

## Syntax and Structure of C++ program

### First C++ program

```
#include <iostream.h>

using namespace std;

int main()
{
    cout << "Hello this is C++";
}
```

**Header files** are included at the beginning just like in C program. Here `iostream` is a header file which provides us with input & output streams. Header files contained predeclared function libraries, which can be used by users for their ease.

**Using namespace std**, tells the compiler to use standard namespace.

Namespace collects identifiers used for class, object and variables. Namespace can be used by two ways in a program, either by the use of using statement at the beginning, like we did in above mentioned program or by using name of namespace as prefix before the identifier with scope resolution (`::`) operator.

Example: `std::cout << "A";`

**main()**, is the function which holds the executing part of program its return type is `int`.

**cout <<**, is used to print anything on screen, same as `printf` in C language. `cin` and `cout` are same as `scanf` and `printf`, only difference is that you do not need to mention format specifiers like, `%d` for `int` etc, in `cout` & `cin`.

## Datatypes and Modifiers in C++

Datatypes are used to define type of variables and contents used. Data types define the way you use storage in the programs you write. Data types can be of two types:

1. Primary or Built-in or Fundamental data type
2. Derived data types
3. User-defined data types

### 1. Primary or Built-in Data Types or Fundamental data types:

These data types are built-in or predefined data types and can be used directly by the user to declare variables. example: int, char, float, bool, etc. Primitive data types available in C++ are:

1. Integer
2. Character
3. Boolean
4. Floating Point
5. Double Floating Point

**Integer:** The keyword used for integer data types is int. Integers typically require 4 bytes of memory space and range from -2147483648 to 2147483647.

**Character:** Character data type is used for storing characters. The keyword used for the character data type is char. Characters typically require 1 byte of memory space and range from -128 to 127 or 0 to 255.

**Boolean:** Boolean data type is used for storing Boolean or logical values. A Boolean variable can store either true or false. The keyword used for the Boolean data type is bool.

**Floating Point:** Floating Point data type is used for storing single-precision floating-point values or decimal values. The keyword used for the floating-point data type is float. Float variables typically require 4 bytes of memory space.

**Double Floating Point:** Double Floating Point data type is used for storing double-precision floating-point values or decimal values. The keyword used for

the double floating-point data type is double. Double variables typically require 8 bytes of memory space.

Example:

```
char var = 'A';    // character type
```

```
int var = 1;       // integer type
```

```
float var = 3.14159; // floating point type
```

```
double var = 6e-4;  // double type (e is for exponential)
```

```
bool var = true;    // Boolean (true or false)
```

use sizeof() operator to check size of data types.

## **2. Derived Datatypes:**

Derived data types that are derived from the primitive or built-in datatypes are referred to as Derived Data Types. These can be of four types namely:

1. Function
2. Array
3. Pointer
4. Reference

## **3. User defined or Abstract data types:**

These are the type, that user creates as a class or a structure. In C++ these are classes whereas in C language user-defined datatypes were implemented as structures.

1. Class
2. Structure
3. Union
4. Enumeration
5. Typedef defined Datatype

## Modifiers in C++

In C++, special words (called modifiers) can be used to modify the meaning of the predefined built-in data types and expand them to a much larger set. There are four datatype modifiers in C++, they are:

1. long
2. short
3. signed
4. unsigned

The above-mentioned modifiers can be used along with built in datatypes to make them more precise and even expand their range.

Below mentioned are some important points you must know about the modifiers,

**long and short** modify the maximum and minimum values that a data type will hold.

A plain int must have a minimum size of **short**.

Size hierarchy : short int < int < long int

Size hierarchy for floating point numbers is : float < double < long double

**long float** is not a legal type and there are no short floating point numbers.

**Signed** types includes both positive and negative numbers and is the default type.

**Unsigned**, numbers are always without any sign, that is always positive.

Signed	Unsigned	Long	Short
Integer	Integer	Integer	Double
Char	Char	Double	
Long – Prefix	Short – Prefix		