

ED1021 - Introduction to computation and visualisation

L6 - Conditional Statement

Ramanathan Muthuganapathy (<https://ed.iitm.ac.in/~raman>)

Course web page: <https://ed.iitm.ac.in/~raman/introcomp.html>

Moodle page: Available at <https://courses.iitm.ac.in/>

When are they used

- To check for some conditions
- To route the flow according to the conditions
- Stop the running of the code when ill conditions are encountered.
- Done using if - else - else if

Example program

ifelse.c

```
#include <stdio.h>
//Printing the greater of the two numbers
int main( )
{
    int a, b;
    scanf("%d %d", &a, &b);
    if (a > b) {
        printf("a is greater than b %d \n", a);
    }
    else {
        printf("b is greater than a %d \n", b);
    }
}
```

Syntax for if-else - Conditional statement

ifelse.c

```
if (L. E.)    // L. E. = Logical Expression
```

```
{
```

```
    St1;
```

```
    St2;
```

```
    .....
```

```
}
```

```
else
```

```
{
```

```
    St11;
```

```
    St12;
```

```
    .....
```

```
}
```



Block Statement - Enclosed within curly braces

Logical Expression

uses relational operators

- $>$
 - $<$
 - $>=$
 - $<=$
 - $==$ ('equal to' not assignment)
 - $!=$ (not 'equal to')
- Examples for logical exp.
 - $a > b$
 - $a >= b$
 - $a == b$
 - $a != 100$

if-else How it works?

L6_ifelse.c

```
int main( )
{
    StA;
    if (L. E.)
    {
        St1;
        St2;
        .....
    }
    else
    {
        St11;
        St12;
        .....
    }
    StX;
}
```

Sequential execution - Line by Line

StA gets executed

Checks for L.E.

if the L.E. is true, then the St1 get executed.

if not, the block/compound statements St11, ...
get executed.

StX then gets executed.

NOTE: Anything that is NON-ZERO is
considered as true

Style: Indentation is very important.

Demo: Find the largest of two numbers? (L6_ifelse.c)

CW: WAP to print whether a given integer is odd or even (use % operator)?

Nested if-else

L6_nestedifelse.c

```
int main( )
{
    StA;
    if (L. E.1)
    {
        if (L.E.2) {
            .....
        }
        else {
            .....
        }
    }
    else
    {
        .....
    }
}
```

Sequential execution - Line by Line

StA gets executed

HW: Draw the flow chart for the Nested if-else.

NOTE: Anything that is NON-ZERO is
considered as true

Style: Indentation is very important.

HW (use nested if-else)

(1) WAP to find the largest of three numbers?

(2) WAP to find the largest of four numbers?

(3) How to use elseif? WAP to use the same?

Logical Operators

More control statements

- Three logical operators - !, &&, ||
- ! (exclamation symbol) is called NOT operator (negation or complement)
- && - 'And' operator (intersection)
- || (double straight line) - 'or' operator (union)
- They can be used to form logical expression

&& Operator

Logical And

L.E. = Exp1 && Exp2

L.E. is true only if both Exp1 and Exp2 are true,
otherwise L.E. is false.

```
int main( )
{
    int a , b;
    scanf("%d %d", &a, &b);
    if (a > 5 && b > 6.)
    {
        St1;
        St2;
        .....
    }
    else
    {
        St11;
        St12;
        .....
    }
    StX;
}
```

|| Operator

Logical or

L.E. = Exp1 || Exp2

L.E. is true only if Exp1 or Exp2 is true,
otherwise L.E. is false.

```
int main( )
{
    int a , b;
    scanf("%d %d", &a, &b);
    if (a > 5 || b > 6)
    {
        St1;
        St2;
        .....
    }
    else
    {
        St11;
        St12;
        .....
    }
    StX;
}
```

! Operator

Logical Not

Say, you have Exp1

if Exp1 is true, then !Exp1 is false

if Expr1 is false, then !Exp1 is true.

```
int main( )  
{  
    // HW: Fill this up for NOT operator  
}
```

Combination of logical operators

any of them can be combined

L.E. = Exp1 && Exp2 || Exp3 && Exp4

How L.E. is evaluated?

```
int main( )
{
    int a , b;
    scanf("%d %d %d %d", &a, &b, &c, &d);
    if (a > 5 && b > 6 || c > 10 && d < 5)
    {
        St1;
        St2;
        .....
    }
    else
    {
        St11;
        St12;
        .....
    }
    StX;
}
```

**CW: Update operator
precedence table.**

Ternary operator

?:

```
c = (a > b) ? a : b;
```

This is equal to the following st:

```
if (a > b)
```

```
    c = a;
```

```
else
```

```
    c = b;
```

NOTE: In case of only one statement in if (or else/elseif), there is no need for brackets.

Ternary operator - Syntax

?:

`var = (L.E.) ? (var1/exp1) : (var2/exp2);`

Largest of three numbers a, b, c:

`d = (a > b) ? a : b;`

`e = (c > d) ? c : d;`

Will e be the largest of the three numbers?

HW: Write the two statements as a 'single' statement.

HW: WAP to find the largest of four numbers using ternary operator.

switch statement with break

```
int main( )
{
    // E.1 - expression must yield integers o/p
    switch (E.1)
    {
        case const1:
            st1;
            ....
            break;
        case const2:
            st.....
            .....
            break;
        default:
            .....
            break;
    }
}
```

Sequential execution - Line by Line
StA gets executed

HW: Draw the flow chart for the Nested if-else.

NOTE: Anything that is NON-ZERO is
considered as true
Style: Indentation is very important.

switch statement

Example

```
int main( )
{
    int a, b, ex; // Take values for a and b
    ex = a + b;
    switch (ex)
    {
        case 1:
            st1;
            ....
            break;
        case 4:
            st.....
            .....
            break;
        default:
            .....
            break;
    }
}
```

HW: Draw the flow chart depicting switch-case

Style: Indentation is very important.

switch-case

a simple example

```
int main( )
{
    int a, b;
    scanf("%d", &a);
    b = a % 2;
    switch (b)
    {
        case 0:
            printf("b is even\n");
            break;
        case 1:
            printf("b is even\n");
            break;
        default:
            break;
    }
}
```