Introduction to Programming in MATLAB

Mr. Edem K. Bankas

Scripts and Functions

- Files containing MATLAB commands are called m-files and have a .m extension. This mean that all programs written in MATLAB are stored as m-files. There are two types
  - **Scripts**
  - **Functions**

Scripts

- A script is simply a collection of MATLAB commands gathered in a single file.
- Scripts can access all variables in the workspace, and variables created by scripts are stored in the workspace.
- A script can also be described as a sequence of statements that works in the same way as if you entered them one after the other in the Command Window

Functions

- A function is similar to a script, but can not accept and return arguments. It has specific input and output parameters.
- A function is characterized by the fact that (in most cases), variables are passed to it to be processed, and the function returns the calculated result to the caller.

Functions

- The syntax for functions:

  function output = function_name(input)

- it should contain one or several commands defining the output

Examples - script

```
    % This program calculates the area of a circle
    % First the radius is assigned
radius = 5
    % The area is calculated based on the radius
area = pi * (radius^2)
```

Example 2

```
% script file to compute exponential of a set of no
   x = [1 2 3 4 5];
   y = exp(x)
```

Input and Output

- To be more general in writing scripts, it is very useful to read the value of the input from an external source, rather than being assigned in the script. Also, the output can be made to appear nicely.
- We make use of the input and output statements

Input Function

- The simplest input function in MATLAB is called input. The input function is used in an assignment statement.

- To call it, a string is passed, which is the prompt that will appear on the screen, and whatever the user types will be stored in the variable named on the left of the assignment statement.

Input Function cont.

Example

>> radius = input (‘Enter the radius: ’)

- If characters or strings are desired, ‘s’ must be added after the prompt

E.g.,

- >> letter = input(‘Enter your name: , ‘s’)

Input Function cont.

- Separate input statements are necessary if more than one input is desired. E.g.,

```
>> x = input (‘Enter the value of x: ’);
>> y = input (‘Enter the value of y: ’);
```

Output Statements-
disp and fprintf

- Output statements display strings and the results of expressions, and can allow for formatting or customizing how they are displayed.
- The simplest output function in MATLAB is disp , which is used to display the result of an expression or a string without assigning any value to the default variable ans.
- However, disp does not allow formatting.

Output Statements-cont

- Examples:
- >> disp(‘Hello my wonderful students’)
- >> disp(2^10)

Formatted Outputs

- Formatted output can be printed to the screen using the fprintf function. E.g.,
- >>fprintf('The value is %d, \n', 2^10)

Scripts with Input and Output

```
% This script calculates the area of a circle
% It prompts the user for the radius
% Prompt the user for the radius and calculate
% the area based on that radius
radius = input('Please enter the radius: ');
area = pi * (radius^2);
% Print all variables in a sentence format
fprintf('For a circle with a radius of %.2f,',radius)
fprintf('the area is %.2f\n',area)
```

Relational Expressions

- Conditions in if statements use expressions that are conceptually/ logically, either true or false.
- These expressions are called relational expressions, or sometimes Boolean or logical expressions.
- These expressions can use both relational operators, which relate two expressions of compatible types, and logical operators, which operate on logical operands.

Relational Operators

Operator	Meaning
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
==	Equality
~=	Inequality

The IF Statement

- The if statement chooses whether or not another statement, or group of statements, is executed.
- The general form of the if statement is:

```
if condition
        expression
else
        other expression
end
```

Example 1

```
>> num = −4;
>> if num < 0
       num = abs(num);
   end
```

Example 2

```
% Prompt the user for a number and print its sqrt
   num = input('Please enter a number: ');
% If the user entered a negative number, change it
   if num < 0
   num = abs(num);
   end
   fprintf('The sqrt of %.1f is %.1f\n',num,sqrt(num))
```

Modified version of E.g. above

```
% Prompt the user for a number and print its sqrt
   num = input('Please enter a number: ');
% If the user entered a negative number, tell the user
% and change it
   if num < 0
   disp('OK, we''ll use the absolute value')
   num = abs(num);
   end
   fprintf('The sqrt of %.1f is %.1f\n',num,sqrt(num))
```

Question

- Write a program using the IF Else Statement in MATLAB to compute the area of a circle. Prompt the user if a negative number for radius is entered.

IF Else Statement

```
% This script calculates the area of a circle
% It error-checks the user's radius
   radius = input('Please enter the radius: ');
   if radius <= 0
        fprintf('Sorry; %.2f is not a valid radius\n',radius)
else
   area = calcarea(radius);
        fprintf('For a circle with a radius of %.2f,',radius)
        fprintf('the area is %.2f\n',area)
end
```

Nested If Statements