Lecture 4

C++ Language Basics:

When we consider a C++ program, it can be defined as a collection of objects that communicate via invoking each other's methods. Let us now briefly look into what a class, object, methods, and instant variables mean.

- Object - Objects have states and behaviors. Example: A dog has states -
color, name, breed as well as behaviors - wagging, barking, and eating.
An object is an instance of a class.
- Class - A class can be defined as a template/blueprint that describes the
behaviors/states that object of its type support.
- Methods - A method is basically a behavior. A class can contain many
methods. It is in methods where the logics are written, data is
manipulated and all the actions are executed.
- Instant Variables - Each object has its unique set of instant variables.
An object's state is created by the values assigned to these instant
variables.

C++ Program Structure:
Let us look at a simple code that would print the words *Hello World*.

```
#include <iostream>
using namespace std;
// main() is where program execution begins.
int main()
{
cout << "Hello World"; // prints Hello World
return 0;
}
```

Let us look at the various parts of the above program:

1. The C++ language defines several headers, which contain information
that is either necessary or useful to your program. For this program, the
header <iostream> is needed.

2. The line using namespace std; tells the compiler to use the std
namespace. Namespaces are a relatively recent addition to C++.

3. The next line '// main() is where program execution begins.' is a
single-line comment available in C++. Single-line comments begin with //
and stop at the end of the line.

4. The line int main() is the main function where program execution begins.

5. The next line cout << "This is my first C++ program."; causes the
message "This is my first C++ program" to be displayed on the screen.

6. The next line return 0; terminates main() function and causes it to return the value 0 to the calling process.

Compile & Execute C++ Program:

Let's look at how to save the file, compile and run the program. Please follow the steps given below:

1. Open a text editor and add the code as above.
2. Save the file as: hello.cpp
3. Open a command prompt and go to the directory where you saved the
file.
4. Type 'g++ hello.cpp' and press enter to compile your code. If there are no
errors in your code the command prompt will take you to the next line
and would generate a.out executable file.

5. Now, type 'a.out' to run your program.
6. You will be able to see ' Hello World ' printed on the window.

```
$ g++ hello.cpp
$ ./a.out
Hello World
```

Make sure that g++ is in your path and that you are running it in the directory
containing file hello.cpp.
You can compile C/C++ programs using makefile. For more details, you can
check our ‘Makefile Tutorial’.
Semicolons & Blocks in C++
In C++, the semicolon is a statement terminator. That is, each individual
statement must be ended with a semicolon. It indicates the end of one logical
entity.
For example, following are three different statements:

```
x = y;
y = y+1;
```

C++
8

```
add(x, y);
```

A block is a set of logically connected statements that are surrounded by
opening and closing braces. For example:

```
{
cout << "Hello World"; // prints Hello World
return 0;
}
```

C++ does not recognize the end of the line as a terminator. For this reason, it
does not matter where you put a statement in a line. For example:

```
x = y;
y = y+1;
add(x, y);
```

is the same as

```
x = y; y = y+1; add(x, y);
```

C++ Identifiers

A C++ identifier is a name used to identify a variable, function, class, module, or
any other user-defined item. An identifier starts with a letter A to Z or a to z or
an underscore (_) followed by zero or more letters, underscores, and digits (0 to

9).
C++ does not allow punctuation characters such as @, $, and % within
identifiers. C++ is a case-sensitive programming language.
Thus, **Manpower** and **manpower** are two different identifiers in C++.
Here are some examples of acceptable identifiers:

```
mohd zara abc move_name a_123
myname50 _temp j a23b9 retVal
```

C++ Keywords
The following list shows the reserved words in C++. These reserved words may not be used as constant or variable