Post Graduate Diploma in Advanced Computing August 2018

**Course Objective:** The course aims to groom the students to enable them to work on current technology scenarios as well as prepare them to keep pace with the changing face of technology and the requirements of the growing IT industry. The course curriculum has been designed keeping in view the emerging trends in advanced computing as well as contemporary and futuristic human resource requirements of the ICT industry.

**Eligibility Qualification:**

1. Graduate in Engineering or equivalent (e.g. BE / BTech / 4-year BSc / AMIE / DoEACC B Level, etc.) in Electronics/ Computer Science/ IT or related areas, OR
2. Post Graduate in Engineering Sciences (e.g. MSc in Computer Science, IT, Electronics, etc.), OR
3. Graduate in any discipline of Engineering or equivalent, OR
4. MCA, MCM, OR
5. Post Graduate in Physics/ Computational Sciences/ Mathematics or allied areas, OR
6. Post Graduate in Management with graduation degree in Science/ IT/ Computers
7. The candidates must have secured a minimum of 50% marks in their qualifying examination.

**Course Modules:**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Module Name** | **No. of hrs** |
| 1 | Operating System Concepts | 60 |
| 2 | OOPs with C++ Programming | 60 |
| 3 | Algorithm & Data structures | 60 |
| 4 | Software Application Development Tools & Techniques | 40 |
| 5 | Advanced Web Programming  | 50 |
| 6 | JavaScript Framework | 60 |
| 7 | Database Technologies | 60 |
| 8 | Java Technologies-I( Core Java) | 70 |
| 9 | Java Technologies-II( Web Based Java) | 90 |
| 10 | Microsoft .Net Technologies | 70 |
| 11 | Application Security & Testing | 40 |
| 12 | Upcoming Technology (Parallel Computing) | 20 |
| 13 | Effective Communication | 50 |
| 12 | Aptitude & General English | 50 |
| 13 | Project | 120 |
|  | **Total** | **900** |

**Detailed Contents**

**Operating Systems Concepts (60 Hours)**

Linux Commands, Vi editor, Shell Scripting

Overview of OS

Processes

Scheduling & Synchronization

Memory management

File Systems

Case Study with Linux System Programming

 Process

 Signals

 Semaphores & Mutex

 Inter – Process Communication

 POSIX Threads

**OOPs with C++ Programming (60 Hours)**

Revision of C Programming, Pointers, Functions (Call by value and reference), Recursion, Arrays using Pointers, Structures, Union, Enumeration and Typedef, File handling

**Discussion on Object oriented concepts**

Classes and Objects, Access Specifiers, Overloading, Inheritance, Polymorphism

Beginning with C++

C++ Tokens, Initialization, C++ Operators

Static Members, Constant Members

Expressions

Control Structure

Functions in C++

Constructors, Encapsulating into an object

Destructors

Associations, Inner Classes

Memory Management and pointers

Inheritance, Virtual Functions, Polymorphism

Interfaces

Exception Handling

Managing Console I/O Operations

Working with files

**Advance Topics in C++**

Object Design and Templates

STL (Standard Type Libraries)

RTTI (Run Time Type Identification)

Advanced Typecasting

new data types

new operators

class implementation

namespace scope

operator keywords

new headers

C++ Containers

**Algorithms & Data Structures using C++ (60 Hours)**

**Define the problem**

Identify the problem

Introduction to Problem Solving

Problem solving basics

Defining creativity v/s innovation

**Find Creative Solutions using creativity tools**

Effective problem solving approaches

Critical thinking and information analysis

Brainstorming, Reverse Brainstorming, Imagineering, Mind Mapping,

Six Thinking Hats: A Tool to Strengthen Critical Thinking, Collaboration, Communication, and Creativity Skills

Analyzing the situation, Gathering information, Identifying solution criteria

Decision Making Methods

Charts and Diagrams

Applying outcome-based thinking

**Evaluate and Select solution**

Pro’s and Con’s, Force field analysis, Feasibility/Capability Analysis,

Decision analysis, evaluating problems

Choosing among alternatives

Qualitative analysis, discussing qualitative analysis techniques

Establishing objectives

Assigning weight to objectives in order to make the best decision

Creating a satisfaction scale to choose between alternatives

**Implementing Decisions**

Create an action plan

Break solution into action steps

Prioritize actions and assign roles (setting priorities for taking action)

Follow-up at milestones

**Algorithm & Data Structures**

Introductory Concepts

Algorithm Constructs

OO design: Abstract Data Types (ADTs)

**Basic Data Structures**

Arrays

Stacks

Queues

Priority Queues

Deques

Linked lists

**Trees and hierarchical orders**

Introduction to trees

Abstract trees

Tree traversals

Forests

Ordered trees

Binary trees

Perfect binary trees

Complete binary trees

Search trees

Binary search trees

AVL trees

**Searching & Sorting algorithms**

Objectives of Searching

The Sequential Search

Analysis of Sequential Search

The Binary Search

Analysis of Binary Search

Introduction to sorting

Insertion sort

Bubble sort

Heap sort

Merge sort

Quick sort

**Hash functions and hash tables**

Hashing & Introduction to hash tables

Hash functions

Mapping down to 0 ... M − 1

Chained hash tables

Scatter tables

Open addressing

Linear probing

Quadratic probing

Double hashing

Poisson distribution

Collision Resolution

Analysis of Hashing

**Graph algorithms**

Introduction to graph theory

* Graph data structures
* Graph traversals
* Connectedness, Single source un-weighted path length, identifying bipartite graphs
* Minimum spanning tree algorithms, Prim's algorithm, Kruskal's algorithm,
* Single-source shortest path algorithms, Dijkstra's algorithm, A\* search algorithm, Bellman-Ford algorithm
* All-pairs shortest path, Floyd-Warshall algorithm, Johnson's algorithm
* Maximum flow algorithms, Ford-Fulkerson algorithms

**Algorithm design**

* Introduction to algorithm design techniques
* Greedy algorithms
* Divide-and-conquer algorithms
* Dynamic programming
* Backtracking algorithms
* Branch-and-bound algorithms
* Stochastic algorithms
* Analysis of different Algorithms
	+ Asymptotic analysis
	+ Algorithm analysis
* Complexity
	+ Complexity Analysis
* Application of Data structures

**Software Application Development Tools & Techniques (40 Hours)**

Object Oriented Analysis and Design

UML

Software Engineering

Brief concept of Software Life Cycle Models

Agile Techniques for software development

* + Agile Principle and Mindset
	+ Value-Driven Delivery
	+ Stakeholder  Engagement
	+ Team Performance
	+ Adaptive Planning
	+ Problem Detection And Resolution
	+ Continuous Improvement
	+ Brief of Tools and technologies

Software Development Tools & Techniques

Software Quality Assurance

Introduction to Coding Standards

Software Testing

Different Testing Tools

Test Driven Development (TDD)

Project Management

Risk Analysis and Management

Case Study

**Advanced Web Programming (50 Hours)**

**Web Programming concepts (10 Hours)**

Architecture of the Web

**HTTP Protocols**

* Difference HTTP1.0 , HTTP 1.1 and HTTP2.0
* TCP/IP
* Stateless nature of the protocol
* Methods (GET, POST, HEAD, PUT, DELETE
* HTTP session
* Status codes
* Persistent connections
* HTTPS

**HTML**

* Document Object Model (DOM)
* Elements
* Events

**HTML 5**

* Elements
* Objects
* Events
* Canvas
* Audio & Video Support
* Geo-location Support

**CSS**

* Styling HTML with CSS
* Inline Styling (Inline CSS)
* External Styling (External CSS)
* CSS Fonts
* The CSS Box Model
* The id Attribute
* The class Attribute
* HTML Style Tags

**PHP (16 Hours)**

Introduction to PHP

Working with arrays

Functions

Forms

Handling date and Times

Working with Files

Session and state management

Database operations from PHP

**XML & Web Security (12 Hours)**

**XML**

* Introduction to XML
* XML Validation
* Reason for XML
* XML Tree Structure
* XML DOM
* XML DTD
* XML Schema
* XML style language
* XML and XSLT
* XML Parsing
* XML parsers (DOM & SAX)
* XML WSDL
* RSS Feed

**Web Security**

* SQL Injection
* Cross-Site Scripting (XSS)
* Security standards (OWASP)

**AJAX (2 Hours)**

Introduction to Ajax

Web services and Ajax

Ajax using HTML, CSS, JavaScript

Ajax Framework and DOM

XMLHttpRequest

Ajax Architecture

**Working with JSON (6 Hours)**

JSON – Introduction

Need of JSON

JSON Syntax Rules

JSON Data - a Name and a Value,

JSON Objects,

JSON Arrays,

JSON Uses JavaScript Syntax,

JSON Files

JSON & Security Concerns, Cross Site Request Forgery (CSRF), Injection Attacks,

JS XMLHttpRequest functions

JavaScript XMLHttpRequest & Web APIs

JSON & Client Side Frameworks

JSON & Server Side Frameworks

Replacing XML with JSON

JSON parsing

AJAX using JSON and jQuery

**Responsive Web Design (4 Hours)**

Introduction

The Best Experience for All Users

* Desktop
* Tablet
* Mobile

**Bootstrap**

Overview of Bootstrap

Need to use Bootstrap

Bootstrap Grid System, Grid Classes, Basic Structure of a Bootstrap Grid

Typography

Tables, Images, Jumbotron, Wells, Alerts, Buttons, Button Groups, Badges/Labels, Progress Bars, Pagination, List Groups, Panels, Dropdowns, Collapse, Tabs/Pills, Navbar, Forms, Inputs

Bootstrap Grids, Grid System, Stacked/Horizontal

Bootstrap Themes, Templates

**JavaScript Framework (60 Hours)**

**JavaScript**

Introduction to JavaScript

Variable, statements, Operators, Comments, constructs, Functions, expressions

Javascript console

Scope, Events, Strings, String Methods, Numbers, Number Methods, Dates, Date Formats, Date Methods

Arrays, Array Methods

Booleans, Comparisons

Control Structures

Conditions, Switch,

Loop For, Loop While, Break

Functions, Function Definitions, Function Parameters, Function Invocation, Function Closures

Objects, Object Definitions, Object Properties, Object Methods, Object Prototypes

Object Oriented Programming

* Method
* Constructor
* Inheritance
* Encapsulation
* Abstraction
* Polymorphism

Javascript Validations

Document Object Model, Document and Events (DOM Manipulation)

HTML DOM, DOM Intro, DOM Methods, DOM Document, DOM Elements, DOM HTML, DOM CSS, DOM Animations, DOM Events, DOM EventListener, DOM Navigation, DOM Nodes, DOM Nodelist

Debugging

Type Conversion, Regular expressions,

Errors, Debugging

Forms, Forms Validation, Forms API

JS Browser BOM, Window, Screen, Location, History, Navigator, Popup Alert, Timing, Cookies

Javascript Windows

Pushing code quality via JSLint tool

Security in Java Script

**Jquery**

* Basics of jQuery
* jquery selection and events
* jQuery Effects
* jquery traversal and manipulation
* Data attributes and templates
* jQuery Plugins
* Jquery / Google Web Toolkit

**Node.js**

* Introduction to Node.js
* Node modules
* Developing node.js web application
* Event-driven I/O server-side JavaScript

**Express**

* Introduction to Express
* First Express Application
* Application, Request and Response Objects
* Implementing MVC Pattern
* Express application configuration
* Rendering Views

**AngularJS**

* Introduction to AngularJS
* Structuring AngularJS application
* MVC in AngularJS
* AngularJS routing
* AngularJS services

**Testing Web Applications**

* Introduction to JavaScript Testing
* Testing Express Applications
* Testing AngularJS Applications

**Database Technologies (60 Hours)**

**Database Concepts**

Client/Server Computing

RDBMS Technologies

Codd’s Rules

Data Models

Normalization Techniques

ER Diagrams

**SQL**

Overview of OORD

Introduction SQL\*Plus

DDL, DML and DCL

Tables, Indexes and Views

Clusters, Sequences and Snapshots

Cursors

Stored Procedures, Triggers, Packages

Introduction to No SQL

MongoDB (Virtual DB)

**Java Technologies-I( Core Java) (70 Hours)**

Data Types, Operators and Language

Constructs

Inner Classes and Inheritance

Interface and Package

Exceptions

Collections

Threads

Java.lang

Java.util

Java.awt

Java.io

Java Virtual Machine

Java Persistent

Servlets

**Java Technologies-II( Web Based Java)(90 Hours)**

Java Server Pages

JDBC

JavaBeans

Java Security

Naming Services

Java Annotations

Java Mail

Java Messaging Services

Transactions

Apache maven

Introduction to Struts Framework (2hrs)

Introduction to hibernate

HQL, Hibernate

Spring Framework

Hands on Web services – JSON/XML/oData (data format conversation)

**Microsoft .Net Technologies (70 Hours)**

**MS .NET 4.5 Framework (10 Hours)**

Introduction to NET 4.5 Frameworks

Application Domain

Language Interoperability

.NET Framework Class Library

Assemblies

Introduction of Windows Presentation Foundation

Introduction of Windows Communication Foundation

**C# .NET 4.5 (24 Hours)**

Need of C#

Operators

Namespaces & Assemblies

Arrays

Preprocessors

Delegates and Events

Boxing and Unboxing

Regular Expression

Collections

Exceptions Handling

Introduction to win forms

ASP .NET 4.5 (36 Hours)

Building .NET components

ADO.NET 4.5

Querying with LINQ

Custom Control

Master Pages, Themes and skins

Introduction to Web Services

MS.NET MVC Framework

Enterprise Services

Personalization and Localization

Deployment

**Application Security & Testing(40 Hours)**

Web Application Security challenges

Advanced Web application attacks and remedy

Secure Software Development Life Cycle (SDLC)

Web server security

Security standards (OWASP)

Ways to strengthen web application security

Database testing: SQL Injection

Security testing

Performance testing

**Upcoming Technology (Parallel Computing)(20 Hours)**

Architecture for software

What is a thread, Strategies of Multi-Threading

Task/Data Parallelism

OpenMP, OpenMP Programming Model, OpenMP constructs

|  |
| --- |
|  |
|  |

**Effective Communication(50 Hours)**

**Aptitude & General English(50 Hours)**

**Project (120 Hours)**