C++ Programming INTERNSHIP COURSE—

C++ is the most powerful and flexible object oriented programming computer language which is used for operating systems, compilers, interpreters, search engine and graphic programs. We offer this course to help students clear their C++ concepts through a combination of theory and lab practice.

Course objectives:

It covers C++ programming language and its interactions with software design. We offer this advance course with the following objectives:

- Describes the concept of OOPS and its terminology including encapsulation, abstraction, polymorphism, inheritance, etc.
- Tells the advantages of C++ over C
- Show how to declare and define classes and objects
- Introduce the concept of constructors and destructors
- Describes operators, their overloading, comparison, data conversion, etc.
- Detailed theory on inheritance, polymorphism, virtual functions
- Tells about templates, exceptions, streams and strings
- Tells about object oriented systems development

C++ is a highly popular language in today's time and is regarded as the first choice of the professional programmers.

IOTIANHUB INTERNSHIP COURSE

Week-1--

Object Oriented Paradigm

- OOPS...!
- Structured versus Object Oriented Development
- Elements of Object Oriented Programming
- Objects
- Classes
- Encapsulation
- Data Abstraction
- Inheritance
- Polymorphism
- Templates
- Exception Handling

Moving from C to C++

- Scope resolution Operator
- Variables aliases(reference variables)
- Parameters passing by References
- Inline functions
- Function Overloading
- Default Arguments

Classes and Objects

- Introduction
- Structures and Classes
- Class specification
- Class objects
- Class, Objects and memory resources
- Accessing class members
- Defining Member Functions
- Outside member functions as inline
- Accessing member functions with in class
- Data Hiding
- Passing Objects as arguments
- Friend Classes
- Static data members
- Static Functions

Week-2

Constructors and Destructors

- Introduction
- Need of the Constructor
- Parameterized constructor
- Constructor overloading
- Constructor with default arguments
- Name less objects
- Copy constructors
- New and delete operators
- Dynamic initialization through constructor

Operator Overloading

- Introduction
- Over loadable operators
- Unary operator overloading
- Operator return values
- Name less Temporary Objects
- Limitations of Increment and Decrement Operators
- Binary Operator Overloading
- Overloading New and Delete Operator
- Comparison Operators
- Data Conversion
- Conversion between Datatypes
- Conversion between basic and Objects
- Conversion between Objects of different classes
- Assignment operator overloading
- Overloading with friend functions

Inheritance

- Introduction
- Derived class declaration
- Forms of inheritance
- Member Accessibility
- Constructors in derived classes
- Overloaded Member functions
- Abstract classes
- Multilevel Inheritance
- Multiple Inheritances
- Hierarchical Inheritance
- Multipath Inheritance
- Virtual Base Class
- Hybrid Inheritance

Week-3

Virtual Functions & Polymorphism

- Introduction
- Need for virtual functions
- Pointers of derived class objects
- Definitions of Virtual Functions
- Pure Virtual Functions
- Dynamic Binding
- Rules For virtual functions

Templates

- Introduction
- Function Templates
- Overloaded Function Templates
- Multiple Argument Function Templates
- Class Templates
- Class Templates with overloaded operators

Week-4

Exception Handling

- Introduction
- Error Handling
- Exception Handling
- Try, throw, catch
- List of Exceptions
- Specified, Unspecified exceptions
- Handling Uncaught Exceptions

Streams Computation With console

- What are streams?
- Console Streams
- Unformatted, Formatted Console O/P

Manipulating Strings

- Creating(string) objects
- Manipulating String Objects
- Relational Operations
- String Characteristics

Object-Oriented Systems Development

- Procedure-Oriented Development Tools
- Object-Oriented Notations and Graphs
- Steps in Object-Oriented Analysis
- Steps in Object-Oriented Design
- Implementation