20-Loops Part 1

text: Chapter 6.4-6.5

ECEGR 101

Engineering Problem Solving with Matlab

Professor Henry Louie

Overview

- For Loop
- For Loop Versus Element-by-Element Operations
- Special Case of For Loops

FOR Loop

- Used to repeat a set of commands a known number of times.
- Syntax:

```
for loop_counter = m:s:n
        set of commands
end
```

Example

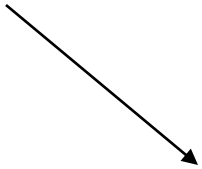
```
for k = 1:3
    disp('hello world')
end
```

```
hello world
hello world
hello world
>>
```

MATLAB increments index k automatically.

Exercise

Modify the "hello world" program so the user enters the number of times the message is generated.



```
for k = 1:3
    disp('hello world')
end
```

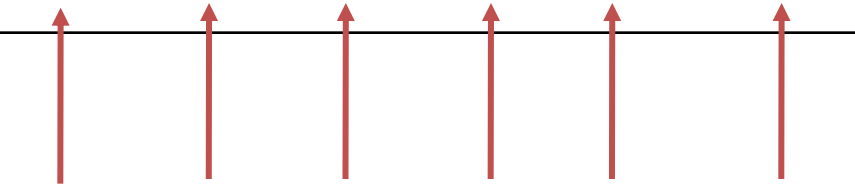
Exercise

```
n = input('How many times should the message
    be displayed?');
for k = 1 : n
    disp('hello world');
end
```

Example

Use a FOR loop to display the elements of vector x = 5:10 in the Command Window.

```
>> x = 5:10
x =
     5     6     7     8     9    10
```




```
for i = 1:length(x)
    x(i)
end
```

```
ans =
     5
ans =
     6
ans =
     7
ans =
     8
ans =
     9
ans =
    10
```

Example

Use a FOR loop to display the elements of vector x = 5:10 in the Command Window.

```
>> x = 5:10
x =
     5     6     7     8     9    10
```

```
for i = 1:length(x)
    disp(x(i))
end
```

```
     5
     6
     7
     8
     9
    10
```

Example

FOR loop can be used to create a vector:

```
clear all;
for i = 1:5
    x(i) = i^2
end
```

```
x =
     1
x =
     1     4
x =
     1     4     9
x =
     1     4     9    16
x =
     1     4     9    16    25
```

Example

Use the semicolon (;) to suppress the intermediate output.

```
clear all;
for i = 1:5
    x(i) = i^2;
end
```

```
>> x

x =

     1     4     9    16    25

>>
```

Exercise

What are the values of x and y after the following loop is executed?

```
clear all;
x = 1:5;
for i = 1:length(x)
    y(i) = 10*x(i);
end
```

Exercise

What are the values of x and y after the following loop is executed?

```
clear all;
x = 1:5;
for i = 1:length(x)
    y(i) = 10*x(i);
end
```

```
x =
     1     2     3     4     5
>>
```

```
>> y
y =
    10    20    30    40    50
>>
```

Exercise

How can we compute the same vector x but without the FOR loop?

```
>> k
k =
    6
>>
>> x
x =
        1        4       27      256     3125    46656
>>
```

Exercise

How can we compute the same vector x but without the FOR loop?

```
>> k = 1:6;
>> x = k.^k;
>>
>> k
k =
   1    2    3    4    5    6
>> x
x =
        1        4       27      256     3125    46656
```

```
for k = 1:6
    x(k) = k^k;
end
```

FOR Loop vs. Element-By-Element Operations

- Where possible, always use element-by-element operations instead of FOR loops.
  - Element-by-element operations are faster than FOR loops.
  - Program is easier to read with element-by-element operations.
- Using element-by-element operations instead of loops is called vectorizing the program.

Example

We are going to use MATLAB functions tic and toc to compare the time of execution of different set of commands.

- tic - starts a stopwatch timer
- toc - prints the elapsed time since tic was used
- Syntax:

tic, operations, toc

Example

```
>> tic; for k = 1:10^5, x(k) = k^2; end, toc
Elapsed time is 52.979007 seconds.
```

FOR loop is the slowest.

```
>> tic; k = 1:10^5; x = k.^2; toc
Elapsed time is 0.124595 seconds.
```

Element-by-element calculations are the fastest.

Example

```
>> tic; x = zeros(1, 10^5);
    for k = 1:10^5, x(k) = k^2; end, toc
Elapsed time is 0.219823 seconds.
```

Pre-allocating the array speeds up the FOR loop.

Exercise

Use a FOR loop to write a program that creates a vector of length N (N is the input) such that each element of the vector is an N-th power of its index.

For example, if N = 4; the output vector is [1 16 81 256] that is [1^4 2^4 3^4 4^4].

Exercise

```
N=input('Enter N: ')
for i= 1:N
    v(i)=i^N
end
```

```
>> loop_examples
Enter N: 2
N =
     2
v =
     1     4    27
>> loop_examples
Enter N: 3
N =
     3
v =
     1     8    27
>> loop_examples
Enter N: 4
N =
     4
v =
     1    16    81   256
```

Vectorized implementation:

v = (1:N).^N;

Exercise

Write a program that computes the sum of all elements in a vector using a FOR loop (not the built-in *sum* function).

Exercise

```
x=input('Enter x: ');
s=0;
for i=1:length(x)
    s = s+x(i);
end
s
```

```
Enter x: [1 2 4 6 2 3]
s =
    18
```

Exercise

Write a program to calculate y(t) defined as

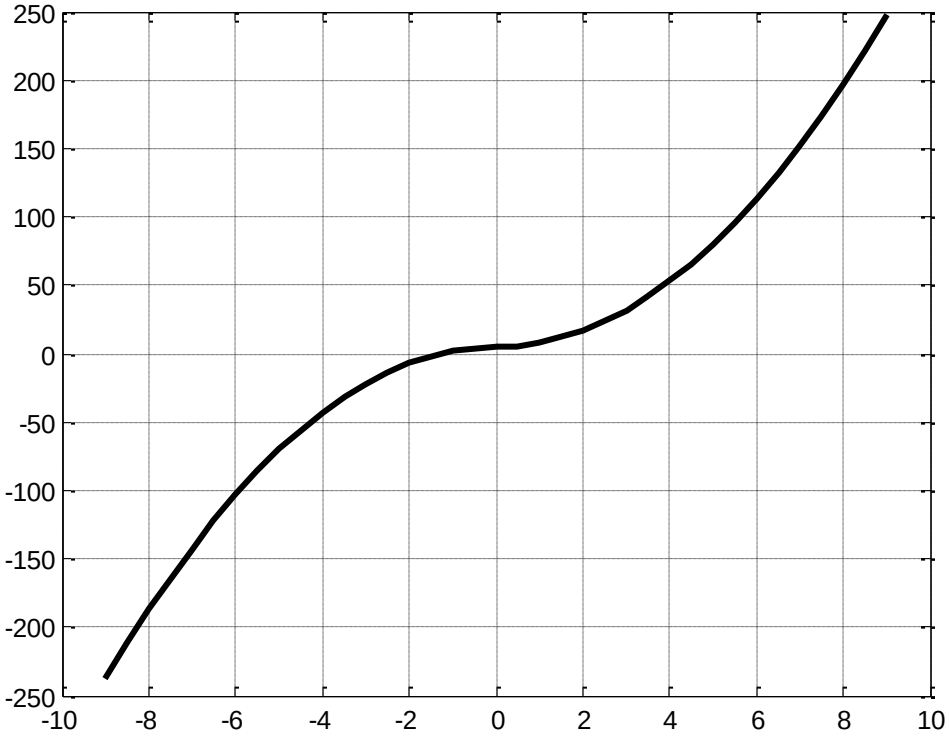
$$y(t) = \begin{cases} -3t^2 + 5 & t \geq 0 \\ 3t^2 + 5 & t < 0 \end{cases}$$

Write a script that uses the above function in a FOR loop structure to calculate and plot this function for t between -9 and 9 in steps of 0.5. Show grid lines and make the trace thick and black.

Exercise

```
t=[-9:.5:9];
for i=1:numel(t)
    if t(i)<=0
        y(i) = -3*t(i)^2 + 5;
    else
        y(i) = 3*t(i)^2 + 5;
    end
end
figure;
plot(t,y,'color','black','linewidth',2.5)
grid on
```

Exercise



Nested FOR Loops

Syntax

```
for loop_counter1 = m1:s1:n1
      set of commands1
      for loop_counter2 = m2:s2:n2
            set of commands2
      end
end
```

Example

```
for a=1:3
    for b=1:3
        y(a,b) = a + 2*b
    end
end
```

```
y =
     3
y =
     3     5
y =
     3     5     7
y =
     3     5     7
     4     0     0
y =
     3     5     7
     4     6     0
y =
     3     5     7
     4     6     8
y =
     3     5     7
     4     6     8
     5     0     0
y =
     3     5     7
     4     6     8
     5     7     0
y =
     3     5     7
     4     6     8
     5     7     9
```

A total of a*b elements are created

Exercise

Use nested for loops to find the minimum value of z when x and y are integers between 1 and 10.

$$z = (x-6)^2 + (2y-8)^2$$

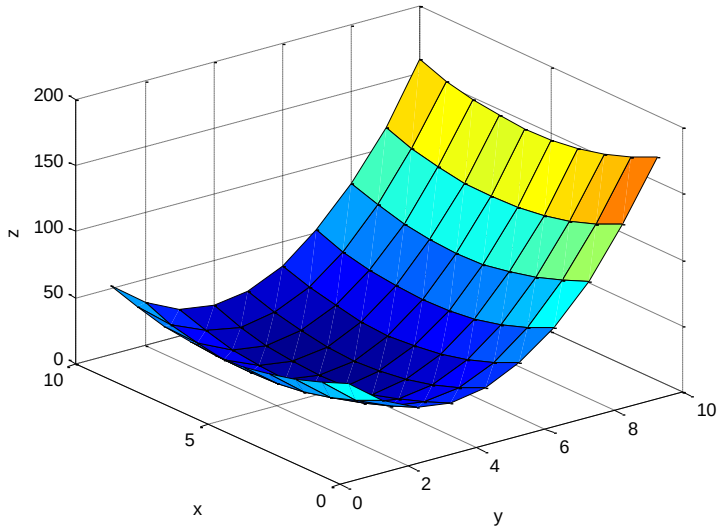
Exercise

```
for x=1:10
    for y=1:10
        z(x,y) = (x-6)^2 + (2*y - 8)^2;
    end
end
z
[values,min_y]=min(z)
[min_value,min_x]=min(values);%%x value for min. z
min_y(min_x)%%y value for min. z
surf(z)%%plot the function z
```

```
z =

    61    41    29    25    29    41    61    89   125   169
    52    32    20    16    20    32    52    80   116   160
    45    25    13     9    13    25    45    73   109   153
    40    20     8     4     8    20    40    68   104   148
    37    17     5     1     5    17    37    65   101   145
    36    16     4     0     4    16    36    64   100   144
    37    17     5     1     5    17    37    65   101   145
    40    20     8     4     8    20    40    68   104   148
    45    25    13     9    13    25    45    73   109   153
    52    32    20    16    20    32    52    80   116   160
```

Minimum z occurs when
X = 6, y =4



Special Case of FOR Loops

FOR loops can be used to sweep through any fixed set of elements.

```
I = [1.3 4.6 -2.3 9.8];

for i = I
    fprintf('Value of i: %.2f.\n', i)
end
```

```
Value of i: 1.30.
Value of i: 4.60.
Value of i: -2.30.
Value of i: 9.80.
```

Special Case of FOR Loops

```
>>
>> for k = [1 3 9]
v(k) = k^3;
end
>>
>> v
v =
     1     0    27     0     0     0     0     0   729
>>
```

Special Case of FOR Loops

```
>> for k = [1 3 9]
       v(k) = k^3
     end

v =
    1
v =
    1    0   27
v =
    1    0   27    0    0    0    0    0  729
```