

- Previous topic:
 - Introduction to `for`-loop
- Now:
 - Review `for`-loop
 - User-defined function

In the loop body, never change the value of the loop variable

```
n= _____  
for k= 1:n  
    % code to do  
    % that something  
end
```

3

Drawing ASCII diagrams

Print these diagrams on the Command Window

```
*****  
*****  
*****  
*****  
*****
```

```
 *  
**  
***  
****  
*****  
*****
```

An independent task—make it a function

Printing is done left to right and top to bottom.
What is a simpler (sub)problem?
→ Print just one row of asterisks.
What is an even simpler (sub)problem?
→ Print just one single asterisk!

12

Functions that we have seen

- E.g., to evaluate the cosine of 2 radians you type
`x = cos(2)`
i.e., you call the built-in function by its name and give it a value to work on. Here I store the result in a variable.
- To draw a colored disk, you may type
`DrawDisk(1,9,5,'y')`

→ A function (usually) requires some “starting data” and either returns a value or shows some output

Now you learn how to create your own functions

13

What is the code to draw in one row some character n times?

```
% n needs a value  
% theChar needs a value (character)  
  
for k= 1:n  
    fprintf('%c', theChar)  
end  
fprintf('\n')
```

16

Keyword to indicate this is a function file (not a script)

```
function printRepeatChar(theChar,n)  
% Print to the Command Window the  
% character in variable theChar n times.  
% n is non-negative integer.  
% Add a linebreak.  
  
for k= 1:n  
    fprintf('%c', theChar)  
end  
fprintf('\n')
```

18

Function name, same as the file name

```
function printRepeatChar(theChar,n)
% Print to the Command Window the
% character in variable theChar n times.
% n is non-negative integer.
% Add a linebreak.

for k= 1:n
    fprintf('%c', theChar)
end
fprintf('\n')
```

19

Parameters, also called input parameters

```
function printRepeatChar(theChar,n)
% Print to the Command Window the
% character in variable theChar n times.
% n is non-negative integer.
% Add a linebreak.

for k= 1:n
    fprintf('%c', theChar)
end
fprintf('\n')
```

20

Function header

```
function printRepeatChar(theChar,n)
% Print to the Command Window the
% character in variable theChar n times.
% n is non-negative integer.
% Add a linebreak.

for k= 1:n
    fprintf('%c', theChar)
end
fprintf('\n')
```

21

Function comments, i.e., specifications
(VERY IMPORTANT to have this comment)

```
function printRepeatChar(theChar,n)
% Print to the Command Window the
% character in variable theChar n times.
% n is non-negative integer.
% Add a linebreak.

for k= 1:n
    fprintf('%c', theChar)
end
fprintf('\n')
```

Function body

22

Calling a function (invoking a function)

- I have a function with this header:

```
function printRepeatChar(theChar,n)
```
- I will call function printRepeatChar like this:

```
printRepeatChar('*',8)
```
- The function header tells you everything you need to know about how to call the function

23

Drawing ASCII diagrams

```
*****          *
*****          **
*****          ***
*****          ****
*****          *****
*****          *****
```

```
% printRectangle
for r= 1:4
    % Print rth row
    printRepeatChar('*',7)
end
```

24

Given this function:

```
function m = convertLength(ft,in)
% Convert length from feet (ft) and inches (in)
% to meters (m).
. . .
```

How many proper calls to `convertLength` are shown below?

```
f= ...; n= ...;
d= convertLength(f,n);
d= convertLength(f+n/12);
d= convertLength(f+n/12);
x= min(convertLength(f,n), 1);
y= convertLength(pi*(f+n/12)^2);
```

A: 1 B: 2 C: 3 D: 4 E: 5 or 0

25

General form of a user-defined function

```
function [out1, out2, ...]= functionName (in1, in2, ...)
% 1-line comment to describe the function
% Additional description of function
```

Executable code that at some point assigns values to output parameters `out1, out2, ...`

- `in1, in2, ...` are defined when the function begins execution. Variables `in1, in2, ...` are called function *parameters* and they hold the function *arguments* used when the function is invoked (called).
- `out1, out2, ...` are not defined until the executable code in the function assigns values to them.

26

Implement this function

```
function r = randReal(lo,hi)
% r is a random real number in the
% interval (lo,hi)
```

27

Implement this function

```
function [x,y] = randPt(lo,hi)
% x and y are random real numbers in the
% interval (lo,hi)
```

29