

Lecture 10

Lecture 10: MATLAB Programs

➤ Scripts and User-Defined Functions in Lecture 3 and 4

- Scripts: sequences of statements stored in M-files, and then are executed
- User-defined functions: stored in M-files, calculate and return a single value. The single value could be a single vector when the input argument are vectors
- In this lecture, we are going to introduce some other kinds of user-defined functions

Lecture 10: MATLAB Programs

➤ More Types of User-Defined Functions

- Functions that calculate and return one value
- Functions that calculate and return more than one value
- Functions that just accomplish a task, such as printing, without returning any values

➤ Components of MATLAB Functions

- The function header (the first line), which includes
 - The reserved word function
 - If the function returns values, the names of the output arguments, followed by the assignment operator =
 - The name of the function (This should be the same as the name of the M-file)
 - The input arguments in parentheses, if there are any input arguments. If there are more than one, they are separated by commas.
- A comment that describes what the function does
- The body of the function, which includes all statements, including assigning values to all output arguments if there are any

Lecture 10: MATLAB Programs

➤ Functions that Return More than One Value

- General form

functionname.m

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

```
% Comment describing the function
```

```
Statements (must include assigning values to all of the output arguments  
listed in the header)
```

The number of output arguments do not have to be the same as the number of the input arguments.

Lecture 10: MATLAB Programs

➤ Functions that Return More than One Value

▪ Example

areacirc.m

```
function [area, circum] = areacirc(rad)
% This function calculates the area and the circumference of a circle
area = pi * rad .* rad;
circum = 2 * pi * rad;
```

```
>> [a c] = areacirc(4)
```

```
a =
```

```
50.2655
```

```
c =
```

```
25.1327
```

If **this is not done**, only the first value returned is retained. In this case, the area:

```
areacirc(4)
```

```
ans =
```

```
50.2655
```

```
>> disp(areacirc(4))
```

```
50.2655
```

What would happen if a vector of radii was passed to the function?

```
>> [a c] = areacirc(1:4)
```

What if we call the function as

```
>> a = areacirc(4)
```

Lecture 10: MATLAB Programs

➤ Functions that Return More than One Value

- The function could be also called from a script. Here is a script that will prompt the user for the radius of just one circle, call the areacirc function, to calculate and return the area and circumference of the circle and print the result:

calcareacirc.m

```
% This script prompts the user for the radius of a circle
% calls a function to calculate and return both the area and circumference,
and prints the results
radius = input('Please enter the radius of the circle: ')
[area circ] = areacirc(radius);
fprintf('For a circle with a radius of %.1f\n', radius)
fprintf('the area is %.1f and the circumference is %.1f\n', area, circ)
```

>> calcareacirc

Please enter the radius of the circle: 5.2

For a circle with a radius of 5.2,

the area is 84.9 and the circumference is 32.7

Lecture 10: MATLAB Programs

➤ Functions that Accomplish a Task **Without Returning Values**

- General form

functionname.m

```
function functionname(input arguments)
% Comments describing the function
Statements here
```

Notice: there is no output argument, and no assignment operator.

Example

printem.m

```
function printem(a,b)
% This function prints two numbers in a sentence format
fprintf('The first number is %.1f and the second is %.1f\n', a, b)
```

```
>> printem(3.3, 2)
```

The first number is 3.3 and the second is 2.0

Since the function does not return any value, it cannot be called from an assignment statement. For example, `var = printem(a,b)` is wrong.

Lecture 10: MATLAB Programs

➤ Functions that Return Values Versus Printing

- A function that calculates and returns values (through the output arguments) **does not normally also print them.** That is left to the calling script or function
- If a function just prints a value, rather than returning it, **the value cannot be used later on** in other calculations.
(**Why? That value was not assigned to any variable**)

Lecture 10: MATLAB Programs

➤ Passing Arguments to Functions

- In some cases, however, it is not necessary to pass any arguments to the function.

printrand.m

```
function printrand()  
% This function prints one random number  
fprintf('The random # is %2f\n', rand)
```

>> printrand()

The random # is 0.94

No input arguments
and no output
arguments. **Can we
have output
arguments?**

In fact, the parentheses are not even needed in either the function or the function call.

>> printrand

The random # is 0.94

Lecture 10: MATLAB Programs

□ MATLAB Program Organization

➤ Modular Programs

- In a modular program, the solution is broken down **into modules**, and each is implemented as a function.
- For the example of calculating the area of a circle, there are three steps in the algorithm
 - Get the input (the radius)
 - Calculate the area
 - Display the result
- In a **modular program**, there would be **one main script** that calls three separate functions to accomplish these tasks:
 - A function to prompt the user and read in the radius
 - A function to calculate the area of the circle
 - A function to display the results

Since **both scripts and functions are stored in M-files**, there would be four separate M-files altogether for this program.

Lecture 10: MATLAB Programs

➤ Modular Programs

calcandprintarea.m

```
% This is the main script to calculate the area of a circle
% It calls 3 functions to accomplish this
radius = readradius;
area = calcarea(radius);
printarea(radius, area)
```

readradius.m

```
function radius = readradius
% This function prompts the user and reads the radius
disp('When prompted, please enter the radius in inches. ')
radius = input('Enter the radius: ');
```

calcarea.m

```
function area = calcarea(rad)
% This function calculates the area of a circle
area = pi * rad * rad;
```

printarea.m

```
function printarea(rad, area)
% This function prints the results
fprintf('For a circle with a radius of %.2f inches,\n', rad)
fprintf('the area is %.2f inches squared.\n', area)
```

Lecture 10: MATLAB Programs

➤ **Homework 10 on Canvas**