

LAB 3

Fundamental Concepts

Objective

The aim of this introductory lab is to introduce you to the basic functions in the Matlab and Numerical Methods with Matlab toolbox. By the end of today's lab, you should be able to understand the script and function files.

Submission Requirements

You are expected to complete the assigned tasks within the lab session and show them to the lab engineer/instructor. Some of these tasks are for practice purposes only while others (marked as '*Exercise*' or '*Question*') have to be answered in the form of a lab report that you need to prepare. Following guidelines will be helpful to you in carrying out the tasks and preparing the lab report.

Guidelines

- In the exercises, you have to put the output in your Lab report. You may add screen print to the report by using the 'Print Screen' command on your keyboard to get a snapshot of the displayed output. This point will become clear to you once you actually carry out the assigned tasks.
- Name your reports using the following convention:
Lab#_Rank_YourFullName
 - '*#*' replaces the lab number
 - '*Rank*' replaces Maj/Capt/TC/NC/PC
 - '*YourFullName*' replaces your complete name.
- You need to submit the report even if you have demonstrated the exercises to the lab engineer/instructor or shown them the lab report during the lab session.

M-Files

- Files that contain a MATLAB code are called the *m-files*. There are two kinds of m-files:
 - script files*

Script files is a sequence of MATLAB statements and do not take the input arguments or return the output arguments
 - function files*

The function files may take input arguments or return output arguments. Function name should be same as the name of the file without .m extension
- To make the m-file click on File next select New and click on M-File
- To open the m-file from within the Command Window type `>>edit filename`

Script Files

- A Script file is a sequence of Matlab command also call a program.
- When a script file runs, Matlab executes the commands in the order they are written.
- When a script file displayed the output in command window.
- It is convenient way to edit and execute command many times.
- Script file can be typed and edited in any text editor and saved as .m file.
- Script files is also called M-files because the extension .m is used.

Search path

- When Matlab is asked to run a script file or to execute a function, it searches in directories that are listed in the search path.
- Set path window displayed search path that can be set by selecting **Set Path** in **File** menu.
- New folders can be added to, or removed from, the search path in set path window

The *input* command

The input command is used to prompt user to assign a value to variable in command window.

`variable_name=input('string with a message that displayed in command windows')`

The string is a message prompting the user to enter a value that is assigned to variable.

The *disp* command

The input command is used to display the elements of a variable and to display text.

`disp(name of the variable)` or `disp('text as string')`

Every time the disp command is executed, the display it at new line.

Practice 1

```
>>abc=[5 9 1; 7 2 4];           %A 2x3 array is assigned to variable abc.
>>disp(abc)
>>disp('The problem has no solution.')
```

%Create a script file with name average.m

```
disp('Calculate the average of points scored in three games');
game1=input('Enter the point scored in the first game ');
game2=input('Enter the point scored in the second game ');
game3=input('Enter the point scored in the third game ');
avg_points=(game1+game2+game3)/3;
disp(' ')                               %Dispaly empty line
disp('The average of points scored in a game is:') %Dispaly text.
disp(' ')                               %Dispaly empty line
disp(avg_points)                        %Display value of the variable avg_points.
>>average
```

%Create a script file with name PopTable.m

```
yr=[1984:2:1996];
pop=[127 130 136 145 158 178 211]
tableYP(:,1)=yr';           %yr is entered as the first column in the array tableYP
tableYP(:,2)=pop';          %pop is entered as the second column in the array tableYP
disp(' YEAR          POPULATION ') %Display heading first line
disp('          (MILLIONS)')       %Display heading second line
disp(' ')                         %Dispaly empty line
disp(tableYP)                   %Display the array tableYP.
>>PopTable
```

The *fprintf* command

- The *fprintf* command can be used to display formatted output (text and data) on screen or to save it to a file.
- Text and numerical values of variables can be intermixed and displayed in the same line.

Using to display text

```
fprintf('text typed in as a string')
```

\n used to start with new line in the middle of string.

\t used to insert horizontal tab in string

Using to dispaly mixed text and numerical data

```
fprintf('text as string %-5.2f additional text', variable_name)
```

- (optional) Left Justified + prints a sign character (+ or -) 0 Adds zeros if no is shorter
- 5.2(optional) Field width and precision (optional)
- f (required) Conversion character (required).

Practice 2

%Create a script file with name TempConv.m

```
disp('Converting table Cel 2 Fah');
start=input('Start temp in Celsius: ');
incr=input('In-crement in Celsius: ');
total=input('No of Entries in table: ');
stop=start+(total-1)*incr;
C=start:incr:stop;
F=9/5*C+32;
disp('Celsius Fahrenheit');
fprintf('%10.2f %10.2f\n', [C;F]);
>>TempConv
```

Function files

- A user defined function is a Matlab program that is created by the user saved as function file, and then can be used like a built in function.
- It must be put in a .m-file with same name as function
- The input and output can be one or serveral variables, and each can be a scalar, vector or an array of any size
- A large program can be made up of smaller “building blocks” that can be tested independently.

Structure of a function file

Practice 3

```
function [mpay, tpay]=loan(amount,rate,years)           %function definition line
%loan calculates monthly and total payment of loan.    %The H1 line
%Input arguments:                                     % Help Text.
%amount=loan amount in $                             “
%rate=annual interest rate in percent.                “
%years=number of years.
%Output arguments:
%mpay=monthly payment, tpay=total payment.
format bank                                           %function body
ratem=rate*0.01/12;
a=1+ratem;
b=(a^(years*12)-1)/ratem;
mpay=amount*a^(years*12)/(a*b);                     %Assignment of values to output arg
tpay=mpay*years*12;
>>[M,T]=loan(50000,3.9,10)                          %function call
```

Function Definition Line

The first executable line in a function file must be the function definition line.

Defines the file as a function file

Defines the name of function

Defines the number and order of the input and output arguments.

```
function [output arguments]=function_name(input arguments)
```

H1 line and Help text

These are comments and information about function

Displayed when a user type *lookfor a_word* Matlab searches in the H1 lines of all the functions and if match is found the H1 line.

Displayed including H1 line when user type *help function_name* Matlab searches all built-in functions as well as the user defined functions.

Function body

Computer program that actually performs the computations.

Inline function

- Used when the value of a simple mathematical function has to be determined many times within a program.
- Defined with inline command, within matlab code and then is used in the code.
- Inline function expression can not include preassigned variables.

```
name=inline('math expression typed as a string')
```

or

```
name=inline('mathematical expression','arg1','arg2','arg3')
```

Practice 4

FtoC.m

Function C = FtoC(F)

%FtoC converts degrees F to degrees C

```
C=5*(F-32)./9;
```

```
>>FC=inline('5*(F-32)./9');
```

```
>>z=FC(70);
```

```
%f(x,y)=2x2-4xy+y2
```

```
>>HA=inline('2*x^2-4*x*y+y^2');
```

```
>>z=HA(2,3)
```

```
z=-7
```

The *feval* Command

feval command evaluates the value of a function for given value of arguments.

Variable=*feval*('function name', arguments value)

The function name typed as string

The function can be a built in or user defined function

More then one arguments are separated by commas.

More then one output are stored in matrix ([a,b,c,])

Practice 5

```
>>feval('sqrt',64)
ans=8
>>feval('FtoC',70);
>>[M,T]=feval('loan',50000,3.9,10)
M=502.22
T=60266.47
```

Exercise 1

The population of a certain country is 50 million and is expected to double in 20 years. Calculate the population 5, 10, and 15 years from now by defining a vector *t* with 3 elements and using element by element calculation. Population growth can be modeled by the equation $P=P_0 2^{t/d}$, where *P* is the population at time *t*, *P*₀ is the population at *t*=0, and *d* is the doubling time.

Exercise 2

Write a script file that determines the balance in a saving account at the end of every year for the first 10 years. The account has an initial investment of 1000 and interest rate of 6.5% that compounds annually. Display the information in a table.

For an initial investment of *A*, and interest rate of *r*, the balance *B* after *n* year is given by:

$$B=A(1+r/100)^n$$

Exercise 3

The velocity, *v*, and the distance, *d*, as a function time, of a car that accelerates from rest at constant acceleration, *a*, are given by:

$$v(t) = at \quad \text{and} \quad d(t) = \frac{1}{2}at^2$$

Determine **v** and **d** as every second for the first 10 seconds for a car with acceleration of *a*=1.55 m/s². Display the results in a three-column table in which the first column is time (s), the second distance (m), and the third is velocity (km/h).