

Lecture 2: Write Your First Program in C++

CodeHelp -

This is the 2nd lecture of our DSA series. We will see how a program runs and what does each line mean in this lecture. This is how the compilation process works. So, it is a compiler which makes it possible for us to run a source code. Coding Ninjas has supported us in providing this amazing DSA course to you. Courses are available in both Hindi and English. You need an IDE to write, run and execute a code. There are many IDEs out there - CodeBlocks, VsCode and many more - which you do not even need to install. Replit is one of them. You can directly create a C++ there and run it using Replit. In C++, many namespaces are there which have cout functions in them so if we want to use cout, we have to add this file to our code. We use these signs to output the result into standard namespace. If you will use > > by mistake by mistake, then it will give you a compilation error. So we use this operator to display output.

We are going to talk about data types and variables. Whenever you have to store any information or data in the memory, you have to mention what type of data it is and how much memory it will take. For eg- int a = 5, a is a variable in memory which is of int type and it has value 5 in it. We have certain rules which we have to follow when we are giving names. There is a function named sizeof, which will tell you about the space being used by any variable. In memory, first 29 bits will have 0, and last 3 bits will be 101, so in memory the first 29 bits will have 0 and the last 3 bits will have 101. We have a way to store multiple characters but we will see that in upcoming lectures. In C++, all the characters are mapped to some ASCII value in memory. We will just store binary representation of ASCII value of any character. If we have 4 different bytes with some binary value, how will compiler know whether it is a single int value or 4 separate char values? Compiler will come to know from data types which we used when we declared any variable. This process of converting one data type into another is known as type casting.

There are few steps which I have to follow while storing a negative number. Let's say the number is -5. 1st - Ignore the negative sign. 2nd - Convert into binary representation = 101. 3rd - Take 2's complement and store it. Now we can store negative numbers simply too, by just using their first bit as their sign. With this, our maximum value will be limited to $2^{31}-1$ instead of $2^{32}-1$. 2/5 will give 0 and not 0.4, why? because you are storing your answer in an int variable. To get answer in float, store it in a float variable. If you want to check whether A is equal to B or not, I can do that using this (a==b) relational operators =, >, <, != is an assignment operator, because we are assigning 3 to a. Logical operators have & and & is used to check multiple conditions. We have got the basic idea of operators. We know what things to keep in mind while dividing, how to store negative numbers & operators and ! operators. Now let's move ahead towards bitwise operators. I will teach these when we will use them in upcoming sessions. We will do Bitwise operators later.

Preview from Notesale.co.uk
Page 1 of 1