

Chapter 6 Introduction to Programming

Structure of a C++ program

```
#include<iostream>
using namespace std;
int main()
{
    statements;
    -----
    -----

    return 0;
}
```

Every c++ program starts execution from main() function and ends in main() function

comment lines in c++ :- Non executable statements in a program are called comment lines.

Single line comment line :- // is used for single line comment line.

Multiline comments :- /* and */ are used for comment multiline comment line.

Variable Initialisation :- General format is

data_type variable=value;

data_type variable(value);

const – The access modifier :- The keyword const is used to create constants whose value can never be changed during program execution.

Type Modifiers:- Used to change the size or range of data types.

They are **signed, unsigned, long and short**

Name	Description	Size	Range
char	Character	1 byte	signed: -128 to 127 unsigned: 0 to 255
short int (short)	Short Integer	2 bytes	signed: -32768 to 32767 unsigned: 0 to 65535
int	Integer	4 bytes	signed: -2147483648 to 2147483647 unsigned: 0 to 4294967295
long int (long)	Long integer	4 bytes	signed: -2147483648 to 2147483647 unsigned: 0 to 4294967295
float	Floating point number	4 bytes	$-3.4 \times 10^{+/-38}$ to $+3.4 \times 10^{+/-38}$ with approximately 7 significant digits
double	Double precision floating point number	8 bytes	$-1.7 \times 10^{+/-308}$ to $+1.7 \times 10^{+/-308}$ with approximately 15 significant digits
long double	Long double precision floating point number	10 bytes	$-3.4 \times 10^{+/-4932}$ to $+3.4 \times 10^{+/-4932}$ With approximately 19 significant digits

More Operators

1. Arithmetic assignment operators / Short-hand operators(+=, -=, *=, /=, %=)

$x+=10$ means $x=x+10$

2. Increment(++) and decrement(--) operators- Increment operator will add 1 and decrement operator will reduce 1 from the variable

postfix and prefix form :- postfix form will use 'use then change' method and prefix form will use 'change then use' method.

Precedence of operators

Priority	Operations
1	() parentheses
2	++, --, !, Unary+, Unary -, sizeof
3	* (multiplication), / (division), % (Modulus)
4	+ (addition), - (subtraction)
5	< (less than), <= (less than or equal to), > (greater than), >= (greater than or equal to)
6	== (equal to), != (not equal to)
7	&& (logical AND)
8	(logical OR)
9	? : (Conditional expression)
10	= (Assignment operator), *=, /=, %=, +=, -= (arithmetic assignment operators)
11	, (Comma)

Type conversion

In an expression if the operands are of different types it should be changed to one data type it is called type conversion. There are two methods

a. Implicit type conversion (Type promotion) :- Compiler will automatically change the lower data type to higher data type.

b. Explicit type conversion (Type casting) :- We can change the data type by writing the data type in brackets in an expression.