CECS 174 – Lecture 10 – If / Else Statements

If Statements –

An if statement is used to allow your programs to decide whether or not it should execute a set of steps. The structure of an if statement is made up of two parts: a condition (which evaluates as either true or false), and a set of statements that execute if the condition is true.

```
if (condition)
{
     //statements that execute if condition is true
}
```

Example - mutually exclusive:

```
int num1 = in.nextInt();

if ( num1 > 0 )
{
     System.out.println("The value is positive");
}

if ( num1 <= 0 )
{
     System.out.println("The value is negative or 0");
}
```

Example – not mutually exclusive – several of the following may be true:

```
num1 = in.nextInt();

if ( num1 > 50 )
{
     System.out.println("Num1 is more than 50");
}

if ( num1 > 10 )
{
     System.out.println("Num1 is more than 10");
}

if ( num1 != 25 )
{
     System.out.println("Num1 is not 25");
}

if ( num1 <= 0 )
{
     System.out.println("Num1 is negative or 0");
}
```

If-Else Statements –

An if-else statement is used to allow your programs to decide whether it should execute one set of steps or another. The structure of an if-else statement is made up of three parts: a condition (which evaluates as either true or false), a set of statements that execute if the condition is true, and another set that execute if the condition is false.

```
if (condition)
{
     //statements that execute if condition is true
}
else
{
     //statements that execute if condition is false
}
```

Example:

```
int num1 = in.nextInt();

if ( num1 > 0 )
{
     System.out.println(“The value is positive”);
}
else
{
     System.out.println(“The value is negative or 0”);
}
```

Example:

```
if ( num1 == 7 )
{
     System.out.println(“Num1 is 7”);
}
else
{
     System.out.println(“Num1 is not 7”);
}
```

Example:

```
if ( num1 >= 7 )
{
     System.out.println(“Num1 is 7 or more”);
}
else
{
     System.out.println(“Num1 is less than 7”);
}
```

Logical Operators –

Truth Tables –

AND - &&

A	B	A && B
true	true	true
true	false	false
false	true	false
false	false	false

OR - ||

A	B	A || B
true	true	true
true	false	true
false	true	true
false	false	false

NOT - !

A	!A
true	false
false	true

The order of precedence for logical operators is: !, &&, ||.

Logical operators are used to create more complex Boolean expressions:

```
int x = in.nextInt();
int y = in.nextInt();

boolean value1 = ( x < 10 ) && ( y < 10 );

if (x > 10 || y > 10)
{
     System.out.println(“X or Y is more than 10”);
}
else
{
     System.out.println(“X and Y are less than 10”);
}

if (x > 10 && y > 10)
{
     System.out.println(“X and Y are more than 10”);
}
else
{
     System.out.println(“X or Y is less than 10”);
}
```