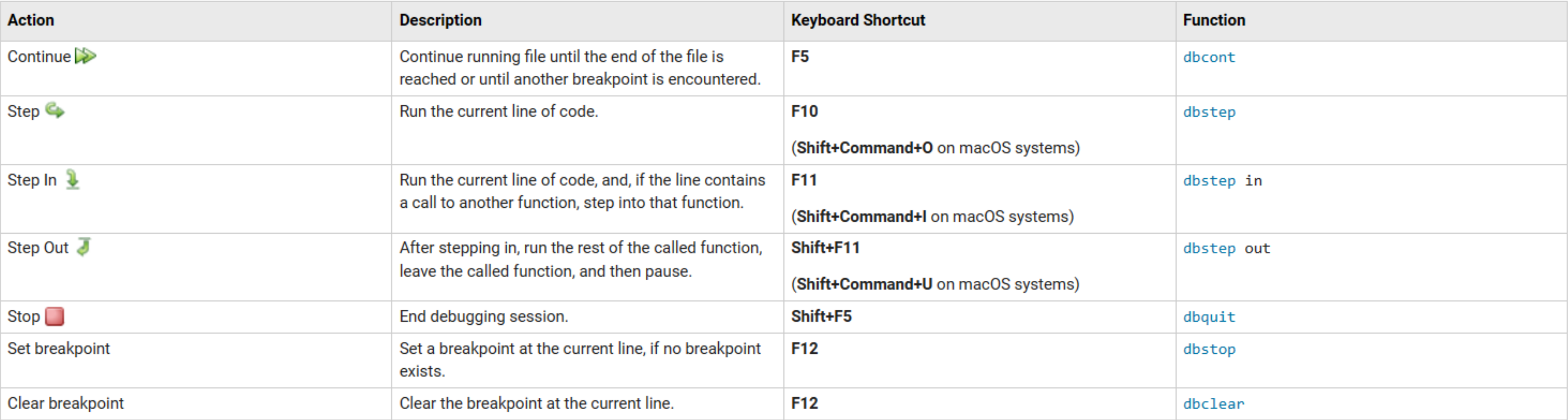
MATLAB Tutorial 05

ENME 303 Computational Methods for Engineers

Parham Oveissi

MATLAB Debugging Tools

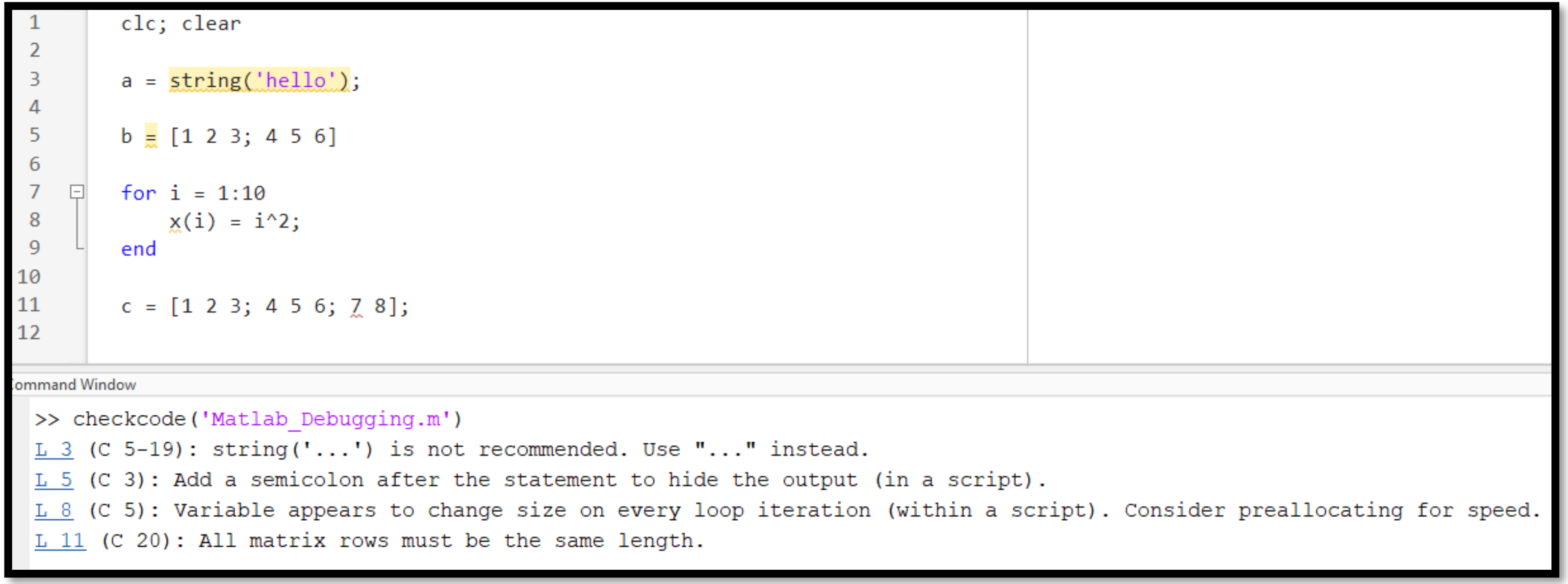


Action	Description	Keyboard Shortcut	Function
Continue	Continue running file until the end of the file is reached or until another breakpoint is encountered.	**F5**	`dbcont`
Step	Run the current line of code.	**F10** (**Shift+Command+O** on macOS systems)	`dbstep`
Step In	Run the current line of code, and, if the line contains a call to another function, step into that function.	**F11** (**Shift+Command+I** on macOS systems)	`dbstep in`
Step Out	After stepping in, run the rest of the called function, leave the called function, and then pause.	**Shift+F11** (**Shift+Command+U** on macOS systems)	`dbstep out`
Stop	End debugging session.	**Shift+F5**	`dbquit`
Set breakpoint	Set a breakpoint at the current line, if no breakpoint exists.	**F12**	`dbstop`
Clear breakpoint	Clear the breakpoint at the current line.	**F12**	`dbclear`

Useful Functions for Debugging Code

- checkcode(“file_name”)
- issues = codeIssues (“file_name”)
  - Since R2022b
- [status, results] = fix(issues, checkID)
  - Since R2023a

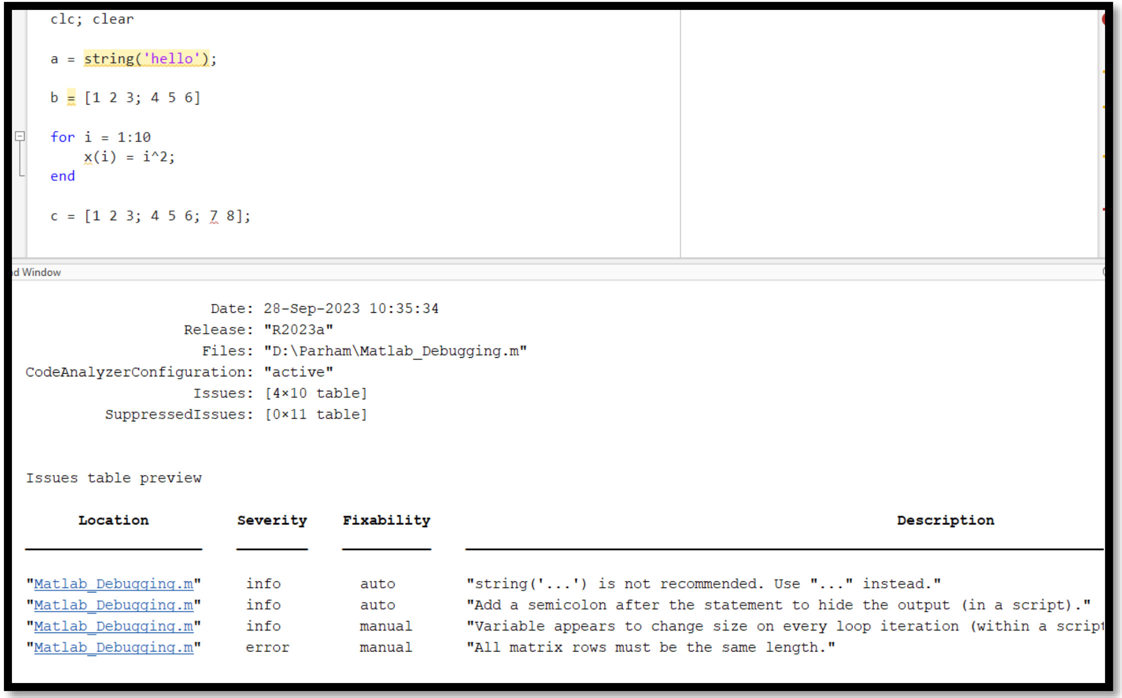
“checkcode” Function



```
1   clc; clear
2   
3   a = string('hello');
4   
5   b = [1 2 3; 4 5 6]
6   
7   for i = 1:10
8       x(i) = i^2;
9   end
10  
11  c = [1 2 3; 4 5 6; 7 8];
12  
```

```
>> checkcode('Matlab_Debugging.m')
L 3 (C 5-19): string('...') is not recommended. Use "..." instead.
L 5 (C 3): Add a semicolon after the statement to hide the output (in a script).
L 8 (C 5): Variable appears to change size on every loop iteration (within a script). Consider preallocating for speed.
L 11 (C 20): All matrix rows must be the same length.
```

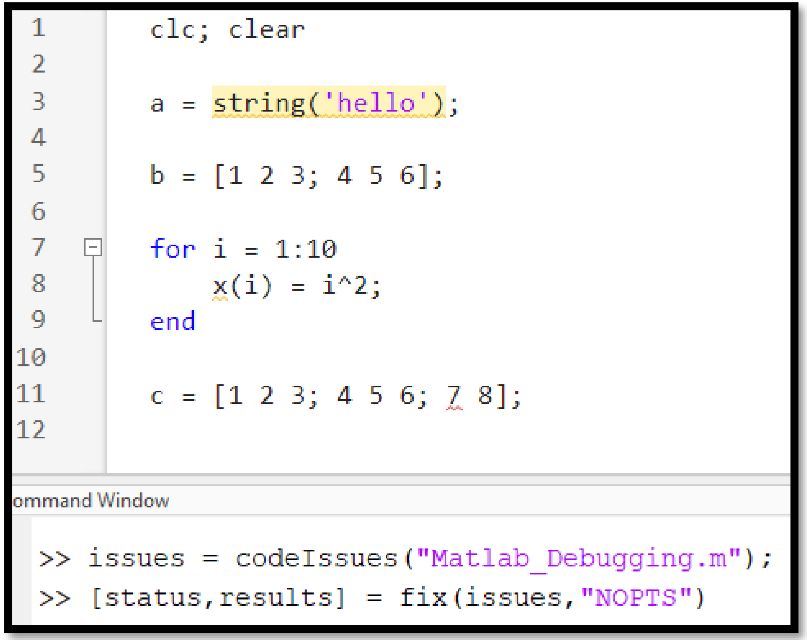
“codeIssues” Function



```
clc; clear

a = string('hello');

b = [1 2 3; 4 5 6]

for i = 1:10
    x(i) = i^2;
end

c = [1 2 3; 4 5 6; 7 8];
```

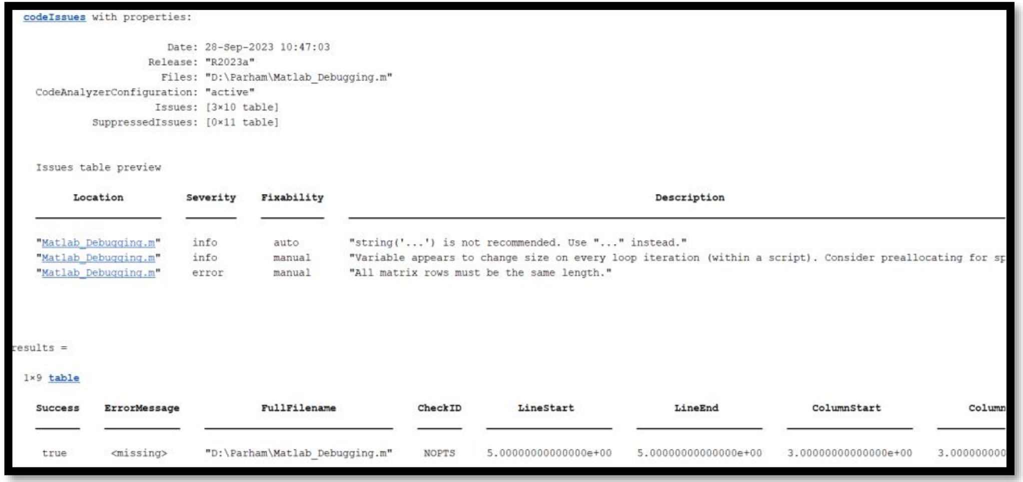
```
                     Date: 28-Sep-2023 10:35:34
                  Release: "R2023a"
                    Files: "D:\Parham\Matlab_Debugging.m"
CodeAnalyzerConfiguration: "active"
                   Issues: [4×10 table]
         SuppressedIssues: [0×11 table]


Issues table preview

      Location             Severity    Fixability                                  Description
____________________       ________    __________    ________________________________________________________________________

"Matlab_Debugging.m"         info        auto        "string('...') is not recommended. Use "..." instead."
"Matlab_Debugging.m"         info        auto        "Add a semicolon after the statement to hide the output (in a script)."
"Matlab_Debugging.m"         info        manual      "Variable appears to change size on every loop iteration (within a scrip
"Matlab_Debugging.m"         error       manual      "All matrix rows must be the same length."
```

“fix” Function



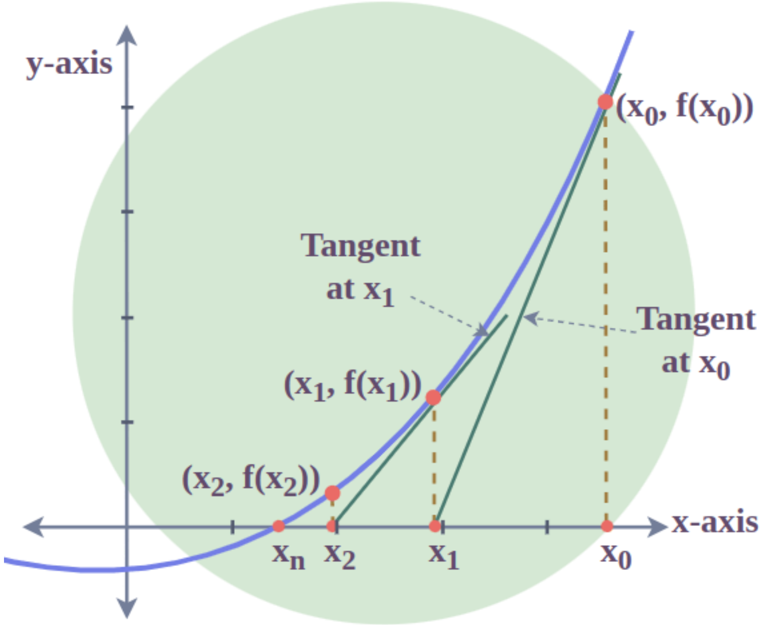
```
clc; clear

a = string('hello');

b = [1 2 3; 4 5 6];

for i = 1:10
    x(i) = i^2;
end

c = [1 2 3; 4 5 6; 7 8];
```

```
>> issues = codeIssues("Matlab_Debugging.m");
>> [status,results] = fix(issues,"NOPTS")
```



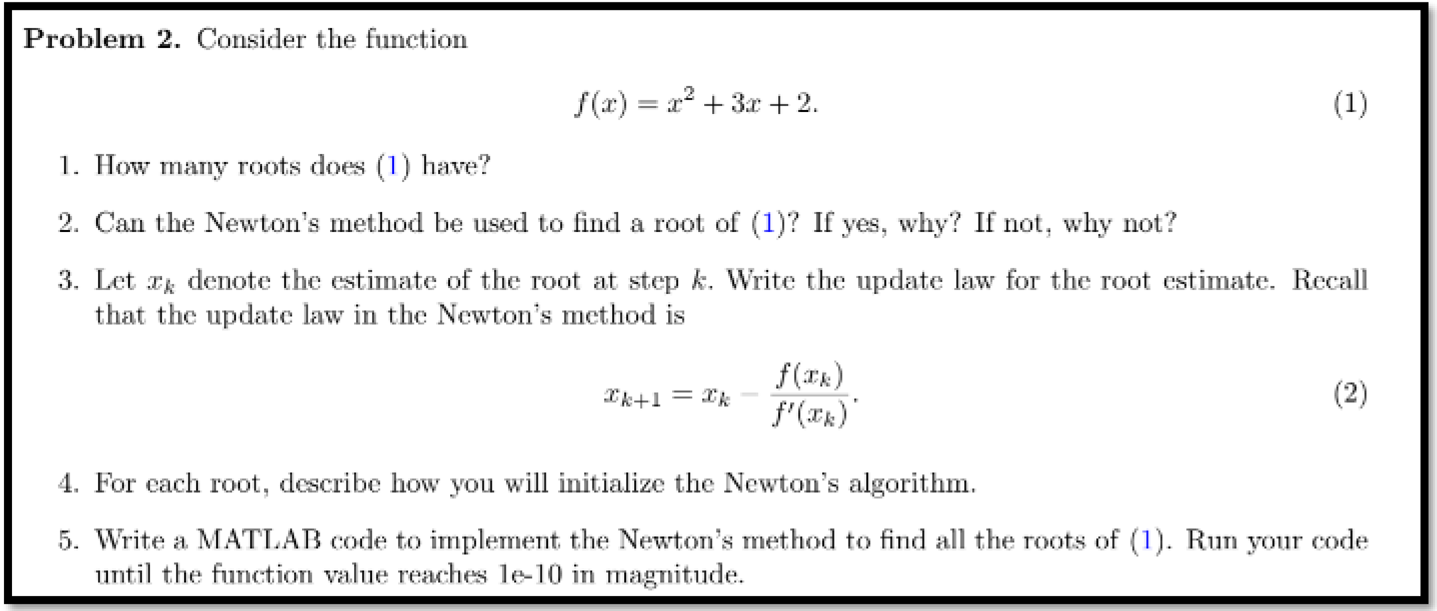
```
codeIssues with properties:

                    Date: 28-Sep-2023 10:47:03
                 Release: "R2023a"
                   Files: "D:\Parham\Matlab_Debugging.m"
 CodeAnalyzerConfiguration: "active"
                  Issues: [3×10 table]
          SuppressedIssues: [0×11 table]


 Issues table preview

       Location          Severity   Fixability                                  Description
 ____________________    ________   __________    ______________________________________________________________________________

 "Matlab_Debugging.m"     info        auto        "string('...') is not recommended. Use "..." instead."
 "Matlab_Debugging.m"     info        manual      "Variable appears to change size on every loop iteration (within a script). Consider preallocating for sp
 "Matlab_Debugging.m"     error       manual      "All matrix rows must be the same length."


esults =

  1×9 table

 Success    ErrorMessage            FullFilename            CheckID         LineStart               LineEnd              ColumnStart          Column
 _______    ____________    ______________________________   _______    ____________________   ____________________   ____________________   _______

  true       <missing>     "D:\Parham\Matlab_Debugging.m"    NOPTS     5.00000000000000e+00   5.00000000000000e+00   3.00000000000000e+00   3.000000000
```

Newton’s Method



$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Implementing Newton’s Method



Problem 2. Consider the function

$$f(x) = x^2 + 3x + 2. \tag{1}$$

1. How many roots does (1) have?
2. Can the Newton's method be used to find a root of (1)? If yes, why? If not, why not?
3. Let x_k denote the estimate of the root at step k. Write the update law for the root estimate. Recall that the update law in the Newton's method is

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{2}$$

4. For each root, describe how you will initialize the Newton's algorithm.
5. Write a MATLAB code to implement the Newton's method to find all the roots of (1). Run your code until the function value reaches 1e-10 in magnitude.

Thanks!