

# **Programming-MATLAB**

| P r a c t i c e   c l a s s   c o u r s e |

## ❖ Boolean operators and relational operators

- Boolean operators truth table

inputs		outputs					
		and	or	Not	xor	Short circuit and	Short circuit or
A	B	A & B	A   B	~A	xor(A,B)	A && B	A    B
0	0	0	0	1	0	0	0
0	1	0	1	1	1	0	1
1	0	0	1	0	1	0	1
1	1	1	1	0	0	1	1

- relational operators

Relational operator	Meaning
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equal to
~=	Not equal to

## ❖ Boolean operators and relational operators

- Boolean operator truth table

inputs		outputs					
		and	or	Not	xor	Short circuit and	Short circuit or
A	B	A & B	A   B	~A	xor(A,B)	A && B	A    B
0	0	0	0	1	0	0	0
0	1	0	1	1	1	0	1
1	0	0	1	0	1	0	1
1	1	1	1	0	0	1	1

```
>>x = [1 2 3 0]; y = [0 2 9 1];
```

```
>>k = x & y
```

```
>>k = ~ (x | y)
```

```
>>k = xor(x, y)
```

```
>>k = x && y
```

```
>>k = x || y
```

## ❖ Boolean operators and relational operators

- relational operator

Relational operator	Meaning
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equal to
~=	Not equal to

```
>>x = [1 2 3 0]; y = [0 2 9 1];
```

```
>>k = x < y
```

```
>>k = x <= y
```

```
>>k = x ~= y
```

### ❖ Boolean operators and relational operators

- Given  $x = [1 \ 2 \ 3 \ 4]$  and  $y = [0 \ 2 \ 9 \ 1]$ , find all the elements in  $x$  that are **less than  $y$**  the corresponding elements in  $y$  and **less than 4**.

```
>>x = [1 2 3 4]; y = [0 2 9 1];  
>>k = x( (x<y) & (x<4) )
```

## ❖ if statement

- Syntax :  
if *logical expression*  
  *statements*  
end

```
>>n = input('# of apples:');  
>>cost = n*500;  
>>if (n>=10)  
>>  cost = 0.8*cost;  
>>end  
>>sprintf('# of apples = %d cost = %d', n, cost)
```

`n = input(prompt);` displays the text in *prompt* and waits for the user to input a value and press the Return key(Enter key).  
input function is similar to scanf in the C.

## ❖ if statement

- Syntax :  
if *logical expression*  
    *statements*  
elseif *logical expression*  
    *statements*  
else  
    *statements*  
end
- 'end' appears only at the bottom of the conditional statement

```
>>a = input('coefficient of x^2:')
>>b = input('coefficient of x:')
>>c = input('constant :')
>>D = b^2-4*a*c;
>>if (D>0)
>>    sprintf('two distinct real number solutions')
>>elseif (D==0)
>>    sprintf('repeated real number solution')
>>else
>>    sprintf('neither of the solutions are real number')
>>end
```

`n = input(prompt);` displays the text in *prompt* and waits for the user to input a value and press the Return key(Enter key).  
input function is similar to scanf in the C.

## ❖ switch statement

- Use switch statement instead of if statement.
- Find the roots of quadratic polynomial
- Using built-in function : sign(x)
  - $Y = \text{sign}(x)$  returns an array Y the same size as x, where each element of Y is:
    - 1 if the corresponding element of x is greater than 0
    - 0 if the corresponding element of x is equals 0
    - -1 if the corresponding element of x is less than 0.

```
>>a = input('coefficient of x^2:')
>>b = input('coefficient of x:')
>>c = input('constant :')
>>D = b^2-4*a*c;
>>if (D>0)
>>    sprintf('two distinct real number solutions')
>>elseif (D==0)
>>    sprintf('repeated real number solution')
>>else
>>    sprintf('neiter of the solutions are real number')
>>end
```

→ If statement → switch statement



## ❖ for Loops

- Calculating multiplication of an array elements

1.

```
>>m = 1;  
>>for n = [2 3 5 7 11 13 17 19]  
>>    sprintf('m = %d\t', m)  
>>    sprintf('n = %d\t', n)  
>>    m = m*n  
>>end
```

m = ??

2.

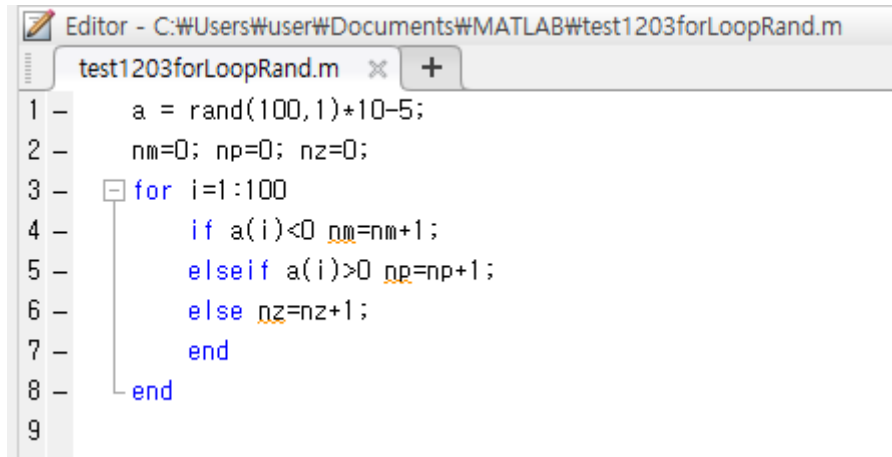
```
>>m = 1;  
>>for n = [2, 3; 5, 7; 11, 13; 17, 19]  
>>    sprintf('m = %d\t', m)  
>>    sprintf('n = %d\t', n)  
>>    m = m*n  
>>end
```

m = ??

## ❖ for Loops

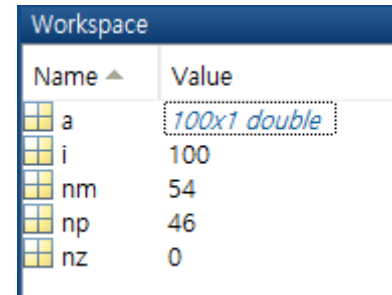
- Counting array elements
- Random function, rand(100,1) generates 100x1 column vector of uniformly distributed numbers in the interval (0,1)

&lt;source code&gt;



```
Editor - C:\Users\user\Documents\MATLAB\test1203forLoopRand.m
test1203forLoopRand.m
1 - a = rand(100,1)*10-5;
2 - nm=0; np=0; nz=0;
3 - for i=1:100
4 -     if a(i)<0 nm=nm+1;
5 -     elseif a(i)>0 np=np+1;
6 -     else nz=nz+1;
7 -     end
8 - end
9
```

&lt;output&gt;



Name	Value
a	100x1 double
i	100
nm	54
np	46
nz	0

❖ Due date : 2018/12/10 23:59PM

- Consider the array A.

- $$A = \begin{bmatrix} 3 & 5 & -4 \\ -8 & -1 & 33 \\ -17 & 6 & -9 \end{bmatrix}$$

- By using a for loop with conditional statements, write a program that computes the array B by computing the natural logarithm of all the elements of A whose value is no less than 1, and adding 20 to each element that is equal to or greater than 1.

Please upload the m file and the result ( capture the command window) on the Black board.  
Make your mfile name with "student\_number\_name"