Lecture #13

1. Hand in HW 7, Take Quiz 5
2. Pick up new Homework 8 - due next Monday (04.05)
3. Today's Lecture:
   - Control flow - Section 3.5
   - User-defined functions - Section 3.6
   - Random numbers - Sections 3.2 - 3.3

Section 3.5 –Control Flow

```
if expression
        statements
end
```

p. 70

```
if expression
        statements
else
        statements
end
```

p. 74

```
if expression
        statements
elseif expression
        statements
elseif expression
        statements
else
        statements
end
```

p. 74

Section 3.5 –Control Flow (cont'd)

```
while expression
        statements
end
```

p. 77

Used to repeat a set of statements as long as the specified expression is true.

RELATIONAL OPERATORS

- For comparing matrices of same size
- Done element by element
- If result is true … new matrix contains 1 in that position
- If result is false… new matrix contains 0 in that position

RELATIONAL OPERATORS

<	less than
<=	less than or equal to
>	greater than
>=	greater than or equal to
==	equal to
~=	not equal to

p. 71

Section 3.5 –Control Flow (cont'd)

Used in m-files

```
if expression
        statements
end
```

p. 70

```
% create an m-file and try this:
clear
count = 0;
G = input('Please enter G: ');
if G < 50
        count = count + 1;
end
count          % shows value of count
```

Run it a couple of times inputting different values for G.
Try scalar Gs & non scalar Gs.

Section 3.5 –Control Flow (cont'd)

```
% create an m-file and now try this
clear
count = 0;   A=[-2, 4, -6, -1];   B=[3, 5, 1, 0];
C=A<B;
disp (C);
if A < B
        count = count + 1;
else
        count = count - 1;
end
count
```

Try again with A=[-2, 4, -6, 0]

Explain the results.

LOGICAL OPERATORS

Used to combine logical expressions.

Logical operators allow a comparison of the 0-1 matrix results from relational operators.

For example: (A<B) & (B<C)

p. 71

LOGICAL OPERATORS

not	~A
and	A&B
or	A|B
xor	xor(A,B)

TRUTH VALUES OF LOGICAL OPERATORS

A	B	~A	A&B	A|B	AxorB
0	0	1	0	0	0
0	1	1	0	1	1
1	0	0	0	1	1
1	1	0	1	1	0

Logic & Relational Expressions

If you want the expression:	You must write in MATLAB:
$0 \leq x < 10$	`(0 <= x ) & (x <10)`
$x \geq 10$ or $y < 10$	`(x>=10) | (y < 10)`

If you want to do a test like:	You must write in MATLAB:
if x is equal to y plus 100	`if (x==y+100)`

This whole expression is evaluated as TRUE (=1) or FALSE (=0)

Let's try some logical operators

```
A= [0,  1,  1];    B= [0,  0,  1];
% example1 ..    or
A|B
% example 2  … not
~A
% example 3  … and
A&B
% example 4 … exclusive or
xor(A,B)
% Try each example with A= [ 0     2     -5]
```

Section 3.5 –Control Flow (cont'd)

```
for index = expression
        statements
end
```

p. 76

- Used to repeat statements a specific # of times (often to "cycle through" all elements of an array or matrix)
- index must be a variable

```
clear;          % try this example
A= [0,  5.5,  -231, 15, -1, 20, -5];
count=0;
for x=1:length(A)
    if A(x)<0
        count=count+1;
    end
end
count
```