
- stop breakpoint
- save the file
- execute the code

Final code and result



```
function [vals,locs]=local_max(v)
n=length(v);
vals=[];
locs=[];

for m=2:n-1
    m
    if v(m)-v(m-1)> 0 && v(m)-v(m+1)>0
        vals=[vals;v(m)];
        locs=[locs;m];
    end
end
figure; plot (v);
hold on;
plot (locs,vals,'ro');
```



```

m =

          4.00


m =

          5.00


m =

          6.00


m =

          7.00


m =

          8.00


m =

          9.00
```

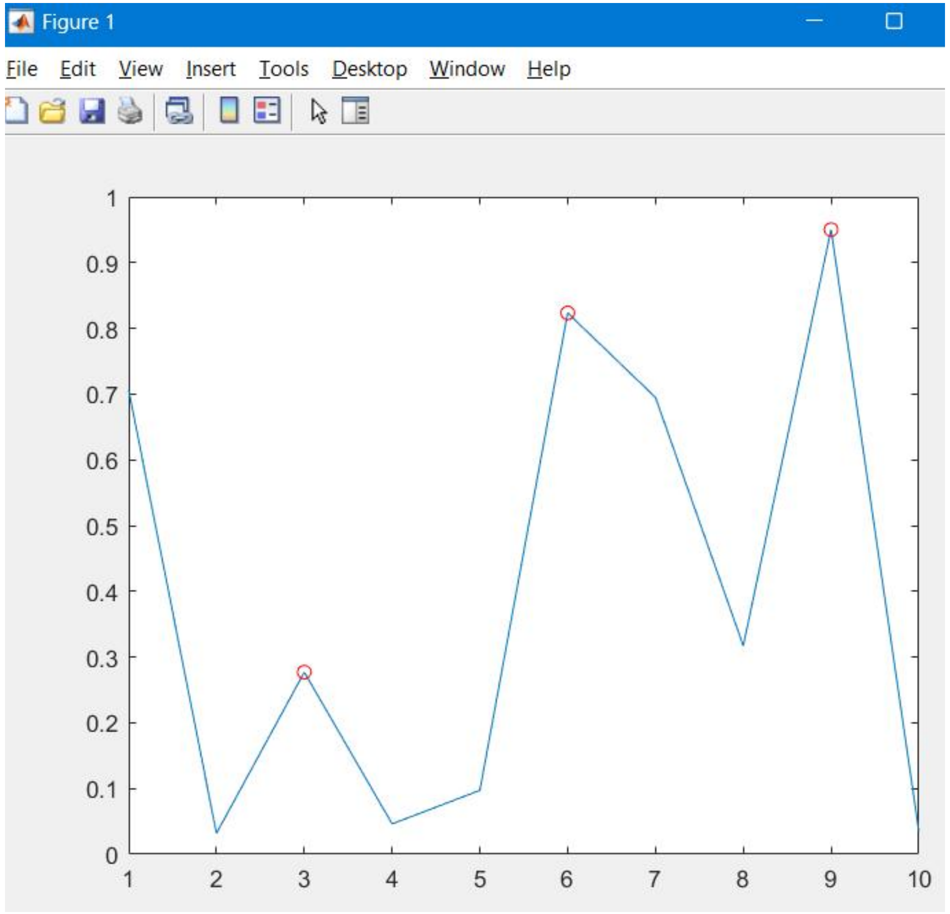


Figure 1
File Edit View Insert Tools Desktop Window Help