



# C++ Training

*A training pack for students & beginners*

Dear Sir/Madam

**Sub: To organize C++ Training in your college.**

This is to bring to your kind notice that **POSITIVE QUADRANT TECHNOLOGIES LLP** is an Indian entity exploring itself in various sectors like Software Development , Augmented Reality , Virtual Reality , IoT , Simulation ,Games Development ,Mobile Applications,3D Modelling Development , Practical Educational Training, Professional Training, Corporate Training, Web & IT Services.

**C++ Training** conceptualized by **some top industry professionals** in association with **POSITIVE QUADRANT TECHNOLOGIES LLP**. It is going to be India's first & biggest workshop series based on this market flag bearer of cloud systems.

This workshop will also provide a platform where young engineers can mould their imagination into reality and feel the excitement first-hand. With this end in view, we extend our support and technical expertise to the young engineers of your College in the form of this workshop. We seek your cooperation and look forward towards a successful execution of this workshop in your college.

We are hoping that you will find this training really interesting for the students. If you have any queries, please get back to us anytime.



# SYLLABUS

## Introduction to Programming

- Program and Programming
- Programming Languages
- Types of software's Operating Systems
- Dos commands
- Basic Linux commands and vi editor
- Compiler, Interpreter, Loader and Linker

## Fundamentals in C++

- History of 'C++'
- Migrating from procedural oriented language
- to object oriented languages Program
- Keywords
- Variables
- Constants
- Data type
- Operators
- Manipulators and uses
- Basic Structure of a 'C++' program





## Control statements

–Conditional Control Statements

–if

–if-else

–nested if-else

–else-if ladder

–Multiple Branching Control

Statement

–switch-case

–Loop Control Statements

–while

–do-while

–for

–Nested Loops

–Jump Control statements

–break

–continue

–goto

–exit

–return

–Programming Examples

–FAQ's





## Pointer array Reference

- pointer variable
- Reference variable/alias variables?
- Reference to Reference variable?
- Reference to array?
- Reference vs normal variable?
- Reference vs pointer variable?
- 1D and 2D Arrays
- What is dynamic memory allocation?
- The new and delete operator
- new vs malloc
- delete vs free
- Dynamic 1D and 2D Arrays



## Function

- What is function ?
- Why function ?
- Advantages of using functions
- Function Prototype
- Defining a function
- Calling a function
- Actual and Formal Arguments
- Types of functions
- Parameter Passing Techniques
- Call by Value



- Call by Reference
- Call by Pointer
- Return statement
- Returning More than one value From A **Function**
- Return by value mechanism
- Return by pointer mechanism
- Return by reference mechanism
- Inline Functions
- Default Arguments
- Function Overloading
- Lambda function.
- Recursion



### **Introduction to oops**

- c structure vs c++ structure
- c++ class vs c++ structure
- Class
- Object
- Encapsulation
- Abstraction
- Polymorphism
- Inheritance
- Message Passing

**Up to 16 Hour's**



## Classes and Objects

- Declaring / defining classes
- Data members and member functions
- Access specifiers : public and private and protected
- Creating objects of a class
- Pointers to object
- Implicit this pointer
- Static data members
- Static member functions
- Passing objects to a member function
- Returning objects from a member function
- Friend functions
- Friend classes
- Nested classes
- Local classes
- The const member functions
- The const objects
- Array of objects
- static objects
- What are inline functions?



**Up to 30 Hour's**



## Constructors and Destructors

- Defining Constructor
- Defining Destructor
- Comparing Constructor Member **Function**
- Default Constructor
- Argument Constructor
- Copy Constructor
- Constructor Overloading
- Default Argument in Constructor
- Anonymous object
- Private Constructor and Destructor
- Local vs Global object



## Operator Overloading

- Need of Overloading
- Defining Operator Overloaded Function
- Operator Overloading Rules
- Overloading Binary Operators
- Overloading Binary Operators using Friend
- Overloading Other Operators
- Overloading Unary Operators
- Overloading Unary Operators using Friend



## **Inheritance and Composition**

- What is inheritance?
- The is-a relationship
- Single Level Inheritance
- Multilevel Inheritance
- Multiple Inheritance
- Name ambiguities under multiple inheritance
- Hierarchical Inheritance
- Hybrid Inheritance
- Multipath Inheritance
- Why virtual base classes?
- Constructor and Destructor in Inheritance.
- Common constructor.
- Delegation
- What is composition?
- The has-a relationship

## **Polymorphism**

- About polymorphism
- Compile time and runtime polymorphism
- Virtual functions.
- Pure virtual function and abstract base class.
- What is RTTI (Run-time Type Information)?
- VPTR and VTABLE.
- Difference between member Function

**Up to 45 Hour's**





### **Overloading and Overriding**

- Object slicing.
- Constructor and virtual function.
- Virtual destructor.
- Destructor with virtual function.

### **Exception handling**

- What is an exception?
- Throwing an exception
- Catching an exception
- Trying for an exception
- Order of catch blocks
- Catching all exceptions
- Nested try blocks
- Rethrowing an exception
- Exception specifications
- What is stack unwinding?
- Exceptions in ctors and dtors
- The unexpected() function
- The terminate() function
- The standard exceptions
- Creating our own exception classes



### **File handling**

- Hierarchy of File Streams
- Using constructor method
- Using open and close member function



method.

–Object as file stream reader and writer

–Both sequential and random file

accessing mechanism.

–Different error handling mechanism in

files

### **Input / Output in C++: Streams**

–Hierarchy of I/O Streams

–Fundamental stream classes and objects

–Standard input and output functions

–Formatting flags and manipulators

### **Working With String**

–Different C string handling library

–string handling using relational operator

–Different string handling function



### **Namespace**

–Creating name space

–Using name space

–Nested namespace and anonymous

Namespace

### **Command line arguments**

–what is command prompt?

–why command line?

–What are command line arguments?



–Programs using command line

### Database operation

–What is database?

–SQL for relational database.

–About API connect to database.

–Database connectivity MySQL.

–Database manipulation using C++

–Process and Threads

–Graphics

–WEB development

–Web basics.(HTML, java script, CSS).

Project

**Up to 60 Hour's**

