Scheme for One Year Postgraduate Diploma in Computer Applications(PGDCA)

Semester – I

		Theory	Lab.	Marks
1.1	Fundamentals of Programming	3		100
1.2	Database Management Systems	3		100
1.3	Software Engineering	3		100
1.4	Operating System	3		100
1.5	Multimedia and Linux Lab		4+4	100
1.6	Software Lab-based on 1.1, 1.2		6+6	100
		12 T+ 20L= 32		500

Semester – II

		Theory	Lab.	Marks
2.1	Object Oriented Programming	3		100
2.2	Front End Programming	3		100
2.3	Web Technology	3		100
2.4	Network Basics and Administration	3*		100
2.5	Software Lab based on 2.1, 2.2, 2.3		6+6+6	200
2.6	Project			100
		12 T+ 18L= 30		600

* 5 classes in a semester shall be dedicated for live case-study.

Note:

1. Programming Language shall be specified at the beginning of each year. To begin with, C++ in semester I (Paper 1.1), Java and C# in Semester II (Papers 2.1 and 2.2) shall be taught.
2. References shall be updated at the beginning of every semester.
3. 25 marks in each theory paper shall be for the internal assessment and the remaining 75 marks for the end semester theory examination. The duration of examination shall be three hours.

4. In each practical/project paper, 50% marks will be for internal assessment and the remaining 50% for the end semester examination and in order to pass the practical/project paper, the student must secure at least 40% marks in each component.

5. The medium of Instruction and Examination shall be English.

6. There shall be an examination at the end of each semester. Semester I and II examinations will be held at the end of semester I and II respectively. Examination for courses shall be conducted only in the respective odd and even semesters as per the Scheme of Examinations. Regular as well as Ex-Students shall be permitted to appear/re-appear/improve in courses of odd semesters only at the end of odd semesters and courses of even semesters only at the end of even semesters.

7. The student should submit the project synopsis to the internal supervisor who will be a faculty member of the college. Internal Assessment marks would be based on monthly progress report submitted by the students to his/her internal supervisor followed by an oral presentation. The dissertation shall be evaluated at the end of the semester in a centralized manner as it is being done currently.

8. **Passing Criteria**: In order to be eligible for the award of Diploma, a student must successfully complete Semester I and Semester II examination separately. A student will be deemed to have successfully completed the course in a semester, if he/ she secures at least 40% marks in

    (i) each of the practical/ project paper prescribed in a semester.

    (ii) The aggregate of theory papers prescribed in a semester i.e. 40% marks in the aggregate of internal assessment and 40% marks in the aggregate of end semester examination separately.

9. **Division Criteria**: The final result will be declared on the basis of the combined marks obtained in the courses prescribed for the two semesters.

    The successful candidates will be classified as follows:

    a) First Division: 60% or more marks.

    b) Second Division: greater than or equal to 50% but less than 60% marks.

    c) Pass: greater than or equal to 40% but least than 50% marks.

10. **Eligibility Conditions**: Any graduate who has studied Mathematics at +2 stage of Senior Secondary Certificate Examination and at least one Paper of Mathematics during graduation and has secured a minimum of 50% marks in aggregate from the University of Delhi or any other recognized University.

11. **Admission Procedure**: Admission to the course will be based on Entrance Test comprising of questions based on Mathematical Ability, Deductive & Logical Ability and English Comprehension. Entrance Test shall be conducted by the Department of Computer Science with the help of colleges running PGDCA course in rotation in alphabetical order.

12. **Span Period**: A student must complete the course within two years of the initial enrollment.

13. The intake to the course will be limited to 20 students + OBC quota (reservation of seats as per Delhi University Rules).

14. Course fee Rs. 10,000/- for the complete course in addition to College fee.

Syllabii

1.1 Fundamentals of Programming

Fundamentals of Programming: Flowcharts and pseudocodes,Development environment, compiling and running programs using a text editor. Programming language features: comments, data types, variables, assignments and initialization, operators, strings, arithmetic expressions and built-in operators, console based I/O and related built-in I/O functions, controlling program flow- selection and iteration,Working with arrays and strings.
Functions: Concept of a subprogram, function definition and declaration, using library functions.

User Defined Types: Structures, Preprocessor Directives, Macros, Handling File I/O

REFERENCES

1. Schaums Series "Programming in C++",
2. N. Dale, C. Weems, M. Headington, "Programming in C++", Narosa Publishing House, 1999

3. J. P. Cohoon, J. W. Davidson, "C++ Program Design", Tata McGraw-Hill Publishing Company, 1999

1.2 Database Management Systems

Introduction: Overview of Database Management System and Database Systems Architecture

Data Modeling using ER Models: Entity type, entity set, relationship Type, relationship set, keys, constraints, ER diagrams, higher degree relationship representations.

Relational Model: Relational model concepts, relational model constraints and relational database schema, conversion of ER diagrams to relational database design, Functional Dependencies and Normalization (up to 3NF).

SQL: SQL Data Definition and Data Types, Data Retrieval, Data Manipulation, Built in functions, Views, Introduction to Oracle Dictionary, DBA statements and Transaction Control statements.

PL/SQL Programming: PL/SQL Overview, Conditional Logic, Loops, Exception Handling, Triggers, Procedure and Functions, Packages.

REFERENCES

1. R.A. Mata-Toledo, P.K. Cushman, "Database Management Systems", Tata McGraw Hill, 2007
2. I. Bayross, "SQL, PL/SQL – The Programming Language of Oracle", BPB Publications 2003
3. J.A Hoffer, M.B Prescolt, F.R. McFadden, "Modern Database Management", Pearson Education.

1.3 Software Engineering

Introduction: The Software Engineering Approach

Software Processes: Software Development Process Models: Waterfall Model, Prototyping, Iterative Development, Timeboxing Model, Comparison of Models, Software Configuration Management Process

Software Requirements Analysis & Specification: Software Requirements: Need for SRS, Requirement Process, Problem Analysis: Data Flow Modeling, Prototyping, Requirements Specification: Characteristics of SRS, Components of SRS Validation, Metrics: Size, Quality metrics

Planning a Software Project: Process Planning, Effort Estimation, Project Scheduling & Staffing, Software Configuration Management Plan, Quality Plan, Risk Management Concepts, Project Monitoring Plan

Function – Oriented Design: Design Principles: Problem Partitioning, Abstraction, Modularity, Module – Level Concepts: Coupling, Cohesion, Design Notation & Specification

Software Testing: Black Box Testing: Equivalence Class Partitioning, Boundary valve analysis, Course Effect Graphing, White Box Testing: control Flow-based criteria, Data flow-based Testing, Testing Process: Levels of Testing

REFERENCES

Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Publishing House. .
Roger Pressman, Software engineering : a practitioner's approach (sixth edition) McGraw-Hill Science

1.4 Operating System

Introduction to Operating System: Operating system as user interface, Types of operating system, Operating System structure (system components, system calls, system program) Basics of Process management, File Management and I/O Management.

Linux: Overview of linux system − running commands, pipes and filters, shell programming, working with bash shell, I/O streams and piping, user administration: user, groups and permissions
linux file system, file processing, working with Vi editor

REFERENCES

1. M G Sobell , "A Practical Guide to Linux Commands, Editors, and Shell Programming", Prentice-Hall, 2nd Edition
2. B W Kernighan , R Pike, "Unix Programming Environment", 5th edition, Prentice-Hall, 2004
3. Ida M. Flynn and Ann M. Mchoe, "Understanding Operating System" (3rd ed.), Publisher: Thomson Learning.

1.5 Multimedia and Linux Lab

Use and development of multimedia: Combining text, graphics, sound, still and video images and animation into a unified, seamless document, capturing the various media, creating, editing and storing the various media.
Digital Imaging using Photoshop : Performing basic photo correction, retouching and repairing of images, understanding the basics of layers, create and use masks and channels, use basic pen tool techniques, create painting and editing of digital artwork, create special effects using filters, create images for use on the web, managing color in an image.
Macromedia flash for creating interactive movies and animations, use of flash to make the website more interactive and dynamic: creating animations, creating special effects, sound embedding, video embedding, preparing and publishing movies, importing and modifying graphics, building complex animations, using action script

Linux Lab: Practical based on Paper 1.4, Using Linux Commands.

REFERENCES

1. Judith Jeffcoate, "Multimedia in Practice: Technology and Applications", Pearson Education, 1st Ed.
2. "Adobe Photoshop 7 – classroom in a book", Adobe Press, 2002
3. J Armstrong, J deHaan, "Macromedia Flash, A tutorial Guide", Macromedia Press, 2005
4. M Pizzi, "Macromedia Flash MX Unleashed", Sams 2002
5. D Tyler, "How to use Macromedia Flash MX and Action script", 2nd Ed., Que, 2002

2.1 Object Oriented Programming

Basics of Object oriented Programming: Concept of object oriented programming, object-oriented languages, applications of OOP.
Introduction to an Object Oriented Programming language: Development environment, fundamental programming structures in language used, comments, data types, variables, assignments and initialization, operators, strings, control flow, arrays
Implementing Object oriented concepts: Classes, constructors, destructors
Inheritance: inner classes, access specifiers.
Polymorphism: Overloading, Overriding, Abstract classes and functions.
Interfaces and Packages, Exception Handling
Designing Text/Graphics/Web Applications, Introduction to Applets
Introduction to AWT (Abstract Window Toolkit)

REFERENCES

1. T Budd,"An Introduction to Object-Oriented Programming", Addison Wesley; 3 editions, 2001.
2. H Schildt, "The Complete Reference -Java2", Tata McGraw Hill, 5th Edition, 2002
3. H Deitel and P Deitel, "Java How to Program", Prentice Hall, 7th edition, 2007
4. Y.D Liang., "Introduction to Java Programming", Prentice Hall

2.2 Front End Programming

Introduction to .NET: An overview of application architecture, identifying pre .NET technologies, defining .NET, identifying features of .NET
Understanding CLR
Getting Started with C#.NET: Introduction to Visual Studio.NET
Language Fundamentals of C#: Building blocks of C#.NET programming language , working with windows form, working with data types and variables, using operators, working with arrays ,controlling flow in programs, recursion.

Object-Oriented Programming in C#.NET: Creating and using classes, objects and structures in C#.NET, constructor, destructor, garbage collection, inheritance procedures, procedure overloading, inheritance, restricting access to data member, overriding methods, implementing interfaces, event handling in C#.NET.
String handling: message class, common dialog classes, event procedure and delegate classes using exception handling, file I/O operations in C#.NET.
Event handling in C#.NET
GUI Programming with Visual C#
Database Programming with ADO.NET:
Designing a simple application using ADO.NET, data binding, executing SQL commands, error provider class, data updation and conflict management
Introduction to web services
Application deployment in C#.NET

REFERENCES

1. K Watson, "Beginning Microsoft Visual C# 2008", Wrox 2008
2. Deitel & Deitel, "Visual C# 2008 How to Program", Pearson Education, 2008

3. A Hejlsberg, M Torgersen, S Wiltamuth, P Golde, "The C# Programming Language", Addison:Wesley Professional, 3rd Ed. 2008

2.3 Web Technology

Introduction to Web Development: Fundamentals of HTML, DHTML, JavaScript and active X Control.

Web Form Designing using ASP.NET: Visual web developer, creating a new website, Introduction to Visual Studio.NET Designing a simple web form, including C# code in ASP.NET, hosting the web page, introducing IIS Web Server, selecting service provider, hosting on IIS web server.

Structure of an ASP.NET Page, Using HTML server controls, using ASP.NET built-in objects, Themes and Skins, Working with different types of web application in ASP.NET Event handling in a web application, Data binding, Navigating between forms.

REFERENCES

1. S Walther, "ASP.NET Unleashed"
2. D G Andrew, "MICROSOFT® ASP.NET PROGRAMMING WITH MICROSOFT VISUAL C#® .NET STEP BY STEP", (Microsoft Press), 2003
3. P Sheriff and K Getz, "ASP.NET Developer's Jump Start", Addison-Wesley Professional, 2002.

2.4 Network Basics and Administration

Networks Overview; Network Fundamentals and Topologies, Client server architecture network protocols, OSI reference model

Issues of TCP/IP Networking; Networking interfaces, IP addresses, address resolution, IP routing, The Internet Control Message Protocol

Configuration of Network Hardware; PLIP and SLIP drivers, PPP, configuring TCP/IP networking, activating and de-activating NIC , subnets , enable forwarding , TCP/IP commands, network security, installation and maintenance, (practically preparing a LAN) TCP/IP Firewall, Electronic Mail, Sendmail, Proxy servers, Network server configuration, Backups and antivirus policies, Implementation of DNS, Troubleshooting network problems

Case Study: DU Network

REFERENCES

1. O Kirch & T Dawson, "Linux Network Administrator's Guide", O'REILLY, 2nd Edition, 2000
2. D.E. Comer, "Internetworking with TCP-IP: Principles, Protocols and Architecture", Prentice Hall, 5th Edition, 2002