KDN
INFOTECH PVT.LTD
DEDICATED TO EXCELLENCE

Diploma In Testing

C course content

Basics content of C Language

- What is C language
- History of C Language
- Features of C Language
- How to Install C
- The Structure of a C Program ○ How to Write C Programs?
- Compilation process in C
- Printf() and scanf() in C

C language Overview

- Veriables in C and Its Types
- Data Types in C
- C laguage Keywods
- Identifiers in C and their type
- Role of operater in C language

- Comments in C and their Types
- Format Specifier in C language
- Esacape sequence
- (ASCII) American Standard Code for information interchange values in C
- Role of a constraints in C
- The Types of Literals that exist in C programs
- Importance of Tokens in C
- Boolean Data Type in C
- Static keyword in C
- Role of programming errors in C and their types
- Role of compile time and run time error in C
- Conditional operater in C language
- Bitwise operater in C and their types

Control Statements Which Used C Language:

- The IF.....ELSE Statement in C and variants of if statement in C language
- IF......ELSE ladder (C switch statement)
- The difference between If ...else and switch statement
- Loops in C programming and their three types
- Do while loop in C
- While loop in C (pre-tested loop)
- For loop in C Language
- Nested loop in C language ○ Infinite loop in C language ○ Role of C break statement
- C continue statement
- goto statement in c
- type casting in c

Function in C

- Advantages of C Function and their Types
- Call by value method in C and call by reference method in C
- Recursion process in C
- Storage classes in C and their Four types

Array in C

- Array: what and why?

- 1D arrays
- 2Darrays
- Mulri Dimentional array
- Dynamic arrays

Pointers in C

- Pointers in C and declaring a pointer
- Double pointer in C
- Dangling pointer in C
- Sizeof() operater in c
- Constraint pointer in c and their syntax
- Void pointer in c
- Dereference pointer (indirection operator) in C and why we use

Dynamic memory in C

- The concept of dynamic memory allocation in c language and their 4 types of functions

String Handling in C

- gets() puts() function in C
- Role of string function
- String Length: strlen function in C language
- The strcpy function
- String concatenation: strcat()
- Compare string function in c …..strcmp()
- The reverse string function…..strrev()
- String lowercase function ….strlwr()
- String uppercase function…strupr()
- String strstr()
- Match function in C

The Structures and union in C

- Advantages of a structure in c
- Defining structure …..typedef keyword in C
- Uses of an array structures in c
- Nested structure in C
- Structure padding in c
- Defining union datatype in c

File Management in C

- Introduction of file management and need of file handling in C
- Fprintf() and fscanf() function
- Writing file fputs() function and reading file fgets() function
- fseek() function in c
- rewind() function in c
- ftell() dunction in c

Dynamic Memory allocation

- ✓ Introduction to Dynamic Memory Allocation
- ✓ Malloc
- ✓ Calloc
- ✓ Realloc
- ✓ Free

• CPP COURSE CONTENTS

C++ Overview

- Features of C++ language
- Installation

- Cpp basic input/output operation
- Variable and its syntax
- Datatypes in CPP(Basic,Derived,Enumaration,user defined)
- List of all keywords in cpp
- The Role of an operators
- The role of an Identifier
- Cpp expression

Control statements of CPP

- Types of statements
- If _else statement
- Cpp Switch statement
- For loop
- While loop
- Do while loop
- Break statement
- Continue statement
- Goto statement
- Commnet statement

Functions in CPP

- call by value and call by reference
- recursion in CPP
- storage classes in cpp

Arrays In CPP

- Advantages And disadvantages of arrays
- Types of an Arrays
- Passing arrays to Function in cpp
- Multidimensional arrays in cpp

Pointers Variable in CPP

- Advantages and uses of pointer

- Sizeof() operater in cpp
- Array of pointer in cpp
- Void pointer in cpp
- How to creat references
- Difference between references and pointers
- Function pointer in cpp
- Memory management in cpp
- Malloc() vs new in cpp
- Free vs delete in cpp

Object class in cpp

- Oops concept in cpp
- Object and class in cpp
- Constructer and their types in cpp
- Copy constructer
- Destructor in cpp
- This pointer in cpp
- Static keyword in cpp
- Struct in cpp
- Enumeration data type
- Friend function in cpp

Inheritance in cpp

- Advantages and cpp supports five types of inheritance
- Aggregation HAS-A relationship)

Polymorphism in cpp

- Types of polymorphism
- Overloading (function and operater)
- Types of overloading in cpp
- Overriding function in cpp
- Virtual function

- **Interfaces in cpp (abstract class)**

- **Data abstraction in cpp**

 Name spaces in CPP

 Strings in cpp

 Advantages and their classes Cpp files and streams

 Cpp getlines()

 Cpp int to string

Functions and Variables

- ✓ Functions: Declaration and Definition
- ✓ Variables: Definition, Declaration, and Scope
- ✓ Variables: Dynamic Creation and Derived Data
- ✓ Arrays and Strings in C++
- ✓ Qualifiers

Classes in C++

- ✓ Defining Classes in C++
- ✓ Classes and Encapsulation
- ✓ Member Functions
- ✓ Instantiating and Using Classes
- ✓ Using Constructors
- ✓ Multiple Constructors and Initialization Lists
- ✓ Using Destructors to Destroy Instances
- ✓ Friendship

Operator Overloading

- ✓ Operator Overloading
- ✓ Working with Overloaded Operator Methods

Initialization and Assignment

- Initialization vs. Assignment
- The Copy Constructor
- Assigning Values
- Specialized Constructors and Methods
- Constant and Static Class Members

Storage Management

- Memory Allocation
- Dynamic Allocation: new and delete

Input and Output in C++ Programs

- Standard Streams
- Manipulators
- Unformatted Input and Output
- File Input and Output

CORE JAVA

BASIC OF JAVA:

- Java - What, Where and Why?
- History and Features of Java 7 And Java 8
- Internals of Java Program
- Difference between JDK,JRE and JVM in Java
- Internal modules of JVM
- Variable and Data Type(user defined and system defined)
- Unicode System in Java
- Naming Convention in java

OOPS CONCEPTS:

- Advantage of OOPs (C++ REVISION)
- Object and Class Difference.

Method Overloading and Overriding.
Constructor with their types.
static variable, method and types of java blocks
this keyword in java
Inheritance Aggregation(IS-A) and Composition(HAS-A) in java
Covariant Return Type in inheritance
super keyword and Final keyword.
Instance Initializer block in java

- Runtime Polymorphism in java
- static and Dynamic binding in java
- Abstract class and Interface difference.
- Downcasting with instanceof operator in java
- Package and Access Modifiers in java
- Encapsulation vs abstraction
- Object class
- Object Cloning with Cloneable interface
- Java Array object and properties.
- Call By Value and Call By Reference in java
- strictfp keyword in java

String Handling

- String importance in java
- Immutable String in java
- String Comparison with their types.
- String Concatenation with methods and operators.
- Substring in java
- Methods of String class
- StringBuffer class And StringBuilder class in Java
- Creating Immutable class in Java
- toString method with object

Exception Handling

- Exception Handling in java
- try and catch block in java
- Multiple catch block in java
- Nested try in java class
- finally block with try and catch block.
- throw keyword in java
- Exception Propagation with checked and unchecked exceptions.
- throws keyword with custom excpetions
- Exception Handling with Method Overriding in java
- Custom Exception or user defined exceptions.

Nested Classes

- Nested Class in java
- Member Inner class in java
- Annonymous or unnamed Inner class
- Local Inner class in method of java class
- static nested class in java
- Nested Interface java

Multitheading

✓ Multithreading in java

- ✓ Life Cycle of a Thread class
- ✓ Creating Thread using different types
- ✓ Thread Schedular behaviour.
- ✓ Sleeping a thread with sleep method in java.
- ✓ Joining a thread with join method
- ✓ Thread Priority techniques.
- ✓ Daemon Thread in java
- ✓ Thread Pooling in java
- ✓ Thread Group in java
- ✓ ShutdownHook method in java
- ✓ multiple task by multiple thread
- ✓ Garbage Collection with GC
- ✓ Runnable class and their methods
- ✓ Synchronization in java thread
- ✓ synchronized method ,block
- ✓ static synchronization in java
- ✓ Deadlock situation in thread
- ✓ Inter-thread Communication with wait notify and notify all methods
- ✓ Interrupting Thread with interrupt method.

Input and output

- ✓ FileOutputStream & FileInputStream API in java I/O
- ✓ ByteArrayOutputStream, SequenceInputStream, BufferedOutputStream, BufferedInputStream
- ✓ FileWriter & FileReader in java I/O
- ✓ CharArrayWriter with their methods.
- ✓ Input from keyboard by InputStreamReader, Console, Scanner.
- ✓ PrintStream, PrintWriter class
- ✓ Compressing and Uncompressing File (zip-unzip files)
- ✓ Reading and Writing data simultaneously using thread
- ✓ DataInputStream and DataOutputStream class.

- ✓ Serialization & Deserialization with file
- ✓ Serialization with IS-A and Has-A inheritance relation
- ✓ transient keyword with serialised object

Networking

- ✓ Socket Programming server and client API
- ✓ URL class in java
- ✓ Displaying data of a web page using methods
- ✓ InetAddress class and their methods
- ✓ DatagramSocket and DatagramPacket in java
- ✓ Two way communication (server-client)

Reflection API

✓ Reflection API in java

Class class and Object class in java Reflection.

✓ newInstance() ,forname(),.class methods for Determining the class object
✓ system defined javap tool
✓ creating custom javap tool
✓ Accessing private method from outside the class using method class.

Collection

✓ Collection Framework API
✓ ArrayList class in java
✓ LinkedList class java
✓ ListIterator interface in java
✓ HashSet class in java
✓ LinkedHashSet class in java
✓ TreeSet class in collection
✓ PriorityQueue class
✓ Map interface in java
✓ HashMap, LinkedHashMap, TreeMap, Hashtable class

✓ Comparable and Comparator interface in java
✓ Properties class with properties file

Java New Features 7/8

✓ Assertion, For-each loop, Varargs, Static Import in java
✓ Lambda expression and functional interface
✓ Autoboxing and Unboxing with wrapper class
✓ Annotation in java

Internationalization

✓ Internationalization in java
✓ ResourceBundle class in java
✓ I18N with Date,Time,Number,Currency

Regular Expression in java 8

✓ Regular Expression API
✓ Pattern and Matcher class in java
✓ Regular Expression operators

Manual Testing

Objectives

- To understand what is testing?
- To understand Software development model.
- To Understand Architectures of software development.
- To learn the features of Software development models.
- To learn major concepts of the testing methodologies.
- To know different approaches to Testing.
- To understand of the types of testing.
- To plan and create test plan
- To execute the test plan.
- To create and manage test cases and defect profiles.
- To build strategies to track testing processes in the bug tracking systems.
- To do document of the test report in the testing enclosure document.

Module 1: Software Testing Introduction

In this module you learn about Importance of testing. Why Testers need industry, software program/application/product: meets the business and technical requirements that guided its design and development works as expected.

- What is testing?
- Importance of testing
- Roles and Responsibilities

- Principles of software testing
- What is Quality?
- How much testing is enough?
- Differences between Manual and Automation Testing.

Module 2: Software Development Life Cycle

In this module you learn about development procedure .SDLC stands for Software development life cycle. It is a process that describes how to develop, design and maintain the software project ensuring that all the functional & user requirement, goals and objective are met.

1. SDLC Phases Requirements Phase.

   Analysis Phase.

   Design phase.

   Coding Phase.

   Testing phase.

   Delivery and Maintenance Phase
2. SDLC Models Waterfall Model.

   V Model

   Agile Model.

   Prototype Model.

   Spiral Model.

Module 3: Software Testing Methodologies

In this module you learn about deferent types of software testing. Software Testing Methodology is defined as strategies and testing types used to certify that the application under test meets client expectations.

- White Box Testing.
- Black Box Testing.
- Grey Box Testing.

Module 4: Test Case Design Techniques

In this module you learn design test cases in such a way that we get the maximum coverage using an optimal set of Test cases. Focus on highlighting the various Methods and Techniques in designing test cases for both Black Box Testingand White Box testing.

Static Techniques:

- Informal Reviews
- Walkthroughs
- Technical Reviews
- Inspection

Dynamic Techniques:

Structural Techniques:

- Statement Coverage Testing
- Branch Coverage Testing
- Path Coverage Testing
- Conditional Coverage Testing
- Loop Coverage Testing

Black Box Techniques :

- Boundary Value Analysis
- Equivalence Class Partition
- State Transition Technique
- Cause Effective Graph
- Decision Table
- Use Case Testing

Experienced Based Techniques:

- Error guessing
- Exploratory testing

Module 5: Levels of Testing

In this module you learn about levels of testing are frequently grouped by where they are added in thesoftware development process, or by the level of specificity of the test.

1. Functional Testing Unit Testing

- Integration Testing
- System Testing
- User Acceptance Testing.
- Sanity/Smoke Testing.
- Regression Test.
- Retest.

2. Non Functional Testing

- Performance Testing
- Memory Test
- Scalability Testing.
- Compatibility Testing.
- Security Testing.
- Cookies Testing.
- Session Testing.
- Recovery Testing.
- Installation Testing.
- Adhoc Testing.
- Risk Based Testing
- 18N Testing.
- L1ON Testing.
- Compliance Testing.

Module 6: Software Testing Life Cycle

In this module learn about in detail description of Test Life Cycle, importance of Test Plan roles and responsibilities of Test Manager, Test Lead, Test Engineer,

1. Requirements Analysis/Design

- Understand the requirements
- Prepare Traceability Matrix

2. Test Planning Object.

- Scope of Testing.
- Schedule.
- Approach.
- Roles & Responsibilities.
- Assumptions.
- Risks & Mitigations
- Entry & Exit Criteria.
- Test Automation.
- Deliverables.

3. Test Cases Design

- Write Test cases
- Review Test cases
- Test Cases Template
- Types of Test Cases

- Difference between Test Scenarios and Test Cases.

4. Test Environment setup

- Understand the SRS
- Hardware and software requirements
- Test Data

5. Test Execution

- Execute test cases
- Defect Tracking and Reporting
  - Types of Bugs.
  - Identifying the Bugs.
  - Bug/Defect Life Cycle.
  - Reporting the Bugs.
  - Severity and priority

6. Test Closure

- Criteria for test closure
- Test summary report

7. Test Metrics

- What is Test Measurements?
- Why Test Metrics?
- Metric Life Cycle.
- Types of Manual Test Metrics

Module 7: QA & QC & Testing

In this module you learn about QA & QC and How to log bugs in Project management tool, how to give severity, priority to bugs. What is Quality Assurance?ϖ What is Quality Control?ϖ Differences of QAϖ & QC & Testing

Test Management with TFS Tool (Learn &

Implement) Defect Tracking Tools (Learn &

Implement) TFS

Selenium Automation Testing

- **What is Selenium? Introduction to Selenium Automation Testing**
- Basic Terminology
- Selenium Features
- Selenium Limitations
- Selenium vs QTP
- Selenium Tool Suite
- **Selenium Integrated Development Environment (IDE)**
- Introduction
- Selenium IDE-Installation
- Selenium IDE Download and Install
- Selenium IDE-Features
- Selenium IDE- First Test Case

- **Selenium IDE- Commands (Selenese)**
  - Types of Selenium Commands
- **Selenium IDE- Creating Test Cases Manually**
  - Insert Commands
  - Executing the Test Script
- **Selenium IDE- Locating Strategies**
  - Locating by Identifier

  - Locating by ID Element
  - Locating by Name
  - Locating by XPath
  - Locating by CSS
  - Locating by DOM
- **What is Selenium WebDriver?**
- **Introduction**
- **Selenium WebDriver- Architecture**
- **Selenium WebDriver- Features**
- **Difference with RC**
- **Selenium WebDriver- Installation**
  - Download and Install Java
  - Download and Configure Eclipse IDE
  - Download Selenium WebDriver Java Client
  - Configure Selenium WebDriver
  - Selenium WebDriver- First Test Case

- **Selenium WebDriver- Commands**
- **Selenium WebDriver- Running test on Chrome Browser**
- **Selenium WebDriver- Running test on Firefox Browser- Gecko (Marionette) Driver**
- **Selenium WebDriver- Running test on IE Browser**
- **Selenium WebDriver- Running test on Safari Browser**
- **Selenium WebDriver- Locating Strategies**
- **Selenium WebDriver- Handling drop-downs**
- **Selenium WebDriver- Drag and Drop**
- **Selenium WebDriver- Handling Alerts**
- **Selenium WebDriver- Scrolling a web page**
  - Invoke Firefox browser
  - Open URL:
  - Scroll down through the web page in order to show the Java Technology section
- **Selenium WebDriver - Browser Commands**

- **Selenium WebDriver - Navigation Commands**
- **Selenium WebDriver - WebElement Commands**
    - What is Web Element?
- **Handling Radio buttons**
- **Handling Checkbox**
- **Assertions**
    - Hard Assertion
    - Soft Assertion
- **What is Selenium Grid?**
    - Architecture of Selenium Grid
    - Why Selenium Grid?
    - How to set up Selenium Grid
- **Test NG framework in selenium with Maven project**

Aptitude syllabus :

Quantitative Aptitude

- **1 Problems Based on Ages.**
- **2 Test on Divisibility.**
- **3.Addition and Subtraction.**
- **4.Multiplication.**
- **5.Division.**
- **6.Square root and Cube root.**
- **7.BODMAS Rule.**
- **8.Indices**

- **9. Quadratic Equation.**
- **10.Word problems (Simplification with Profit Questions).**
- **11.HCF and LCM.**
- **12.Average and Percentage .**
- **13.Ratio**
- **14.Time and work.**
- **15. Pipes and Cistern**
- **16.Time and Distance.**
- **17. Boat and Stream**
- **18.Problems on Train.**
- **19. Profit and loss.**
- **20. Probability**

Reasoning Aptitude

- **1. Test on Alphabet**
- **2. Coding-Decoding**
- **3. Letter-Coding**
- **4. Number Symbol Coding**
- **5. Deciphering Message Code**
- **6. Pairs in English Alphabet**
- **7. Blood Relations**
- **8. Number Series.**
- **9. Direction and Distance.**
- **10.Ordering Test**
- **11.Sitting Arrangement**

- **12.Puzzle**
- **13.Input Output**