CHAPTER 2
INTRODUCTION TO C++

[2HRS.]

INTRODUCTION TO C++:

C++ is a high level programming language that supports OOP. It is a improved C, with data abstraction ,OOP, and generic programming support. C++ was to provide simula's class feature for object oriented program development together with C's feature for system programming.

Part of C++ programs are easily reusable and extensible. C++ is read as C plus plus. The ++ can be read as "next", "successor", or "increment" thought it is pronounced as "plus plus".

C++ was used to develop high performance software such as mysql, window XP, Adobe products, etc. Like C, C++ comprises both feature of high level and low level languages so it can be regarded as middle level language. The syntax of C++ influences recent languages like JAVA, C# etc.

Example Program of C++:

```
#include<iostream.h>

#include<conio.h>

void main(){

cout<<“Hello World”;

getch();

}
```

THE NEED OF C++:

- The software developer had to solve more complex problems for big organzations like banking, insurance, railway system, civil aviation, military and so forth. In order to solve such complex problems and to model the real world perfectly, C++ was developed.
- C is not a secure programming language by any stretch of the imagination; no bounds checking on arrays leads to lots of exploitable behaviour. C++ at least gives you containers that throw exceptions if you try to access outside their currently defined range.
- Memory management in C is very labor intensive and error prone compared to the tools C++provides. So to overcome this problem, C++ comes into existence.
- C++ can handle large complex problems to develop high performance and large scale application using object oriented approach.

C VERSUS C++:

C	C++
C supports procedural programming paradigm for code development.	C++ supports both PP and OOP paradigm; therefore C++ is also called a hybrid language.
When compared to C++, C is subset of C++.	C++ is a superset of C. C++ can run most of C code while C cannot run C++ code.
C does not support polymorphism, encapsulation and inheritance.	C++ supports polymorphism, encapsulation and inheritance.
C does not support function and operator overloading.	C++ supports both function and operator overloading.
C uses function for input and output .For example: scanf and printf.	C++ uses objects for input and output. For example: cin and cout.
C does not provide direct supports for error handling or exception handling.	C++ provides support for exception handling.

FEATURES OF C++:

C++ consists of common features of object oriented language and also it has its own features too .Following are some of the important features of C++.

1. **Namespace:** The namespace is used for logical grouping of program elements like variables, classes ,functions, etc. If some program elements are related to each other they can be put into a single namespace. The namespace is useful to develop large application by different programmers independently with out fear of conflict of naming identifiers.

2. **Classes:** A class is a user-defined data type which can be used to define variable (instance) of its type called object. The class logically combines data and function in a single unit . It is the most important aspect of C++.

3. **Access Controller:** private, public, protected keywords are used for access control of members within class. The members declared as private are hidden from other part of the program and are visible by the member function within the class only. The public access specifier allows function or data members in public section to be accessible to other part of the program too. The data member protected in a class makes function or data member accessible from the function of the derived class too.

4. **Function Overloading:** When the same function name is used for different operation, it is called as function overloading. This is useful for manipulating same nature of problems but with different numbers of arguments or types.

5. Operator Overloading: The operator overloading feature of C++ extends the meaning of an operator for user-defined types. The operators such as +, -, +=, >, >> etc are designed to operate only on standard data type.

6. Generic Programming: The template feature of C++ provides generic programming by allowing the development of reusable software component such as function, classes etc supporting different data types in aa single framework.

7. Exception Handling: Exception are error that may occur at run time. Exception handling can be used to support ideas of error handling and fault tolerant computing. They are caused by a wide variety exceptional circumstances, such as running out of memory , not being able to open a file, trying to initialize an object to an impossible value, or using an out-of-bounds index to a vector.

Assignment: History of C++ , Uses of C++.