



COGS 119/219

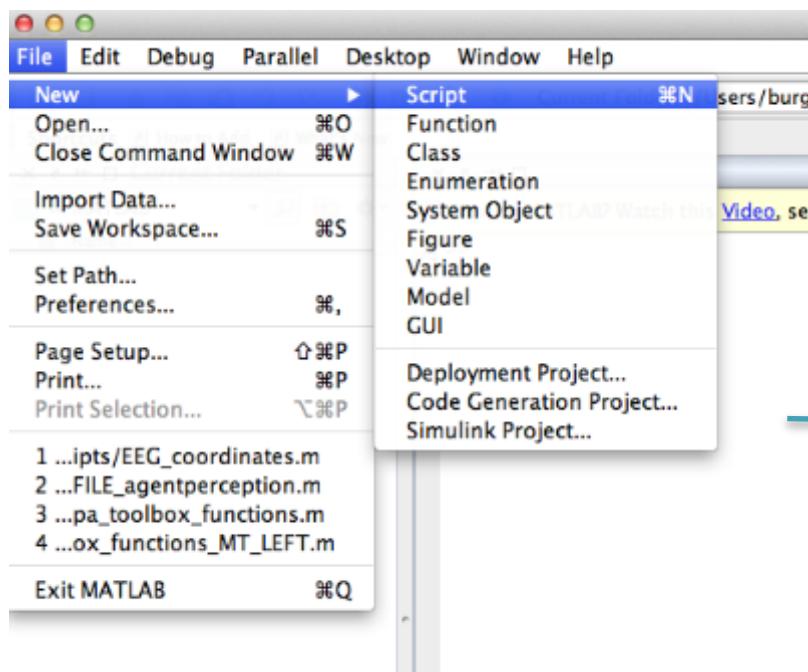
MATLAB for Experimental Research

Fall 2014 – Week 2 (Tue)

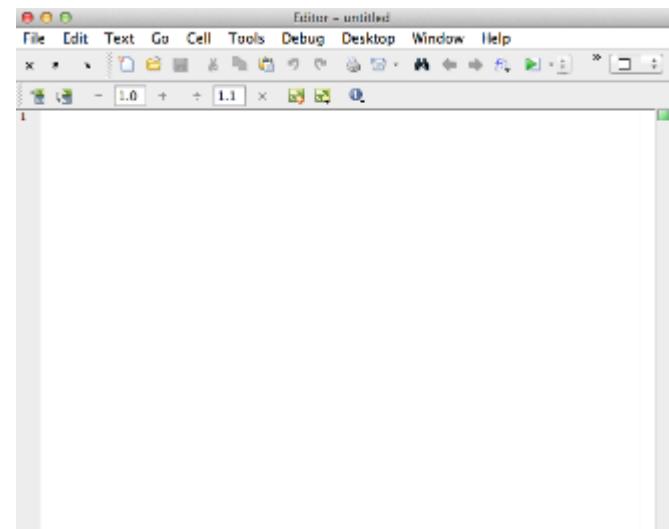
Flow Control: Example with Loops and
Conditional Expressions and
Cell Arrays

Recall how to work with .m files

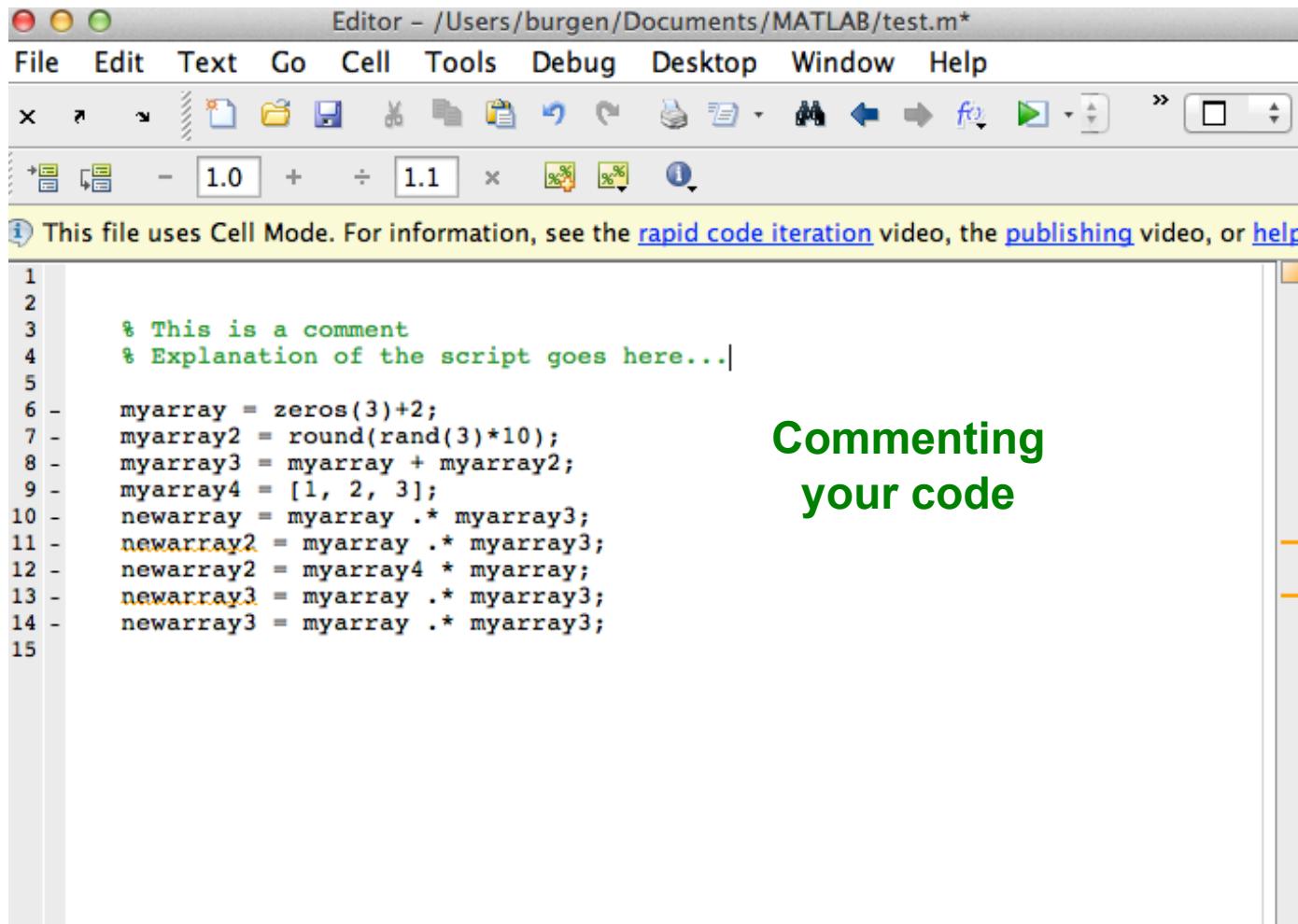
- We can write the MATLAB commands that we type at the command window in a file: extension .m



Editor window for the .m file



Recall how to work with .m files

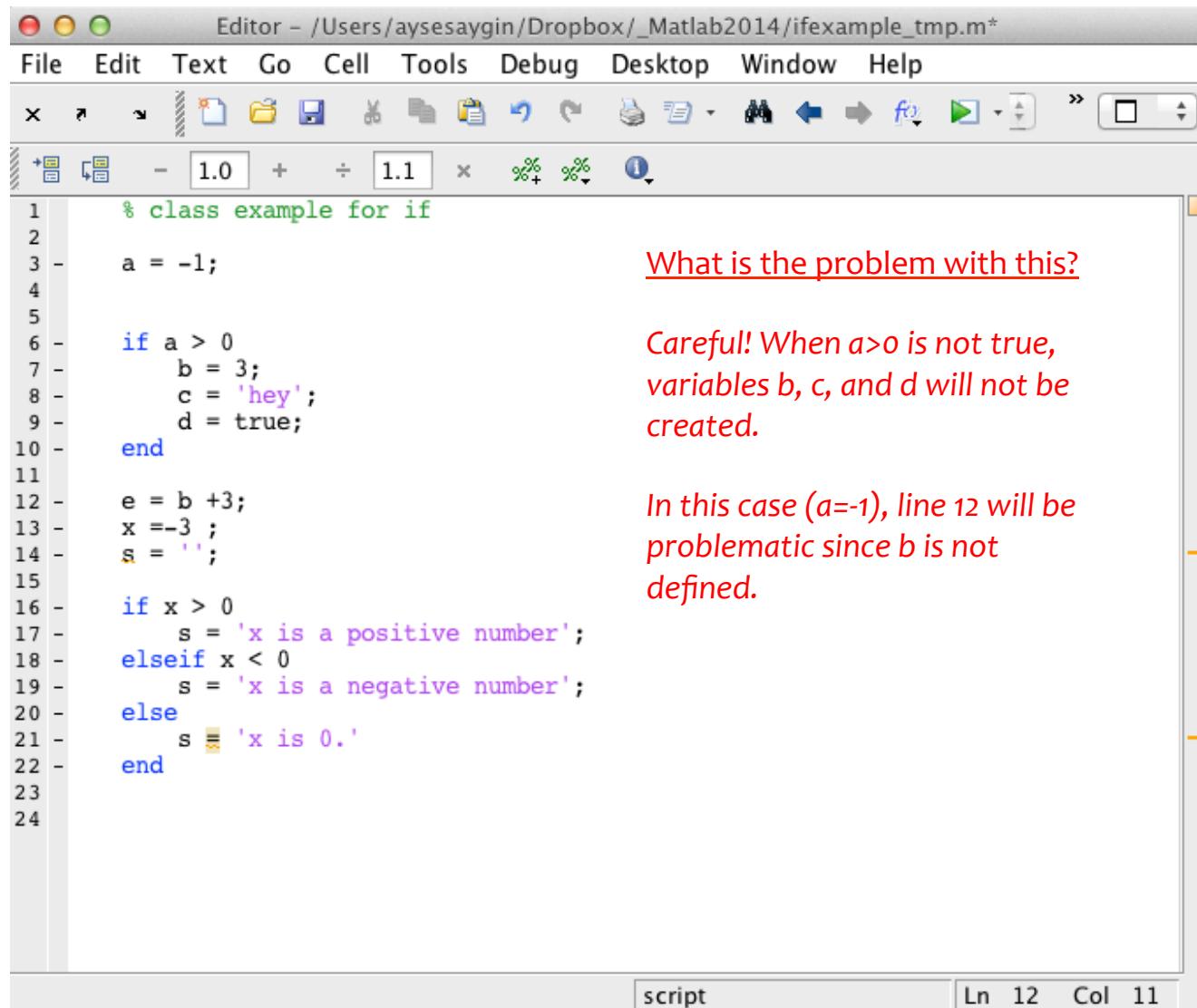


The image shows a screenshot of the MATLAB Editor window. The title bar reads "Editor - /Users/burgen/Documents/MATLAB/test.m*". The menu bar includes File, Edit, Text, Go, Cell, Tools, Debug, Desktop, Window, and Help. Below the menu is a toolbar with various icons. A status bar at the bottom shows numerical values: 1.0, 1.1, and some percentage symbols. A message in the status bar says: "This file uses Cell Mode. For information, see the [rapid code iteration](#) video, the [publishing](#) video, or [help](#)". The main code area contains the following:

```
1
2
3 % This is a comment
4 % Explanation of the script goes here...
5
6 myarray = zeros(3)+2;
7 myarray2 = round(rand(3)*10);
8 myarray3 = myarray + myarray2;
9 myarray4 = [1, 2, 3];
10 newarray = myarray .* myarray3;
11 newarray2 = myarray .* myarray3;
12 newarray2 = myarray4 * myarray;
13 newarray3 = myarray .* myarray3;
14 newarray3 = myarray .* myarray3;
15
```

**Commenting
your code**

An example .m file and using flow control (if statement)



The screenshot shows a MATLAB Editor window with the following code:

```
% class example for if
a = -1;

if a > 0
    b = 3;
    c = 'hey';
    d = true;
end

e = b +3;
x =-3 ;
s = '';

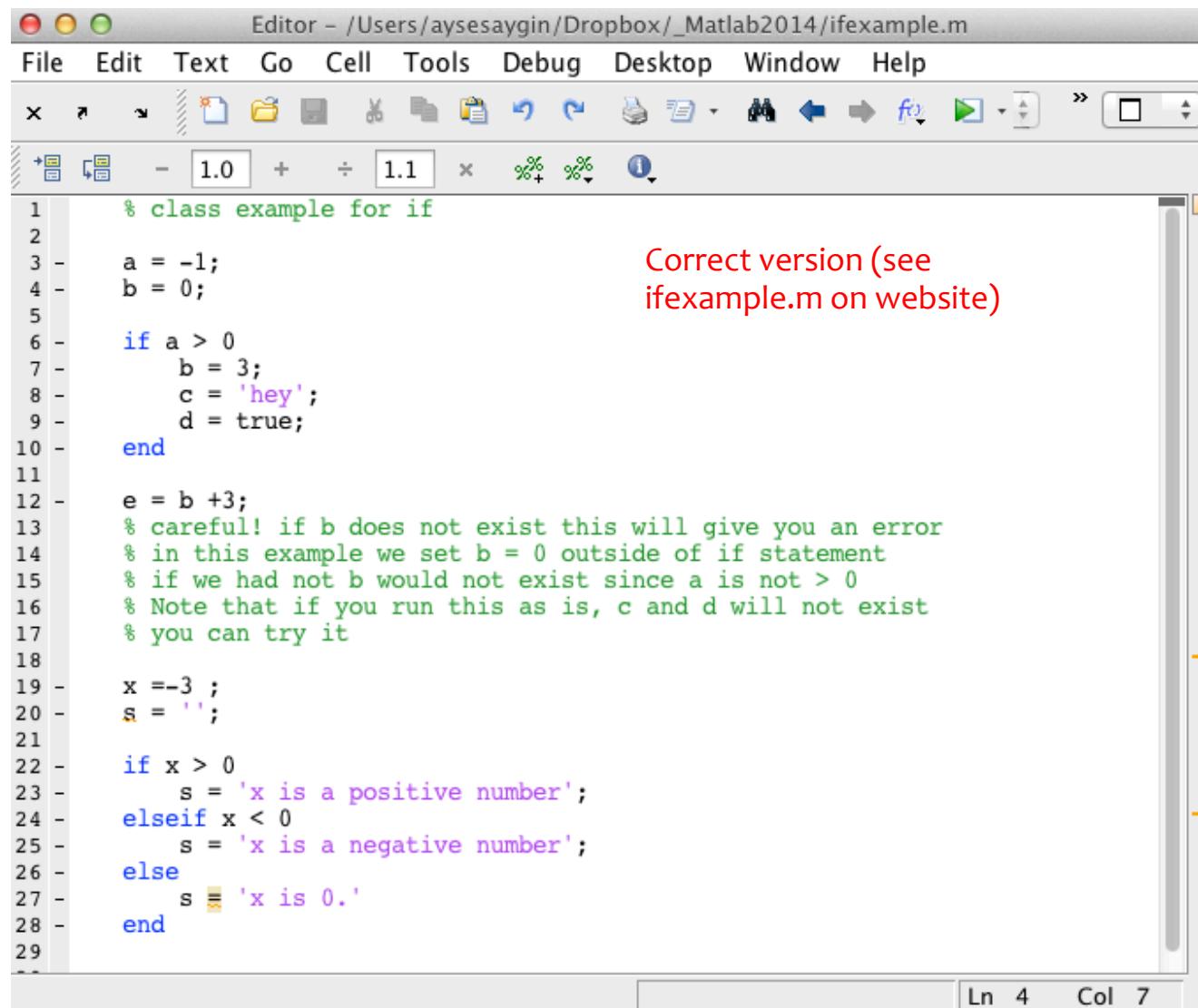
if x > 0
    s = 'x is a positive number';
elseif x < 0
    s = 'x is a negative number';
else
    s = 'x is 0.'
end
```

What is the problem with this?

Careful! When $a>0$ is not true, variables b , c , and d will not be created.

In this case ($a=-1$), line 12 will be problematic since b is not defined.

An example .m file and using flow control (if statement)



The screenshot shows a MATLAB Editor window with the title "Editor - /Users/aysesaygin/Dropbox/_Matlab2014/ifexample.m". The menu bar includes File, Edit, Text, Go, Cell, Tools, Debug, Desktop, Window, and Help. The toolbar below has various icons for file operations like new, open, save, and zoom. The code area contains the following MATLAB script:

```
% class example for if
a = -1;
b = 0;
if a > 0
    b = 3;
    c = 'hey';
    d = true;
end
e = b +3;
% careful! if b does not exist this will give you an error
% in this example we set b = 0 outside of if statement
% if we had not b would not exist since a is not > 0
% Note that if you run this as is, c and d will not exist
% you can try it
x =-3 ;
s = '';
if x > 0
    s = 'x is a positive number';
elseif x < 0
    s = 'x is a negative number';
else
    s = 'x is 0.'
end
```

A red annotation on the right side of the code states: "Correct version (see ifexample.m on website)". The status bar at the bottom right shows "Ln 4 Col 7".

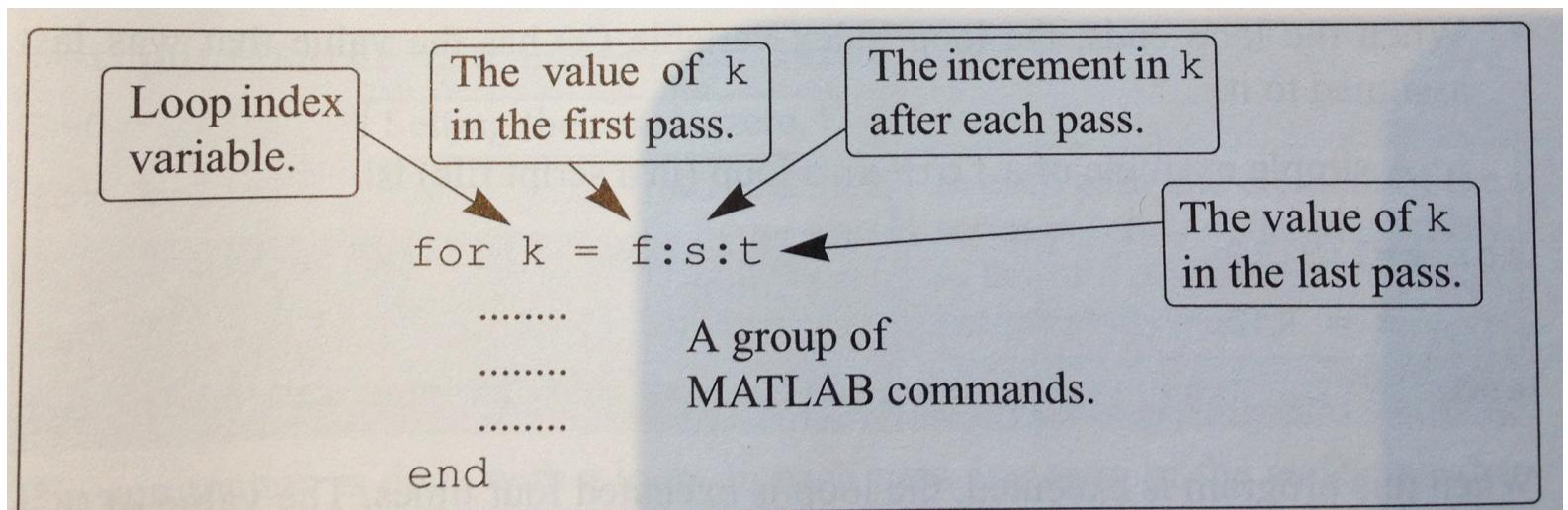


LOOPS

- Another method to alter the flow of a computer program.
- In a loop, the execution of a command, or a group of commands, is repeated several times consecutively.
- In each pass, at least one variable defined within the loop is/are assigned new values.
- MATLAB has two kinds of loops: **for-end** loops and **while-end** loops.

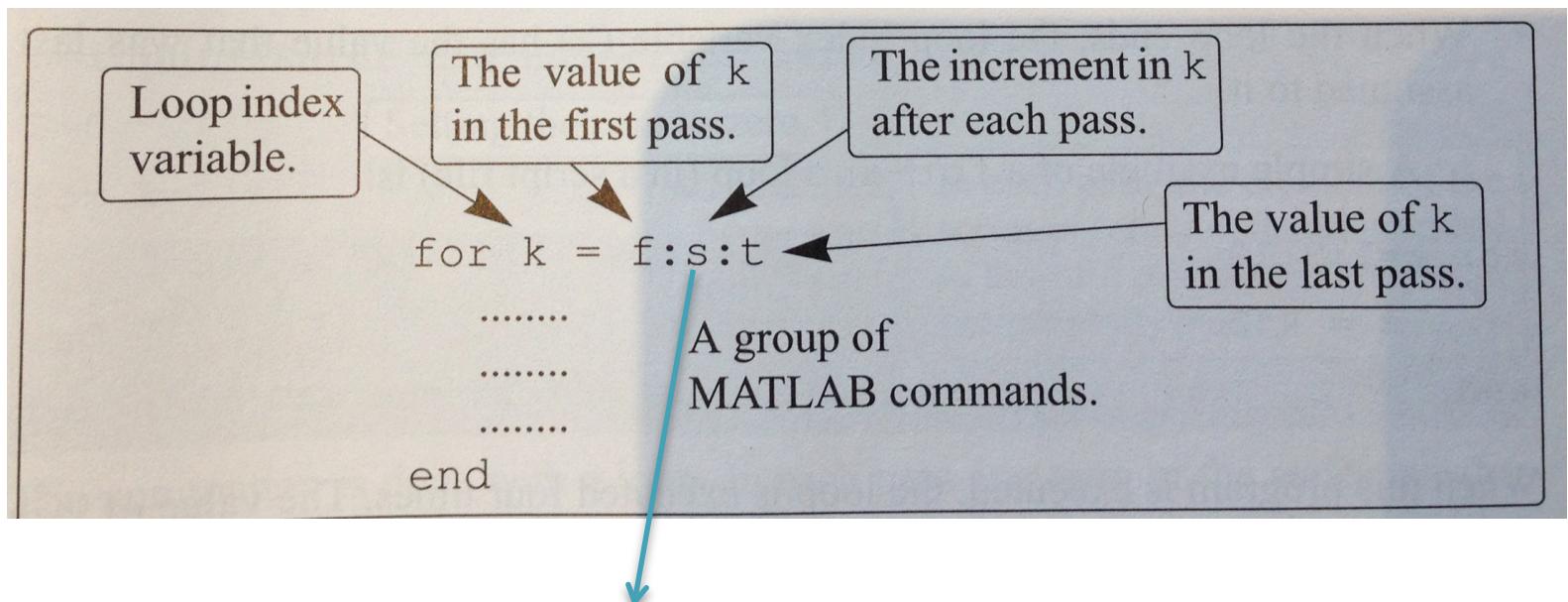
for loops

- The execution of a command, or a group of commands, is repeated a predetermined number of times.



for loops

- The execution of a command, or a group of commands, is repeated a predetermined number of times.



Most of the time, the increment value is omitted.
The default value of `s` is 1 (increment 1 in each iteration).

for loops: Example

```
for k = 1:4  
    x = k^2;  
end
```

for loops: Example

```
for k = 1:4  
    x = k^2;  
end
```

The loop is executed 4 times. Here is how the loop works:

1. In the first iteration, $k = 1$. Then the program executes $x = 1^2$, which is 1.
2. In the second iteration, $k = 2$. Then the program executes $x = 2^2$, which is 4.
3. In the third iteration, $k = 3$. Then the program executes $x = 3^2$, which is 9.
4. In the fourth iteration, $k = 4$. Then the program executes $x = 4^2$, which is 16.

while loops

- while loops are used in situations when looping is needed but the number of iterations is not known ahead of time.
- In while loops, the number of iterations is not specified when the looping starts. Instead, the looping process continues until a stated condition is satisfied.

```
while conditional expression
    .....
    .....
    .....
end
```

A group of
MATLAB commands.

while loops

```
while conditional expression
```

```
.....  
.....  
.....  
end
```

A group of
MATLAB commands.

Conditional expression is checked.

If it is false, MATLAB skips to the end statement and continues with the program.

If it is true, MATLAB executes the group of commands that follow between the **while** and **end** command.

Then, MATLAB jumps back to the **while** command and checks the conditional expression.

This looping process continues until the conditional expression is false.

while loops: Example

```
x = 1;  
while x <= 7  
    x = 2*x ;  
end
```

while loops: Example

```
x = 1;  
while x <= 7  
    x = 2*x ;  
end
```

Here is how the program above works:

1. x is assigned to 1.
2. The conditional statement $x \leq 7$ is checked. Since it is true ($1 \leq 7$), the following command $x = 2*x$ is executed, which makes $x = 2$.
3. It jumps back to while statement and checks the conditional statement. Since $2 \leq 7$, the following command, $x = 2*x$ is executed, which makes $x = 4$.
4. It jumps back to the while statement and checks $4 \leq 7$. Since it is true, it executes $x = 2*x$, which makes $x = 8$.
5. It jumps back to the while statement and checks $8 \leq 7$. Since it is false, it jumps to the end without executing the statements between while and the end.

Note about **while** loops

- You have to be sure that the variable (or variables) that are in the conditional expression are assigned new values during the looping process, and will eventually be assigned values that make the conditional expression false.
 - Otherwise, the looping will continue indefinitely...
 - For instance, consider

```
x = 3;  
while x > 2  
    x = 2*x ;  
end
```

Important: While loops are especially vulnerable to being stuck in infinite loops.

You must make sure your code will reach a point at which the logical statement won't be true.



Nested Loops and conditional statements

- Loops and conditional statements can be nested within themselves and each other.
- This means that a loop and/or a conditional statement can start (and end) within another loop and/or conditional statement.
- There is no limit to the number of loops and conditional statements that can be nested.

Nested loops

```
for k = 1:n
    for h = 1:m
        .....
        .....
        .....
    end
end
```

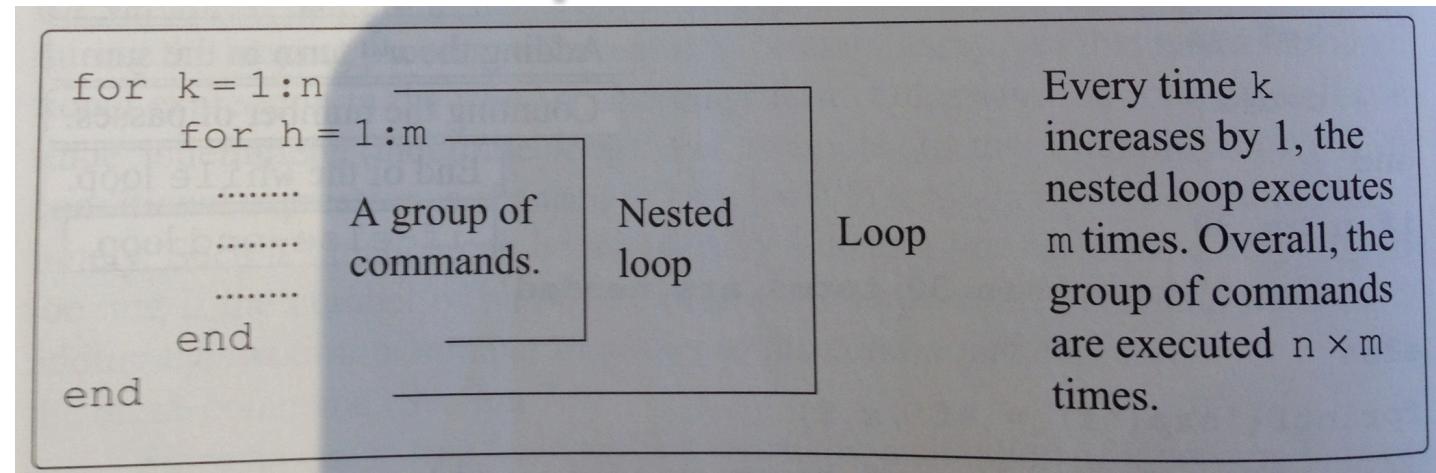
A group of commands.

Nested loop

Loop

Every time k increases by 1, the nested loop executes m times. Overall, the group of commands are executed $n \times m$ times.

Nested loops



For example, if $n = 3$ and $m = 4$:

1. $k = 1$, and the nested loop executes 4 times with $h = 1, 2, 3, 4$.
2. $k = 2$, and again the nested loop executes 4 times with $h = 1, 2, 3, 4$.
3. $k = 3$, and again the nested loop executes 4 times with $h = 1, 2, 3, 4$.



break command

- When inside a loop (`for` and `while`), the `break` command terminates the execution of the loop (the whole loop, not just the last iteration).
- MATLAB jumps to the `end` command of the loop and continues with the next command.
- If the `break` command is inside a nested loop, only the nested loop is terminated.
- When a `break` command appears outside of a loop in a script file, it terminates the execution of the file.



continue command

- The `continue` command can be used inside a loop (for and while) to stop the present iteration and start the next iteration in the looping process.
- The `continue` command is usually a part of a conditional statement. When MATLAB reaches the `continue` command, it does not execute the remaining commands in the loop, but skips to the `end` command of the loop, and then starts a new iteration.

Loops: For and While



Editor - /Users/aysesaygin/Dropbox/_Matlab2014/loopy.m

File Edit Text Go Cell Tools Debug Desktop Window Help

x ↻ ⌂ ⌂ 1.0 + ÷ 1.1 × %%+ %%- ⓘ

```
1 % my first for loop
2 n = 10;
3 m = 5
4 x = [];
5
6 for i = [1:m,m+2:n]
7 x = [x, i^2];
8 end
9
10 % my first while loop
11 a = 5;
12 j = 1;
13 m = 0;
14 while j < a
15 m = m+j+a;
16 j=j+1;
17 end
18 % I could have done this with for loop
19 % IF THEN ELSE exercise|
```