EXPERIMENT NO 1

Introduction to Programming

Objectives:

- To know the importance of programming.
- To know about C++ compilers.
- To learn how to write a simple code in C++.
- To learn what does the compiler do in different situations.
- To learn how to add comments in your code.
- To learn how to take input from user.
- To learn how to display anything (output) on console.

Equipment required:

- Dev-C++/Eclipse/Visual Studio installed in PC/Windows

DISCUSSION

1. Introduction

Computer is organized in different units in which the basic units are input, output, memory, and CPU. Input unit provide data and instructions to the CPU. Memory stores the data and instructions; CPU executes the instructions and pass the results of the execution to the output.
Algorithm is a well-define and ordered set of instructions which lead to a solution within finite number of steps. Compiler is a program that translates high-level language to machine language and creates an executable file. There are many compilers that we can use to code C++, i.e., Eclipse, Dev, Visual Studio, cpp.sh online compiler etc.

2. Pre-Lab

2.1 Writing our First Program

Code

```
#include <iostream>
using namespace std;

// main() is where program execution begins.
int main() {
   cout << "Hello World"; // prints Hello World
   return 0;
}
```

Whenever we write a C++ code, we will always include 3 things.

- #include <iostream>
- using namespace std;

```
o  int main ()
   {
        "Write your code here."
   return0;
   }
```

- **#include <iostream>**
  #include <iostream> tells the pre-processor to include the iostream standard file. This specific file (iostream) includes the declarations of the basic standard input-output library in C++, and it is included because its functionality is going to be used later in the program. The C++ language defines several other headers i.e., #include <string>, all these libraries contain information that is either necessary or useful to our program in some way.

- **using namespace std;**
  "using namespace std" means we use the namespace named std. "std" is an abbreviation for standard. So that means we use all the things with in "std" namespace. If this namespace is not used, then computer finds for the "cout", "cin" and "endl" etc. Computer cannot identify those and therefore it throws errors.

- **int main ()**
  The main function serves as the starting point for program execution. It usually controls program execution by directing the calls to other functions in the program. A program usually stops executing at the end of main, although it can terminate at other points in the program for a variety of reasons. Specifying an 'int' with the 'main' means that our function needs to return some integer at the end of the execution, and we do so by returning 0 at the end of the program. 0 is the standard for the "successful execution of the program".

- **curly "{}" brackets of int main ()**
  Until now all lines are compulsory for every program while this curly bracket is the place where we will write our main logic for different programs. To display anything on console we have to write:
  cout<<"whatever we want to display on console";
  Anything written between these commas " " will be displayed on the console.

2.2 Some general rules for printing

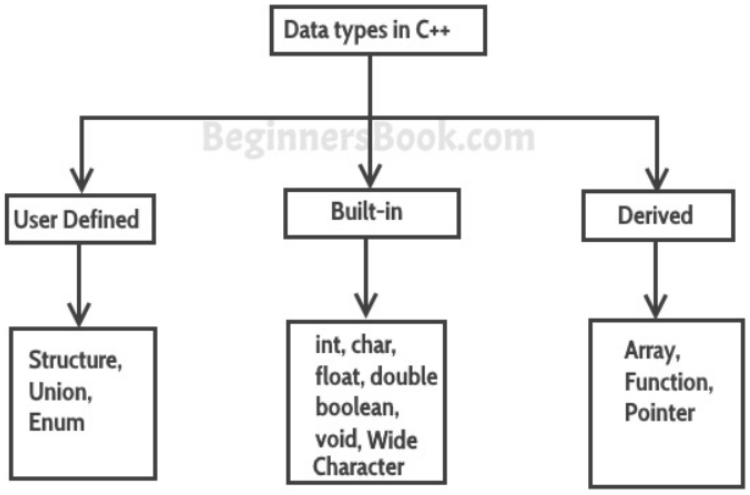
Description	Command	Output
Printing double quotes (")	cout<<"Hello \"World\"";	Hello "World"
Printing back slash (\ )	cout<<"Hello \\World";	Hello \ World
Printing tab space	cout<<"Hello \t World";	Hello World
jump to new line	cout<<"Hello \nWorld";	Hello World
jump to new line	cout<<"Hello"<<endl; cout<<"World";	Hello World

2.3 Comments

Commenting involves placing human readable descriptions inside of computer programs detailing what the code is doing. Proper use of commenting can make code maintenance much easier, as well as helping make finding bugs faster. Further, commenting is very important when writing functions that other people will use.
To include comment at particular line just write \\any comment here

2.4 Variables

Variables are used to store data inside them. The name of a variable can be composed of letters, digits, and the underscore character. It must begin with either a letter or an underscore. Upper and lowercase letters are distinct because C++ is case-sensitive.



Variable type (Data type)	How do define in C++
Integer	int w;
Decimal	float x; or double x;
A Single Character	char y;
Boolean	bool z;

Variables can be used to either put data directly into them or by initializing them and take data from the user. Suppose we want to take an integer value from the user, then we will first initialize integer variable like this:

int x; \\ x is the name of variable.

Now to take input just write “cin>>x” whatever user enters will be stored in the variable x.
We can perform any operation on the variables.

2.5 Arithmetic Operations

Description	Operator	Answer
Addition	cout<<5+2	7
Subtraction	cout<<5-2	3
Multiplication	cout<<5*2	10
Integer Division	cout<<5/2	2
Modulus (Remainder)	cout<<5%2;	1
Division including float	Cout<<5.0/2	2.5

2.6 Assignment Operator

Solve expression on right side and assign the result in variable of left side. For example.

```
int main ()
{
	int x, y;
	cout<<"Enter a number: "
	cin>>x;
	y = x*x;
	cout<<"The square of " << x << " is: " << y;
}
```

3. Post-Lab (Lab Tasks)

1. Display the table of any number on the output console using only cout statements.
2. Write a program that displays the following output on the console.
   She asked, \"Are you alright?"\
   He replied. "Yes! I'm fine"

END