

## **Syllabus for Post of Assistant Programmer**

### **Programming Concepts Using C / C++/Java/Dot Net (50 Marks)**

Arrays: Declaration; initialization; 2-dimensional and 3-dimensional array, passing array to function, strings and string functions, and character arrays.

Pointers: variables, swapping data, swapping address v/s data, misuse of address operators, pointers and arrays , pointers to pointers , strings , pointer arithmetic, additional operators , portability, pointers to functions, using pointers with arrays , void pointers .

Structures and unions: syntax and use, members, structures as function arguments, structure pointers, array of structures as arguments, passing array of structure members, call by reference.

Functions; prototype, passing parameters, storage classes, identifier visibility, Recursive functions. Command-line arguments. Scope rules, Multi-file programming, Introduction to macros.

File processing in C and C++.

Introduction to graphics, graphic initialization, graphic modes, drivers, basic drawing functions, Animations- concept and implementation, Building graphical user interface.

Introduction to classes and objects; Constructor; destructor; Operator overloading; Function overloading; function overriding; friend function; copy constructor;

Inheritance,: Single , Multiple, and Multilevel Inheritance;

Virtual function and Polymorphism: Dynamic binding, Static binding; Virtual functions; Pure virtual function; concrete implementation of virtual functions; Dynamic binding call mechanism; Implementation of polymorphism; virtual destructors.

Templates: Function Templates, Class Templates, Member Function Template and Template Arguments, Exception Handling, Standard Template Library

Java Program Development , Java Source File Structure , Comparison with other languages (C & C++), Java and Internet, Features of Java, Java Virtual machine, ByteCode , Lexical Tokens, Identifiers, Keywords, Literals, Comments , Primitive Datatypes, Variables: Assignment, Initialization and Conversions, Operators: Arithmetic, Assignment, Modulus, Relational, Boolean, Bitwise., Precedence Summary ,Unicode Character Set , Arrays: Single and Multidimensional. Control Statements and Looping Structures

Class Fundamentals , Object reference , Garbage Collection, Constructors, Access Control, Modifiers, methods , Nested , Inner Class & Anonymous Classes , Abstract Class, Argument Passing Mechanism , Method Overloading, Recursion , Dealing with Static Members. Finalize() Method, Native Method. Use of “this “ reference , Cloning Objects, Generic Class Types, Inheritance in Java , Overriding Super Class Methods, Use of “super”, Polymorphism in inheritance , Type Compatibility and Conversion ,Packages & Interfaces: Defining and importing packages , Understanding Class path , Implementing interfaces.

Exceptions & Errors ,Types of Exception ,Control Flow In Exceptions , Use of try, catch, finally, throw, throws in Exception Handling ,In-built and User Defined Exceptions, Checked and UnChecked Exceptions, Operation on String ,Mutable & Immutable String , Using Collection Bases Loop for String , Tokenizing a String ,Creating Strings using StringBuffer , Multi-

Threaded Programming , Thread Life-Cycle , Thread Priorities , Synchronizing Threads , Inter - communication of Threads, DeadLock. Applet & Application , Applet Architecture, Parameters to Applet , Embedding Applets in Web page. Utility Methods for Arrays , Observable and Observer Objects , Date & Times , Using Scanner.

Streams, Input and Output Classes, The Standard Streams, File Object , File I/O Basics , Reading and Writing to Files , Buffer and Buffer Management, Read/Write Operations with File Channel , Serializing Objects , The Collection Framework , Collection Types , Sets , Sequence , Map , Hashing , Use of ArrayList & Vector , Event-Driven Programming , The Event Delegation Model , Event Classes, Event Sources , Event Listeners , Adapter Classes, Anonymous Inner classes , Keyboard and Mouse Event Handling , Avoiding Deadlocks in GUI Code , Networking Basics , Client-Server Architecture , Socket Overview, Networking Classes and Interfaces , Network Protocols , Developing Networking Applications in Java

Introduction to windows Programming. .Net Architecture, The relationship of C# to .net, The Common Language Runtime, Advantages of Managed Code. A Closer Look at Intermediate Language , Support for Object orientation and Interfaces , Distinct Values and Reference types , Strong Data Typing , Error handling and Exceptions , Use of Attributes , Assemblies , Private Assemblies , Shared Assemblies , Reflection , .net framework classes , namespaces , creating .net applications using C# , Creating windows forms , windows controls , windows services , The role of C# in the .net enterprise architecture. The .net Environment, Working with visual studio 2005, Assemblies, .net Security, Localization, Deployment.

C# Basics , variables , predefined data types : Value types and reference types , CTS types , Conditional statements , loops , jump statements , Enumerations , Arrays , Using statement , Namespace , Aliases , The Main() Method , Multiple Main Methods , Passing Arguments to main() . More on compiling C# files, console I/O, Using Comments, the C# preprocessor directives. C# Programming Guidelines. Objects and Type : Classes and Structs , Partial classes , static classes , The object class Inheritance : Types of inheritance , virtual methods , hiding methods , calling base versions of functions , sealed classes and methods , constructors of derived classes , modifiers , interfaces , derived interfaces.

Operators and Casts : Operator shortcuts , The ternary operator , The checked and unchecked operators , The is operator , The as operator , The sizeof operator , The typeof operator , Nullable types and operators , The Null coalescing operator , operator precedence , Type safety, Type conversions , Boxing and unboxing , comparing objects for equality , Operator overloading , User defined casts. Delegates and Events, Memory management and Pointers, Strings and regular expressions, Collections, Array Lists, The Stack, Queue, and Sorted List class, Hash Tables and Dictionaries,

Generics , Generic collection classes , Error and Exception Handling , Threading , Applications with multiple threads , Manipulating Threads , Creating Threads with Thread pool. Data Access with .net : Ado.net overview , Using Database Connections , Executing commands , Fast Access , The Data Reader , Managing data and relationship : The Dataset Class , XML schemas , Populating a Dataset , Persisting Dataset Changes , Working with ADO.net , windows forms , viewing .net data.

## **Software Engineering (15 Marks)**

Concept of Software engineering, Evolving role of software, Concept of software, Software Characteristics, Software Components, Software Engineering Challenges (Scale, Quality Productivity, Consistency and Repeatability, Change), Software standard, Software Engineering approach. Software Process Models: Waterfall Model, Prototyping Model, Spiral Model, Incremental Model, Concurrent Development Model.

**Software Process and Project Metrics** : Measures , Metrics and Indicators , Software measurement : Size -Oriented Metrics , Function - Oriented Metrics , Extended Function point metrics. Capability Maturity Model Integration (CMMI), Process Planning, Estimation, COCOMO Model, Risk Analysis & Management: Software risks, Risk identification, Risk monitoring and management. **Software requirements**: need for SRS, requirement process; Requirement specification (characteristics, components), Concept of Use Cases, Concept of validation

**Design Engineering**: Function oriented design, Design principles, Coupling and Cohesion, Design Notations & Specifications, Structured Design Methodology; Object-Oriented Design, OO Concepts, Design Concepts, Design Methodology, Dynamic & Functional Modeling, Design Verification.

Software Quality Concepts: Quality, Quality control, Cost of quality; Software Quality Assurance (SQA), Formal approaches to SQA, Software Reliability: Measures of Reliability, Software safety, Quality Standards. Software Testing: Testing fundamentals, Black-Box Testing, White Box Testing, Regression Testing, Smoke Testing, Alpha Testing, Beta Testing, Recovery Testing, Security Testing, Stress testing, Performance testing.

Introduction to Software Reliability: Basic Concepts, Software Reliability , Hardware Reliability, System Reliability, Software Reliability metrics, Operational Profile, Reliability Modeling, General Model Characteristics, Execution Time Component , Calendar Time Component , Calendar Time to Execution Time Relationship, Markovian Models: Poisson Type Models, Binomial Type Models, Poisson Type Models versus Binomial Type Models, Numerical examples.

Specific Models: Finite and Infinite Poisson Type Models, Musa Basic Model versus Logarithmic Poisson Model. Numerical examples. Parameter Estimation: Maximum Likelihood Estimation versus Least Squares Estimation. Comparison of SRGMs: Comparison criteria, Calendar Time Modeling and its Estimation.

## **Operating Systems (15 Marks)**

Overview Of An Operating System, Resource Management, Operating System Interface, Process Management Concepts, Inter-Process Communication, Process Scheduling, Synchronization, Deadlocks.

Memory Management, Linking, Loading, Memory Allocation, Design Issues and Problems, Virtual Memory, Fragmentation, Implementing Virtual Memory, Paging, Segmentation, Virtual Memory Design Techniques, Buffering Techniques, Spooling.

File Management - File Systems & I/O. Device Drivers, Access Strategies, File Systems, File System Organization, Design Techniques.

Multiprocessor Systems, Types Of Multiprocessor Operating Systems, Design and Implementation Issues.

Unix/ Linux Operating Systems, Users View, Design Principles, Implementation, Process Management, Memory Management, File System, I/O System.

## **Computer Networks (20 Marks)**

Goals and applications of networks. LAN, MAN & WAN architectures. Concept of WAN subnet. Overview of existing networks. OSI Reference Model Architecture, TCP/IP Model and their comparison.

Internetworking concept and architectural model. Connection-oriented and connection-less approaches. Concept of Autonomous systems and Internetwork Routing. Classful IP addresses. Subnetting, IP Multicasting. Internet Protocol (IP): connectionless delivery of datagrams (MTU, fragmentation, reassembly).

Internet control protocols: ICMP, ARP and RARP. Routing algorithms: Interior (OSPF), Exterior (BGP). Transport Layer: UDP and TCP concepts.

Socket API for Network Programming. Network Byte Ordering.

Client-Server application development using TCP & UDP sockets. Basic Server Architectures. Network Security: Firewalls and their components; Encryption techniques and examples of encryption standards.