CSE 224
Lab 03

Prepared By: Ridwan Mahbub, Lecturer, CSE, BUBT

Description:
Welcome to Lab-3! In this lab, we will continue our journey learning more about using Octave. We will look into using functions and function files with Octave. We will also look into using loops and conditional statements.

Review:

- Declaring Variables:
  (undeclared operations will be saved to a variable named **ans**)

```
>> var_1=5
var_1 =  5
>> var_2=5+6
var_2 =  11
>> var_2/5
ans =  2.2000
>>
```

- Operators: addition, subtraction and multiplication works normally.

  For division, 6/2 and 2\6 both valid, dividing 6 by 2 in both cases.

```
>> 6/2
ans =  3
>> 2\6
ans =  3
>>
```

- Dotted Operations:

  Adding . before the operator is used to perform element wise operations in case of matrix.

```
>> a=[1 2]

a =

   1   2

>> b=[4 3]

b =

   4   3

>> a.*b

ans =

   4   6
```

Without dot the above operation would not take place as it would be treated as matrix multiplication. Try it yourself.

Same goes for addition, subtraction, division, exponentiation.

- Predefined functions:

  There are some predefined functions sin(), cos() etc.

- Declaring Matrix:

```
>> A= [3 4 ; 5 6]
A =

   3   4
   5   6
```

- Accessing matrix values. **Remember, indexing starts with 1 in octave.**

```
>> A(1,1)
ans = 3
>>
```

- Ranged Array Declaration:

```
>> B=[5:1:15]
B =

    5    6    7    8    9   10   11   12   13   14   15
```

Syntax is **[start:increment:end]**

- **Plotting Graphs:**

  plot(x1, y1, color1 ,x2, y2, color2 …)

  x1= x axis values of graph 1

  y1= y axis values of graph 2

  …

  color1: single letter in single quotes that specify color of graph

  r- Red

  g- Green

  k- Black

Lab Content:

- Using function files:

randomscript.m | functionDemo.m

```
function [output1, output2]=functionDemo(in1, in2)
  output1=in1+in2;
  output2=in1-in2;
endfunction
```

In the command window:

```
>> [a,b]=functionDemo(4,3)
a = 7
b = 1
```

- Logical Operator:

0- False
1- True

==, ~=(not equal), <=, >=, <, >, |, &

- Decision Statements

```
function result=functionDemo(in1, in2)
  if in1 > in2
    result=in1;
  else
    result=in2;
  endif
endfunction
```

```
>> functionDemo(4,3)
ans = 4
>>
```

- Loops:

for loop

forLoop.m

```
1 for k=1:1:10
2   k
3 endfor
```

Command Window

```
>> forLoop
k = 1
k = 2
k = 3
k = 4
k = 5
k = 6
k = 7
k = 8
k = 9
k = 10
```

while loop

whileLoop.m

```
1  k=1
2  while k<10
3    k
4    k=k+1;
5  endwhile
```

```
>> whileLoop
k = 1
k = 1
k = 2
k = 3
k = 4
k = 5
k = 6
k = 7
k = 8
k = 9
>>
```

Tasks:

1. Define a function that generates value for the function $f(x)=x^2- 8x + 15$.

```
function fx=functionDemo(x)
  fx =
endfunction
```

2. Take an array from 1 to 10 with an interval of .1 and plot the value for the function and add a grid.
3. Now find one of the roots using a simplified bisection method shown below: