Mathematics for Computing (IT1030)

2020 – Semester 1

Lab Sheet – 01

Introduction to Octave

Octave is an open-source interactive software system for numerical computations and graphics. It is particularly designed for matrix computations. It can be thought of as a very powerful, programmable, graphical calculator. Octave makes it easy to solve a wide range of numerical problems.

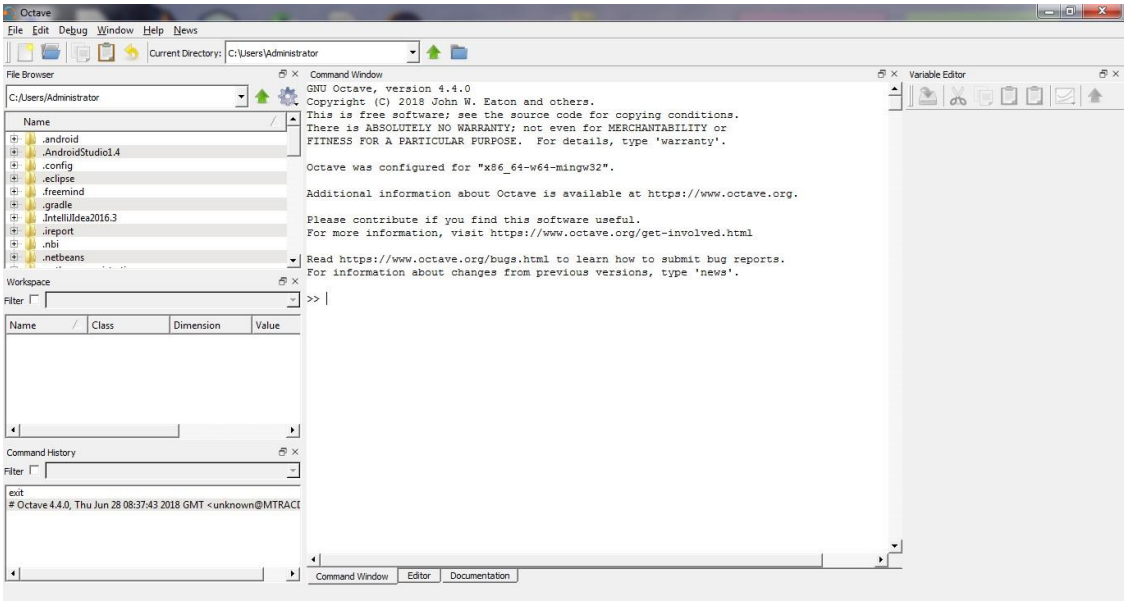
We will be using GNU Octave, version 4.4.0 which can be downloaded from the following url.
https://www.gnu.org/software/octave/download.html

Starting Octave

After logging into your account, you can open Octave by double-clicking on the GNU Octave (GUI) shortcut icon on your Windows desktop. You are now in the Octave environment. The >> is the Octave prompt, asking you to type in a command. If you want to leave Octave at any point, type `quit` at the prompt.

Components of the Octave Window

- The Command Window
- The Command History
- The Workspace
- The Current Directory
- Variable Editor



Exercises

Explore the following options in Octave.

1. Using Octave as a calculator.

   The simplest way to use Octave is just to type mathematical commands at the prompt, like a normal calculator. The basic arithmetic operators are + - * /, and ^ is used to mean ‘to the power of’ (e.g. 2^3=8). Brackets ( ) can also be used.

```
>> 2+2
ans =  4
```

2. Creating variables.

   In any significant calculation you are going to want to store your answers, or reuse values, just like using the memory on a calculator. Octave allows you to define and use named variables. variables are either floating point numbers or strings.

```
>> a = 5
a =  5
>> x = 3+4*(1+6/3)
x =  15

>> str = "example string"
str = example string
```

   If the command is ended with a semicolon, the output will be suppressed.

```
>> x = 3+4*(1+6/3);
```

The value of x may now, if desired, be printed out by typing

```
>> x
x =  15
```

You can create variables using previously defined variables.

```
>> f = x+2
f =  17
```

Octave is case-sensitive. Therefore `X` and `x` will be treated as two different variable names.

```
>> X = 5
X =  5
>> x = 2
x =  2
```

3. Overwriting variables.

   The variable values can be overwritten by re assigning a new value to a previously defined variable.

```
>> y = 2+3
y =  5
>> y = 2+4
y =  6
```

4. Error messages and making corrections.

   Octave reports errors for invalid commands.

```
>> y = 2+4
y =  6
>> f = 2y
parse error:

  syntax error

>>> f = 2y
        ^
```

5. Operator precedence.

   Operator precedence determines how operators are grouped, when different operators appear close by in one expression. For example, '*' has higher precedence than '+'. Thus, the expression a + b * c means to multiply b and c, and then add a to the product (i.e., a + (b * c)).

   You can overrule the precedence of the operators by using parentheses.

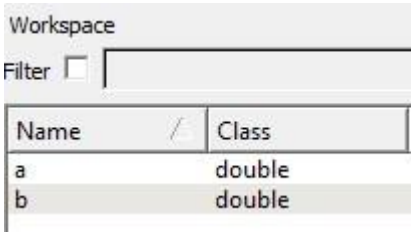
```
>> a=5;
>> b=2;
>> c=3;
>> a+b*c
ans =  11
>> (a+b)*c
ans =  21
```

6. Managing the workspace.

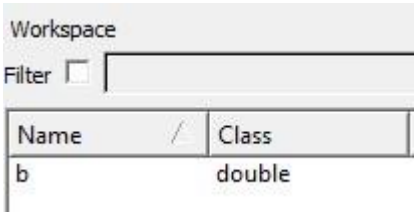
   a. `clear` and `clear all` functions.

Clear function can be used to remove a variable from the workspace.

```
>> a = 5;
>> b = 10;
```




```
>> clear a
```



Clear and clear all functions can be used to remove all the variables from the workspace.

b. `who` and `whos` functions.

When creating programs it can be very convenient to see which variables are available at the prompt. The function *who* and *whos* will show different information about what is in memory.

```
>> a=5;
>> b=6;
>> c=4;
>> who
Variables in the current scope:

a  b  c

>> whos
Variables in the current scope:

   Attr Name        Size                     Bytes  Class
   ==== ====        ====                     =====  =====
        a           1x1                          8  double
        b           1x1                          8  double
        c           1x1                          8  double

Total is 3 elements using 24 bytes
```

7. Saving the work session using the `diary` command.

Octave's diary feature allows you to keep a log of all or part of a session by recording the input you type and the output that Octave produces in a separate file.

```
>> diary '1.txt';
>> diary on
>> 2+4
ans =  6
>> x = 5*5
x =  25
>> diary off
```

8. Other commands in Octave.

a. Clear the Command Window.

```
>> clc
```

b. Abort a computation.

You can stop a running command or script by pressing ctrl + c.

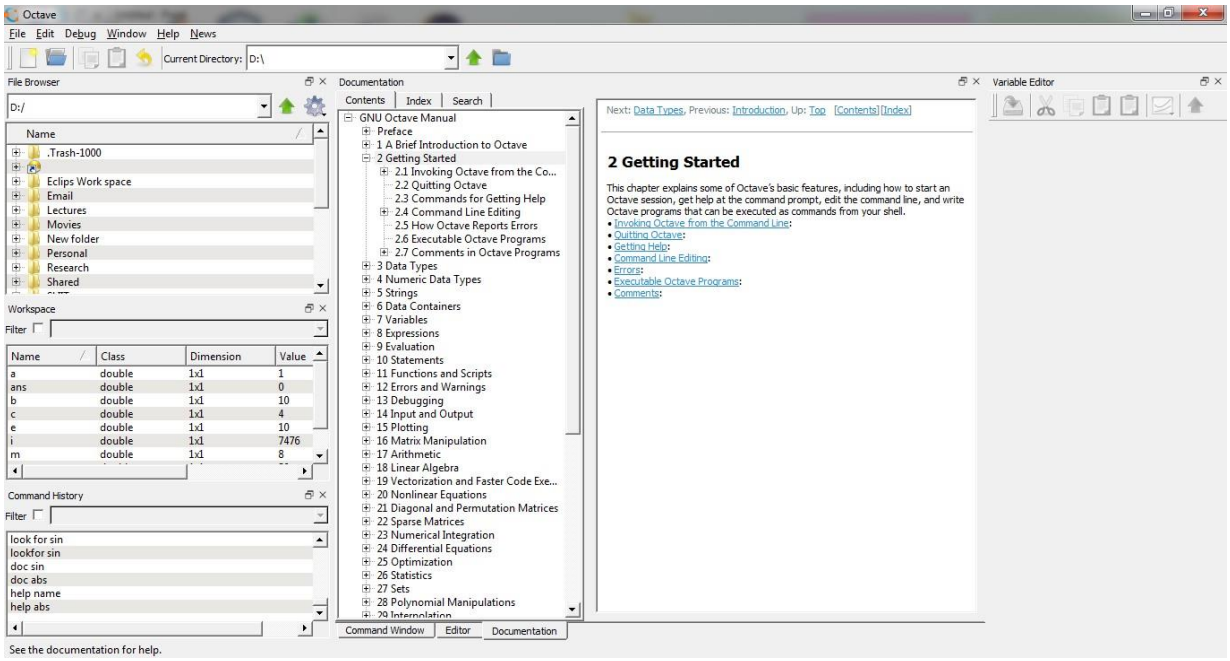
c. Continuing a line.

You can continue a line using three dots.

```
>> s = 1+2+3+4+5+6+...
4+5
s =  30
```

9. Getting help

You can look at the documentation of Octave using the documentation tab.



You can look at the documentation of a known function using the command `doc`.

```
>> doc abs
```

The `lookfor` command can be used to search for a string in the documentation of all functions.

```
>> lookfor abs
abs                    Compute the magnitude of Z.
file_in_loadpath       Return the absolute name of FILE if it can be found in the list of
                       directories specified by â€˜pathâ€™.
file_in_path           Return the absolute name of FILE if it can be found in PATH.
is_absolute_filename Return true if FILE is an absolute filename.
make_absolute_filename Return the full name of FILE beginning from the root of the fil
                       e system.
dir_in_loadpath        Return the absolute name of the loadpath element matching DIR if i
                       t can be found in the list of directories specified by â€˜pathâ€™.
                       .
mad                    Compute the mean or median absolute deviation of the elements of X
```