MAHARAJA SURAJMAL BRIJ UNIVERSITY
BHARATPUR

SYLLABUS OF PGDCA (I & II Semester)

Academic Session 2025-26

डॉ. अरुण कुमार पाण्डेय
उपकुलसचिव
प्रभारी अकादमिक प्रथम

MAHARAJA SURAJMAL BRIJ UNIVERSITY, BHARATPUR

PGDCA

Programme Objectives

1. The primary objective of this program is to prepare students for careers in software industry, understanding and skills, related to the use of computers and its applications.
2. The course is designed to function as an intermediate between the industry and academic institutes.
3. This course provides students with options to specialize in new and upcoming technologies.
4. To impart creativity and pursuit of excellence in computer applications.
5. To provide opportunity for the study of modern methods of information processing and its applications.
6. To develop among students the programming techniques and the problem solving skills through programming.
7. To develop the ability to use this knowledge to analyze new situations.
8. To be able to blend the acquired knowledge, understanding, and experience, for a better and improved intellectual capacity of the real-life problems.
9. To prepare students who wish to go on to further studies in computer science and related subjects.

Programme Outcomes

1. **Fundamental Knowledge**: To provide a strong foundation in computer science and its applications, covering areas such as programming, software development, and database management.
2. **Practical Skills**: To develop practical skills in the use of software tools and technologies, enabling students to handle real-world computing tasks effectively.
3. **Technical Proficiency**: To enhance technical proficiency in various computer applications, including web development, networking, and cyber security.

डॉ. अरूण कुमार पाण्डेय
उपकुलसचिव
प्रभारी अकादमिक प्रथम

4. **Problem-Solving**: To cultivate problem-solving abilities and analytical thinking through the application of computational methods to solve complex problems.

5. **Software Development**: To train students in software development methodologies, from design and development to testing and maintenance.

6. **Project Work**: To provide hands-on experience through project work, enabling students to apply theoretical knowledge in practical scenarios.

7. **Industry Readiness**: To prepare students for careers in the IT industry by imparting knowledge of the latest technologies and industry practices.

8. **Professional Growth**: To promote continuous professional growth and lifelong learning in the rapidly evolving field of computer science.

9. **Ethical and Social Awareness**: To feel a sense of ethical responsibility and awareness of social impacts associated with the use of computer technology.

10. **Further Education**: To lay the groundwork for further studies in computer science or related fields, such as a Master's degree or specialized certifications.

Eligibility :

All the graduate(with 10+_2+3) from recognized university situated in Rajasthan having 48% marks/ CGPA of 3.0 in the UGC Seven Scale for general category & for SC/ST/Non-Creamy Layer OBC/MBC candidates having only pass marks and minimum 60% marks for non-Rajasthan candidate. Reservation as per the University Rules.

Scheme of Examination of PGDCA for the Academic Session 2025-26

Each of the semesters I and II will consist of six theory papers and three practical papers (Laboratories).

1. Each theory paper shall carry 100 marks for the University semester examination of three hours duration.

2. The University Examination of the theory paper will consist of six questions on the pattern mentioned below :-

   a. Candidate has to attempt six questions in all.
   b. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
   c. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

डॉ. अरुण कुमार पाण्डेय
उपकुलसचिव
प्रभारी अकादमिक प्रथम

3. Each practical paper shall be of 3 hours duration on one day and carry 100 marks for the practical examination. The practical examination will involve 3 exercises, each of 20 marks, practical record of 15 marks and viva-voce examination of 25 marks.

4. The medium of instruction and examination shall be English only.

5. (a) The minimum marks for passing each theory and practical examination shall be 40% separately in the University semester end examination.
(b) The candidate may be promoted to the II semester if he/she has cleared at least four theory papers and two practical papers of the Semester-I.

6. At the end of the final examination, the candidate eligible for the award of PGDCA degree shall be classified on the basis of marks obtained in semesters I and II examination taken together as follows:

(a) I division with Honour - 75% or more marks in aggregate provided the candidate has passed all papers and examinations in first attempt.

(b) I Division - 60% or more marks but fails to satisfy the criterion for being classified distinction as lay in the 7(a).

(c) II Division - All other than those included in 7(a) and 7(b) above, and marks 48% or more but less than 60% of the aggregate marks.

(d) All the rest will be declared to have passed the examination, if they obtain a minimum pass marks in each paper, ver., 40%.

7. A candidate must pass the PGDCA Course within Three years of the initial admission to the course.

8. For the award of prizes or ranking, the marks obtained in the first attempt of the examination only will be taken into account.

डॉ. अरूण कुमार पाण्डेय
उपकुलसचिव
प्रभारी अकादमिक प्रथम

PGDCA-First Semester 2025-26

S. No	Subject Code	Subject Title	Course category	Credit	Contact Hours per Week			EoSE Duration(Hr)	
					L	T	P	Thy	P
1	PGD- 50-T-101	Computer Fundamentals	CC	4	3	1	0	3	0
2	PGD 50-T-102	Operating System	CC	4	3	1	0	3	0
3	PGD 50-T-103	Programming in C	CC	4	3	1	0	3	0
4	PGD 50-T-104	Office Management Tools	CC	4	3	1	0	3	0
5	PGD 50-T-105	E-Commerce	CC	4	3	1	0	3	0
6	PGD- 50T-106	Database Management Systems	CC	4	3	1	0	3	0
		Practical							
1	PGD 50-P-107	Programming in C	CC	4	0	0	6	0	3
2	PGD 50-P-108	Office Management Lab	CC	4	0	0	6	0	3
3	PGD 50-P-109	DBMS Lab	CC	4	0	0	6	0	3

PGDCA-Second Semester 2025-26

S. No.	Subject Code	Subject Title	Course category	Credit	Contact Hours per Week			EoSE Duration(Hr	
					L	T	P	Thy	P
1	PGD 50-T-201	Java Programming	CC	4	3	1	0	3	0
2	PGD 50-T-202	Web Application Development	CC	4	3	1	0	3	0
3	PGD 50-T-203	Software Engineering	CC	4	3	1	0	3	0
4	PGD 50-T-204	Data Communication & Computer Networks	CC	4	3	1	0	3	0
5	PGD 50-T-205	Elective –I	CE	4	3	1	0	3	0
6	PGD 50-T-206	Elective –II	CE	4	3	1	0	3	0
		Practical							
1	PGD 50-P-207	Java Lab	CC	4	0	0	6	0	3
2	PGD 50-P-208	Web Development Lab	CC	4	0	0	6	0	3
3	PGD 50-P-209	Elective-III Lab	CE	4	0	0	6	0	3

Elective -I (any One)

1. Data Structures and Algorithms
2. Cloud Computing
3. Computer Architecture

Elective -II (Any One)

1. Python Programming
2. PHP Programming
3. Digital Marketing

Elective - III Lab (ANY One)

1. Python Lab
2. PHP Lab
3. Digital Marketing Lab

Course Content for PGDCA First Semester 2025-26

COURSECODE- 50-T-101

Computer Fundamentals

Theory: 3 hours per week

Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Understand Fundamental Computing Concepts and gain a solid foundation in the basic principles of computer science, including hardware, software, and networking.
- Develop Proficiency in Office Management Tools such as word processors, spreadsheets, presentation tools, and database management systems.
- Apply Information Technology in Business Contexts to Understand how information technology can be applied to solve business problems and improve organizational efficiency.
-

Course Outcomes :

- **Understanding Basic Concepts**: Students will gain an understanding of the fundamental concepts of computers, including the definition and functions of hardware and software.
- **Knowledge of Computer Architecture**: Students will learn about the basic architecture of computers, including the central processing unit (CPU), memory, input/output devices, and storage devices.
- **Understanding Data Representation**: Students will understand how data is represented and processed within a computer, including binary numbers, ASCII, and Unicode.
- **Introduction to Programming**: Students will be introduced to the basic knowledge of programming and their generations, often introduction of old languages to new concepts.
- **Awareness of Emerging Technologies**: Students will become aware of emerging technologies and trends in the field of computing, such as internet and cybersecurity.
- **Ethical and Social Implications**: Students will understand the ethical and social implications of computing, including issues related to privacy, security, and digital citizenship.
- **Effective Use of the Internet**: Students will learn how to effectively use the Internet for research, communication, and collaboration.

- **Digital Literacy**: Students will develop digital literacy skills, enabling them to effectively and responsibly use digital technologies in various contexts.

Unit- I

Building block of computer system : Basic building blocks- I/O, Memory, ALU and its components, Control unit and its functions, Instruction-word, Instruction and execution cycle, branch, skip, jump and shift instruction, Operation of control registers, Controlling of arithmetic operations, Classification of computers(Workstation, Mainframe, Super computer, Client-Server computer, Notebook, Tablet, Palmtop computer).

Unit II

Computer and its generations, Programming languages generations- Machine, Assembly, High leve and OOPs. Language translators. Overview of the Digital Computes –Digital, Analog, Hybrid computers, Digital versus Analog computers, Computer Software – System software and Application Software. Types computer Codes (BCD, ASCII, EBCDIC, Unicode).

Unit-III

Representation of Data:, Digital number system (binary, octal, decimal and hexadecimal numbers,), Conversion from one form to another, fractional numbers and signed numbers, Complements, Arithmetic operations on binary numbers, Fixed point and floating point representations., Logic Gates (NOT, OR, AND), Encoding and decoding.

Unit-IV

Computer Components (Briefly overview) : Mother Board, Processor, types of RAM, RAM, Flash, Cache,; SDRAM, DDR), System clock, Buses (Data, Address, Control).

Input devices & output Devices –Printers, Scanner, different types of scanner

Storage devices : Storage types , random versus sequential access, formatting, tracks and sectors, speed, storage capacity, Hard Disk structure; Hard Drive Interfaces (IDE, EIDE, SCSI, RAID,SATA,ATA). Optical Disks : pits and lands, CD (ROM,R,R/W), DVD (ROM,R,RAM), Magnetic tapes, Modem (Fax/Data/Voice).

Unit- V

Internet Applications: Internet, Internet Applications, e-Mail, IRC, Web Surfing, Web Browsers,, Search Engines, Internet Service Providers, Downloading, Audio and Video Conferencing.

Security issues in Internet — Bugs, Viruses, Anti-viruses, Firewalls etc. Internet threats to the society, Cyber laws and Legal issues.

Suggested Reference Books:

1. M. Morris Mano: Computer System Architecture, 3 Hall of lndia.
2. John D. Carpinell: Computer Systems Organization & Architecture, 3 edition;Pearson Education Asia.
3. Peter Norton's Introduction to Computers, Third Edition, McGraw Hill
4. Sinha PK; Computer Fundamentals;BPB.
5. Malvino B.; Digital Computer Electronics; III Edn;TM}I.
6. Albert Paul Malvino, Electronic Principles, McGraw Hill

COURSE CODE - PGD - 50 - T - 102

Operating System

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

Upon finishing the course, students will be able to:

- To design and understand the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems.
- To evaluate, and compare OS components through instrumentation for performance analysis.
- To analyze the various device and resource management techniques for timesharing and distributed systems
- To develop and analyze simple concurrent programs using transactional memory and message passing, and to understand the trade-offs and implementation decisions

Course Outcomes

- **Understanding Operating System Concepts**: Students will understand the basic concepts and functions of operating systems, including their role in managing hardware and software resources.
- **Knowledge of System Architecture**: Students will learn about different types of operating system architectures, including monolithic, microkernel, and modular architectures.
- **Process Management**: Students will understand the concepts of process management, including process scheduling, creation, termination, and synchronization.
- **Thread Management**: Students will gain knowledge about thread management, including the difference between processes and threads, and the concepts of multithreading and concurrency.
- **Memory Management**: Students will understand memory management techniques, including paging, segmentation, and virtual memory.

- **File Systems**: Students will learn about file systems and their management, including file organization, storage, access methods, and directory structures.

Unit-I

Needs of an Operating System, Operating system structure, Evolution of Operating System (multiprogramming systems, batch systems, timesharing system, distributed systems and Real Time system), Operating system structure, Operating system components and services, system calls, system programs, Virtual machines.

Unit-II

Process management: process concept, process scheduling, cooperating processes, Threads, Inter-process communication, CPU scheduling criteria, Scheduling algorithms, Multiple-processor scheduling, Real time scheduling and Algorithm evaluation.

Unit-III

Process Synchronization and Deadlocks: The Critical section problem, synchronization hardware semaphores, Classical problems of synchronization, Critical regions, Monitors, Deadlocks-System model, Characterization, Deadlock prevention, Avoidance and Detection, Recovery from deadlock, Combined approach to deadlock handling.

UNIT –IV

Storage management: Memory management- Logical and Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation with paging, Virtual Memory, Demand paging and its performance, page replacement algorithms, Allocation of frames, Threshing, Page Size and other considerations, Demand segmentation

Unit-V

File & Disk Management :File systems, secondary storage Structure, File concept access methods, directory implementation, Efficiency and performance recovery, Disk structure, Disk scheduling methods, Disk management , Recovery Disk structure, disk structure, disk scheduling methods, disk management, Swap-Space management, Disk reliability.

Recommended books:

1. Galvin P.B, Silberschatz; Operating System Principles; (Seventh Edition),J Wiley.
2. Tanenbaum A.S, Modern Operating Systems, 2nd Edn. PHI Publ.
3. William Stalling: Operating Systems, Internal & Design Principles, Sixth Edn; Pearson.
4. Gary Nutt: Operating Systems-A Modern Perspective (Second Edition) , Pearson Education.
5. D.M. Dhamdhere: Systems Programming and Operating Systems (Second Edition), Tata McGraw Hill Publishing company Limited.
6. Harvey M. Deitel, Operating Systems, Pearson Education.

COURSE CODE - PGD-50-T-103

Programming in C

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

Upon successful completion of this course, students will be able to learn:

- To Understand the basic language implementation techniques
- Develop ability to learn new languages more quickly
- To understand the concept of functional programming language
- Develop ability to learn and write small programs in different programming Languages

Course Outcomes

- Understanding C Language Fundamentals: Students will understand the fundamental concepts of the C programming language, including syntax, semantics, and structure.
- Proficiency in Writing C Programs: Students will be able to write, compile, and execute C programs, demonstrating proficiency in using the language's constructs.
- Mastery of Data Types and Variables: Students will gain knowledge of various data types, variables, and constants in C, and understand how to use them effectively in programming.
- Control Structures: Students will understand and use control structures such as loops (for, while, do-while), conditional statements (if, else, switch), and branching statements (break, continue, return).
- Functions and Modular Programming: Students will learn to define and use functions for modular programming, including understanding the concepts of function declaration, definition, and scope.
- Arrays and Strings: Students will understand the use of arrays and strings in C, including single-dimensional and multi-dimensional arrays, and string handling functions.
- Pointers and Memory Management: Students will gain a solid understanding of pointers, pointer arithmetic, dynamic memory allocation, and memory management techniques.

- Structures and Unions: Students will learn to define and use structures and unions to group different data types and manage complex data.
- File Handling: Students will understand file handling concepts in C, including reading from and writing to files, and working with different file modes.
- Preprocessors and Macros: Students will learn about the use of preprocessors, macros, and header files to enhance code efficiency and readability.
- Understanding the concepts of Stacks, queues, linked lists, binary search trees concepts to implement effectively.

Unit-I

Problem solving with computers, Flow charts, Basic concepts of programming languages, programming domains.

C Character set, variables and constants, keywords, Type checking, Scope and lifetime data types. Operators, Instructions, assignment statements, arithmetic expression, comment statements, simple input and output, Boolean expressions.

Unit-II

Control structures, decision control structure, loop control structure, case control structure. String and character handling, arrays and string processing, data validation examples .

Functions, function prototype, subroutines, scope and lifetime of identifiers parameter passing mechanism, recursion.

Unit-III

User defined data types, enumerated data types, unions, structures, array of structures,

Unions of structures. Storage class specifies, Pre processors header files and standard lib, Functions.

Unit IV

Pointer : Definition and uses of pointers, arithmetic , pointers and arrays, pointers and function, pointer to pointer, pointer to structures. Dynamic memory allocation.

Unit-V

Implementation of simple data structures : Stacks, Queues, Linked Lists, Binary search tree, searching and sorting algorithms.

Console Input and Output functions, data files, operations on data files, text and binary files, formatted data files.

Recommended reference books:

1. Gottfried B; Programming with C: Schaum Qutlines; Mc Graw Hill Edition.
2. Balagurusamy E; Programming in ANSI C;Fifth Edn; Mc Graw Hill,2011.
3. Kanetkar Y.; LET US C; X Edition, BPB,2010
4. Deitel HM & Deitel JP; C How to program; 5th Edn; Pearson Pub.

COURSE CODE - PGD-50-T-104

Office Management Tools

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Understand Fundamental Computing Concepts and gain a solid foundation in the basic principles of computer science, including hardware, software, and networking.
- Develop Proficiency in Office Management Tools such as word processors, spreadsheets, presentation tools, and database management systems.
- Apply Information Technology in Business Contexts to Understand how information technology can be applied to solve business problems and improve organizational efficiency.

Course Outcomes

- **Proficiency in Word Processing**: Students will learn to create, edit, format, and manage documents using word processing software such as Microsoft Word or Google Docs.
- **Effective Use of Spreadsheets**: Students will gain proficiency in using spreadsheet software like Microsoft Excel or Google Sheets for data entry, manipulation, analysis, and visualization.

- **Presentation Skills**: Students will learn to create engaging and professional presentations using software such as Microsoft PowerPoint or Google Slides, including the use of design templates, multimedia elements, and animation effects.

- **Database Management**: Students will understand the basics of database management using software like Microsoft Access or similar tools, including creating databases, tables, queries, forms, and reports.

- **File Management**: Students will understand file management principles, including organizing, storing, and retrieving files using various office management software.

- **Data Analysis**: Students will gain skills in analyzing data using spreadsheet functions, charts, and data visualization tools.

- **Document Security**: Students will understand the importance of document security and learn how to protect sensitive information using passwords, encryption, and access controls.

- **Understanding Database Concepts**: Students will understand the fundamental concepts of databases, including tables, records, fields, and relationships.

- **Database Design**: Students will learn to design and create databases, including defining tables, setting primary keys, and establishing relationships between tables.

- **Data Entry and Management**: Students will be proficient in entering, updating, and managing data within an Access database, ensuring data integrity and accuracy.

- **Query Creation**: Students will learn to create and run queries to retrieve and manipulate data, including using select, update, delete, and append queries.

- **Form Design**: Students will be able to design and use forms for data entry and navigation, improving user interaction with the database.

- **Report Generation**: Students will learn to generate and format reports to present data in a structured and readable manner.

UNIT-I

The Need and Importance of Office Automation, Role of computer in Office automation and management, Office automation software. Office system user interface, Managing security and privacy in MS Office system, Sharing documents between Office System Components, Different versions of office system, Office management using smart devices.

UNIT- II

MS Word: Word processing, MS-Word features, creating saving and opening documents in Word, interface, toolbars, ruler, menus, keyboard shortcut, editing, previewing, printing & formatting a document, advance features of MS Word, find & replace, using thesaurus, mail merge, handling graphics, tables, converting a Word document into various formats like-text, rich text format, Word perfect, etc.

UNIT- III

MS Excel: Worksheet basics, creating worksheet, entering data into worksheet, data, text, dates, alphanumeric values saving & quitting worksheet, opening and moving around in an existing worksheet, Toolbars and menus, Keyboard shortcuts, working with single and multiple workbook, working with formula & cell referencing, Auto sum, coping formulas, absolute and relative addressing, formatting of worksheet, previewing & printing worksheet, Graphs and Charts, Database, macros, multiple worksheets-concepts.

UNIT- IV

Power Point: Creating and viewing a presentation, managing Slide Shows, navigating through a presentation, using hyperlinks, advanced navigation with action setting and action buttons, organizing formats with Master Slides, applying and modifying designs, adding graphics, multimedia and special effects.

UNIT- V

Microsoft Access: Planning a database (tables, queries, forms, reports), creating and editing database, customizing tables, linking tables, designing and using forms, modifying database structure, Sorting and Indexing database, querying a database and generating reports.

Reference Books:

1. Microsoft; 2007/2010 Microsoft Office System; PHI.
2. Microsoft; Microsoft Office 2007/2010: Plain & Simple; PHI.
3. Sanjay Saxena; A First Course in Computers 2003 Edition; Vikas Pub.
4. Computer Fundamentals by P.K. Sinha, BPB Publication.
5. Computer Fundamentals and Programming in C,Reema Thareja,OXFORD University Press.
6. Introduction to Computer,Peter Norton's, Tata McGraw Hill Publication.
7. MS-Office , Dr. S.S. Shrivastava, Published by Laxmi Publication.
8. Office 2019:In Easy Steps,Michal Price ,BPB Publication.

COURSE CODE – PGD-50-T-105

E-Commerce

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.

2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Understand E-Commerce Fundamentals to gain a comprehensive understanding of the fundamental concepts and theories of e-commerce
- Analyze E-Commerce Infrastructure to learn about the technological infrastructure required for e-commerce, including internet technologies, web development, and e-commerce platforms.
- Develop E-Commerce Strategies to understand how to formulate and implement effective e-commerce
- Implement Online Business Models to explore various online business models and understand how to apply them in real-world scenarios.
- Understand Legal and Ethical Issues to learn about the legal and ethical considerations in e-commerce, including privacy, security, and intellectual property rights.

Course Outcomes

- **Understanding E-Commerce Fundamentals**: Students will understand the basic concepts, types, and models of e-commerce, including B2B, B2C, C2C, and C2B.

- **Knowledge of E-Commerce Technologies**: Students will gain knowledge of the technologies used in e-commerce, including internet infrastructure, web technologies, and online payment systems.

- **Website Development for E-Commerce**: Students will learn the principles of designing and developing e-commerce websites, including user experience (UX) design, web development languages, and e-commerce platforms.

- **Online Payment Systems**: Students will understand the various online payment systems, including credit/debit cards, digital wallets, and cryptocurrencies, and their security mechanisms.

- **E-Commerce Security**: Students will understand the importance of security in e-commerce and learn about various security measures, such as SSL, encryption, authentication, and fraud prevention.

- **Legal and Ethical Issues**: Students will be aware of the legal and ethical issues related to e-commerce, including privacy laws, intellectual property rights, and consumer protection.

- **Supply Chain Management**: Students will understand the role of e-commerce in supply chain management, including inventory management, logistics, and order fulfillment.

- **Customer Relationship Management (CRM)**: Students will learn about CRM strategies and tools used to manage customer interactions, improve customer satisfaction, and foster customer loyalty.

- **Mobile Commerce**: Students will understand the principles and technologies of mobile commerce (m-commerce), including mobile apps, mobile payment systems, and responsive web design.

Unit-I

Basic Concepts : Introduction, Definition, Objectives, Advantages and disadvantages, Traditional commerce Vs E-Commerce, E-Commerce opportunities for industries, Growth of E-Commerce.

Electronic Data Interchange : Concepts of EDI and Limitation, Application of EDI, Disadvantages of EDT, EDI model; MIME and Value-Added Network, Internet-based EDT.

Unit-II

E-Commerce Models: B2C,B2B, C2C, C2B, other models — Brokerage Model, Aggregator Model, Info-mediatory Model, Community Model and value chain Model.

Electronic Payment Systems: Special features required in payment systems, Types of E-payment systems, E-Cash, E-cheque, credit card, Smart Card, Electronic Purses, e-Billing.

Unit-III

E-Transition Challenges in Indian Corporate, E-Commerce and WWW, e- Marketing, E-Customer Relationship Management, E-CRM Problems and Solutions, CRM Capabilities and Customer life cycle, E-Supply Chain Management.

E-Strategy Planning the E-Commerce Project, E-Business Strategy and Data Warehousing & Mining. Customer-effective Web Design.

Unit- IV

M-Commerce: Overview of mobile-Commerce, Mobile Delivery Technology & Switching Methods, m-Commerce Security issues, Mobile ATM(ICICI Bank Case Study). Applications of M-Commerce: Mobile Financial Applications, m-wallet, Mobile Shopping. Case-Study of an e-commerce application.

Unit- V

Security Issues in E-Commerce: Network and Website Security Risks, Security risk of E-Commerce, Types of threats, Security tools and risk management approach. Cyber laws, Business Ethics, Social Ehics, IT Acts of the India.

Suggested Books:

(1) Bharat Bhaskar, Electronic Commerce — Framework Technologies and Applications, Tata McGraw Hill.
(2) Ravi Kalakota & A.B. Whinston, Frontiers of Electronic Commerce, Pearson Education.
(3) Ravi Kalakota & A.B. Whinston, Electronic Commerce — A Manager's Guide, Pearson Education.
(4) Agarwala Kamlesh, N and Agarwala Deeksha, Business on the Net Introduction to the E-Com., Macmillan India.
(5) P. T. Joseph, E-Commerce: A Managerial Perspective, PHI.

COURSE CODE – PGD-50-T-106

Database Management Systems

Theory: 3 hours per week

Note: Examination: Theory Paper – 3 hours; Max. Marks – 100

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

Upon finishing the course, students will be able to:

- Upon finishing the course, students will be able to:
- To study types of MySQL databases (Document oriented, keyValue pairs, Column-oriented and Graph)
- To understand detailed architecture, define objects, load data, query data and performance tune MySQL databases.
- Able to handle large volume of data through queries.

Course Outcomes

- **Understanding DBMS Concepts**: Students will understand the fundamental concepts and architecture of database management systems, including data models, schemas, and instances.
- **Data Modeling and ER Diagrams**: Students will learn to create data models using Entity-Relationship (ER) diagrams to represent the structure and relationships of data within a database.
- **Relational Database Design**: Students will understand the principles of relational database design, including normalization, primary and foreign keys, and referential integrity.
- **SQL Proficiency**: Students will gain proficiency in using Structured Query Language (SQL) to create, query, update, and manage relational databases.
- **Database Design and Implementation**: Students will learn to design and implement databases, including creating tables, defining constraints, and setting up indexes.
- **Transaction Management**: Students will understand the concepts of transactions, concurrency control, and recovery mechanisms to ensure data consistency and reliability.

- **Practical Application**: Students will apply their knowledge through practical projects, demonstrating their ability to design, implement, and manage databases to solve real-world problems.

Unit-I

Overview of DBMS: Basic concepts, Database system architecture, Schemas, Instances, Components, Database users, Three-tier architecture, Centralized, Distributed and Client/Server architecture, Data independence. Database models: Entity relationship model, hierarchical model, relational model, network model, Object-Oriented data model.

Unit II

Data Modeling using ER Model: ER model concepts, ER diagram, mapping constraints, Keys, Generalization, aggregation, reduction of ER diagrams to tables, extended ER model, Relationship of higher degree. Enhanced ER Model : Concepts, Specialization, Generalization, Data abstraction, Knowledge representation and University EER Model as example.

Unit-III

Relational Model : Concepts, Constraints, Languages, Relational database design by ER & EER mapping, Relational algebra relational calculus.
Normalization : Normal forms – First, second, third and BCNF.

Unit-IV

Transaction processing : Transactions atomicity, durability, serializability and isolation. Concurrency control techniques – Two phase locking, timestamp ordering, multiversion, Granularity locking techniques, Database recovery techniques based on deferred & immediate updates and shadow paging.

Unit-V

SQL: Characteristics of SQL, advantages, data types in SQL, SQL Operators, types of SQL commands, Tables indexes, Views Nulls, Aggregate Functions, Select statement, Sub queries, Insert, Update and Delete operations, Joins, Unions. .

Reference Books:

1. Korth H F and Silberschataz A, System Concepts, Sixth Edition; McGraw Hill.
2. Leon, and Leon, SQL Tata McGraw Hill Pub. Co. Ltd.
3. Ivan Bayross; SQL/PL 4th Edn: BPB,2018.
4. Navathe S.B. Elmasri R,; Fundamentals of Database Systems, Fifth Edition, Pearson.
5. Ramakrishan and Gharke, Database Management Systems, 3rd Edition, Tata Mc Graw Hill.
6. Data C J Database Management Systems, Pearson Education Asia.
7. Singh S.K.; Database Systems; I Edition; Pearson.

Practical Subjects in PGDCA Semester-I 2025-26

Each practical paper shall be of 3 hours duration on one day and carry 100 marks for the practical examination. The practical examination will involve 3 exercises, each of 20 marks, practical record of 15 marks and viva-voce examination of 25 marks.

COURSE CODE – PGD-50P-107

1. Programming in C Lab

Recommended Exercises:

1. Programs using I/O statements and expressions.
2. Programs using decision-making constructs.
3. Write a program to find whether the given year is leap year or Not?
4. Write a program to perform the Calculator operations, namely, addition, subtraction, multiplication, division and square of a number.
5. Check whether a given number is Armstrong number or not?
6. Check whether a given number is odd or even?
7. Write a program to perform factorial of a number.
8. Write a C program to find out the average of 4 integers.
9. Show how to display array elements using two dimensional array.
10. Write a C program to perform swapping using function.
11. Display all prime numbers between two intervals using functions.
12. Reverse a sentence using recursion.
13. Write a program in C to get the largest element of an array using the function.
14. Write a C program to concatenate two string.
15. Write a C program to find the length of String.
16. Find the frequency of a character in a string.
17. Write a C program to Store Student Information in Structure and Display it.
18. The annual examination is conducted for 10 students for five subjects. Write a program to read the data and determine the following:
    (a) Total marks obtained by each student.
    (b) The highest marks in each subject and the marks of the student who secured it.
    (c) The student who obtained the highest total marks.

COURSE CODE - PGD-50P-108

2. Office Management Lab

Recommended Exercises:

1. Create a news-paper document with at least 200 words,
   a. Use margins as, top:1.5, bottom:2, left:2, right:1 inches.
   b. Use heading "Gandhi Jayanti", font size: 16, font color: red, font face: Arial Black.
   c. With first letter "dropped" (use drop cap option) of the first paragraph containing a picture at the right side
   d. Use three columns from the second paragraph onwards till the half of the page.
   e. Then use heading "Computer basics"
   f. Create paragraph using two columns till the end of the page.
2. Create a flowchart using,
   a. Proper shapes like ellipse, arrows, rectangle, and parallelogram.
   b. Use grouping to group all the parts of the flowchart into one single object.
3. Create a table using table menu with,
   a. At least 5 columns and 10 rows.
   b. Merge the first row into one cell.
   c. Merge the second row into one cell, then split the second row into three cells.
   d. Use proper table border and color.
   e. Insert proper content into the table with proper text formatting.
4. Create a letter, which must be sent to multiple recipients.
   a. Use Mail-Merge to create the recipient list.
   b. Use excel sheet to enter the recipient.
   c. Start the mail merge using letter and directory format. State the difference.
5. Create and Editing Worksheets by using Excel
6. Creating Excel Sheets with Formula and Functions
7. Sorting and Querying Data
8. Monthly Expenses Report using Excel
9. Calculation of Ratios using basic functions in Excel – Average, Count, Minimum
10. Use of Sum, Sumif, Count and Count if functions
11. Create a power-point presentation with minimum 5 slides.
    a. The first slide must contain the topic of the presentation and name of the presentation.
    b. Must contain at least one table.
    c. Must contain at least 5 bullets, 5 numbers.
    d. The heading must be, font size:32, font-face: Arial Rounded MT Bold, font-color: blue.
    e. The body must be, font size: 24, font-face: Comic Sans MS, font-color: green.
    f. Last slide must contain „thank you".

12. Create a power-point presentation with minimum 10 slides

    a. Use word art to write the heading for each slides.

    b. Insert at least one clip-art, one picture

    c. Insert at least one audio and one video

    d. Hide at least two slides

13. Create a power-point presentation with minimum 5 slides

    a. Use custom animation option to animate the text; the text must move left to right one line at a time.

    b. Use proper transition for the slides.

14. Create a database "Student" with,

    a. At least one table named "mark sheet" with field name "student name, roll number, mark1, mark2, mark3, mark4, total"

    b. The data types are, student name: text, roll number: number, mark1 to mark4: number, total: number. Roll number must be the primary key.

    c. Enter data in the table. The total must be calculated using update query.

    d. Use query for sorting the table according to the descending/ascending order of the total marks.

15. with addition to the table above,

    a. Add an additional field "result" to the "mark sheet" table.

    b. Enter data for at least 10 students

    c. Calculate the result for all the students using update queries, if total>=200, then pass, else fail.

    d. Search the students, whose name starts with "sh".

    e. Show the names and total marks of the students who have passed the examination.

COURSE CODE - PGD-50P-109

3. DBMS Lab

Recommended Exercises

1. Implementation of DDL commands of SQL with suitable examples
    - **a.** Create table
    - **b.** Alter table
    - **c.** Drop Table
2. Implementation of DML commands of SQL with suitable examples
    - **a.** Insert
    - **b.** Update
    - **c.** Delete
3. Implementation of different types of function with suitable examples
    - **a.** Number function
    - **b.** Aggregate Function
    - **c.** Character Function
4. Implementation of different types of operators in SQL
    - **a.** Arithmetic Operators
    - **b.** Logical Operators
    - **c.** Comparison Operator
    - **d.** Special Operator
    - **e.** Set Operation
5. Study & Implementation of different types of constraints.
6. Study and Implementation of
    - **a.** Group By & having clause
    - **b.** Order by clause

Course Content for PGDCA Second Semester 2025-26

COURSE CODE - PGD-50-T-201

Java Programming

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Learn the Object-Oriented Programming concepts to write, compile and debug programs using Java language.
- Apply the concepts of object-oriented programming like polymorphism, inheritance, Exception Handling, and Multithreading.
- Design and develop console and GUI applications using Java Programming Language.
- Work on programming project as individual or as team member is design, development and implementation phase.

Course Outcomes

- **Understanding Java Fundamentals**: Students will grasp the fundamental concepts of Java programming, including syntax, data types, variables, operators, and control structures.
- **Object-Oriented Programming (OOP)**: Students will understand and apply principles of object-oriented programming, such as encapsulation, inheritance, polymorphism, and abstraction.
- **Proficiency in Java Language Features**: Students will gain proficiency in using Java language features, including classes, objects, methods, constructors, packages, and access modifiers.
- **Exception Handling**: Students will learn to handle exceptions and errors in Java programs using try-catch blocks, throw and throws keywords, and understanding checked

and unchecked exceptions.

- **Multithreading**: Students will learn the basics of multithreading in Java, including creating and managing threads, synchronization, and thread safety.
- **File Handling**: Students will gain skills in reading from and writing to files using Java I/O streams, understanding file handling concepts like file input/output, serialization, and deserialization.
- **GUI Development**: Students will be able to develop graphical user interfaces (GUIs) using Java including creating windows, dialogs, menus, and event handling.
- **Networking**: Students will learn about networking concepts in Java, including Java Beans, Servlets, client-server communication, and using Java APIs for networking tasks.
- **Database Connectivity (JDBC)**: Students will understand how to connect Java applications to databases using JDBC (Java Database Connectivity), including executing SQL queries and handling database transactions.

UNIT – I

Introduction to OOP : Basic concepts of Object Oriented Programming , Objects and Classes, Data abstraction and Encapsulation, Inheritance, Polymorphism, Dynamic binding, Message communication: Benefits & applications of OOP.

Introduction to Java : History, Java features, java Enviroment-JDK, API. JVM, Byte code interpretation, Types of Java programs, Data types, type casting, operators (Arithmetic, increment, decrement, relational, logical, bit wise, conditional) and expressions.

UNIT – II

Decision Making and Branching : Decision making and branching (if…else, else if, switch), looping, wrapper classes, Vectors, Arrays, Types of arrays, String Methods, String Buffer Class.

Class and Objects : Defining a class, Methods, Creating Objects, Accessing class members, constructors, overriding methods, Static members, nesting of methods, this keyword.

UNIT – III

Inheritance : Define a subclass, deriving a sub class, Types and applications, Overriding methods, Final variables and methods, final classes, Finalize methods, Abstract methods and classes, Visibility Control- Public access, Private access, protected and default access, Defining interface, Extending interface, Implementing Interface, Accessing interface variables.

Multithreading: Creating treads, life of a thread, defining & running thread, thread methods, thread priority, synchronization, implementing run-able interface, thread scheduling.

.

UNIT – IV

Packages: Java API Packages-System Packages, Naming Conventions, Creating & Accessing a Packages, Finding Packages and CLASSPATH, Adding Class to a Packages, Hiding Classes.

JAVA Streams : Data Flow with Java Streams, Input Streams, Output Streams.

Exception Handling : Exception-Handling fundamentals, Exception types, try, catch, throw, finally, creating exception sub classes.

UNIT – V

AWT controls (Button, Labels, Combo box, list and other Listeners), Layout and component managers, Event handling, string handling (only main functions), graphic programming (line, rectangles, circle, and ellipses).

Networking: Java utility for networking, Manipulating URLs, reading a file on a Web server. Establishing simple Client Server. Introduction to Java Beans BDK, JAR files, Servlets Life cycle of servlet, JDBC connectivity.

Recommended Text Books

1. Mastering java 2, BPB Publications. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies
2. Java Programming John P. Flynt Thomson 2nd
3. The complete reference JAVA2, Herbert schildt. TMH
4. Arnold,Gosling, " The Java Programming Professional 2000", Addison Wesley Publication
5. C.Thomas wu, "An introduction to oop with Java", TMH

COURSE CODE- PGD - 50-T-202

Web Application Development

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Comprehend the optimal technologies for addressing web client/server challenges.
- Evaluate and create real-time web applications.
- Utilize JavaScript for dynamic effects and form input validation.
- Analyze and select suitable client-side and server-side application technologies.

Course Outcomes

- **Understanding Web Development Fundamentals**: Students will understand the foundational concepts of web development, including client-server architecture, HTTP protocol, and web standards.
- **Proficiency in HTML and CSS**: Students will gain proficiency in using HTML (HyperText Markup Language) for creating the structure of web pages and CSS (Cascading Style Sheets) for styling and layout.
- **Client-Side Scripting (JavaScript)**: Students will learn JavaScript programming language to add dynamic behavior to web pages, handle user interactions, and validate form inputs.
- **Testing and Debugging**: Students will learn techniques for testing web applications, including unit testing, integration testing, and debugging tools provided by IDEs and browsers.
- **User Experience (UX) Design**: Students will understand the principles of UX design and usability testing to create intuitive and user-friendly web interfaces that enhance user satisfaction.

Unit-I

Introduction to Web Technologies: Creating and Maintaining Web Sites; Planning, Navigation and Themes, Site types and Architecture, Elements of a Web page, publishing and publicizing site/structuring web site. Search Engine Optimization,, Site Maps and other Navigation Aid, Site Delivery and Management, Web Standards & W3C recommendations.

Unit-II

HTML Fundamentals: Introduction to HTML, Creating HTML Pages, incorporating Horizontal Rules and Graphical Elements, Hyper-links, Creating HTML Tables, Creating HTML Forms, HTML and Image Techniques, HTML and Page, Frames, Development of Website and Webpage

(Planning, Navigation and Themes, Elements of a Web page, steps of creating a site, publishing and publicizing site structuring web site.

Unit-III

Cascading Style Sheets: Understanding Style Sheets, CSS Syntax and Applying Style Sheets to HTML document, Developing Style Sheets: inline, internal and external. <DIV> tag, Using class and ID, Styling Backgrounds, Styling borders, Styling Text, Styling Fonts, Styling Links, Styling Lists, Styling Tables, Margin.

Unit-IV

Java script: Introduction to scripting language, memory concepts, arithmetic decision making. Java script control structures, Java script functions, events, program modules in java script, function definitions duration of identifiers, scope rules, Controlling Programming Flow, recursion java script global functions.

UNIT V

Java script & arrays: introduction, array declaring and allocating memory, passing arrays to functions, multiple subscripted arrays. The Java Script Object Model Java Script language Objects, Developing Interactive Forms, Cookies and Java Script Security Controlling Frames in Java Script, Client – Side Java Script Custom, JavaScript Objects

Recommended Books:

1. M.L. Young; Complete Reference b: Internet; 2nd Edition; Tata McGraw Hill.
2. Thomas A; Powel: Web Design ; C.R. : Second Edition TMH..
3. Thomas A. Powel : HTML & XHTML : C.R. Fourth Edition; TMH.
4. Mastering HTML 4.0 by Deborah S.Ray an Eric J. Ray From BPB
5. G. Roverston; Hands on HTML., BPB Publication
6. D.A. Tauber, B. Kienan; Microsoft From Page 2000, BPB Publications.
7. Joel Sklar: Principles of Web Design BPB Publication

COURSE CODE – PGD-50-T-203

Software Engineering

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Understand the principles and practices of software engineering.
- Apply software engineering processes and methodologies to develop software systems.
- Perform requirements analysis and software design.
- Implement software using appropriate programming languages and development tools.
- Apply software testing and quality assurance techniques.

Course Outcomes

- **Understanding Software Development Lifecycle (SDLC)**: Students will understand the phases of the software development lifecycle, including requirements analysis, design, implementation, testing, deployment, and maintenance.
- **Software Requirements Engineering**: Students will learn techniques for gathering, analyzing, and documenting software requirements, ensuring that they meet stakeholders' needs and expectations.
- **Software Design Principles**: Students will understand and apply principles of software design, including modularization, abstraction, encapsulation, and design patterns to create scalable and maintainable software systems.
- **Software Testing and Quality Assurance**: Students will learn testing techniques, including unit testing, integration testing, system testing, and acceptance testing, to ensure software quality and reliability.
- **Software Project Management**: Students will gain knowledge of project management methodologies (e.g., Waterfall) and tools to plan, schedule, monitor, and control software projects effectively.
- **Software Metrics and Estimation**: Students will understand metrics for measuring software complexity, quality, and performance, as well as techniques for estimating project effort, time, and resources.
- **Risk Management**: Students will learn techniques for identifying, analyzing, and managing risks associated with software projects, including risk mitigation strategies.
- **Software Maintenance and Evolution**: Students will understand the challenges and strategies involved in software maintenance, including bug fixing, enhancements, updates, and software evolution over time.

Unit-I

Software Engineering Fundamentals: Software, Problem Domain, Software Engineering Challenges. Software Processes (processes, projects & products, component).

Software Development Process Models: Waterfall Model, Prototyping, Iterative Enhancement Model, Spiral Model. Introduction to Agile Model: Principles, Steps, Various Agile Process Models.

Software Requirement Analysis & Specification: Need, Characteristics & Components. Introduction to Requirements Modeling: Data Flow Diagram and Use Cases.

Unit-II

Introduction to Metrics: Function Point, Line of Code (LOC) and KLOC.

Software Project Planning: Cost Estimation- Uncertainties in Cost Estimation, Building Cost Estimation Models, On Size Estimation, COCOMO Model.

Project Scheduling: Average Duration Estimation, Project Scheduling & Milestones. Quality Assurance Plans: Verification & Validation, Inspection & Reviews.

Unit-III

Design Engineering: Design Process & Design Quality, Design Concepts (abstraction, architecture, patterns, modularity, information hiding, functional independence, refinement, refactoring, and design classes), The Design Model (data design elements, architectural design elements, interface design elements, component-level design elements, deployment-level design elements).

Unit-IV

Testing Strategies & Tactics: A strategic approach to software testing, Strategic issues, Software testing fundamentals, Test characteristics, Test Strategies for conventional software: Unit Testing, Integration testing, Validation Testing, System testing, Black-Box testing, White Box testing.

Unit-V

Risk Management: Overview, Assessment, Control.

Software Reliability: Measures of Reliability & Availability, Software Safety.

Maintenance and Reengineering: Introduction to: Software Maintenance, Software Supportability, Reengineering, Reverse Engineering, Restructuring, and Forward Engineering.

Reference /Text Books

1. Pressman, Roger (2001) Software Engineering; A Practitioner's Approach, 8th ed. M Graw-Hill,2014.
2. Sommerville Lan; Software Engineering, 9th Ed. Pearson Education,2014
3. Jalote, Pankaj (7) An integrated Approach to Software Engineering 2nd Ed.
4. James Rumbaugh. MichealBlaha, "Object oriented Modeling and Design with UML", 2nd Edition, 2007.
5. Simon Bennett, Steve McRobb and Ray Farmer, " Object-Oriented Systems Analysis and Design Using UML" 4th Edition,McGraw Hill Education, 2010

COURSECODE - PGD-50-T-204

Data Communication and Computer Networks

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- Understand the basic concepts, types of networks, OSI, ans TCP/IP models with working of all the layers in detail
- Learn and understand the working of different hardware components used in networking and various communication protocols
- Learn and understand various issues involved in network security, and methods used to implement network security.

Course Outcomes

- **Understanding Network Fundamentals**: Students will understand the basic concepts and components of computer networks, including protocols, network architecture, and types of networks (LANs, WANs, MANs).
- **Network Models and Layers**: Students will learn about network models such as the OSI (Open Systems Interconnection) and TCP/IP (Transmission Control Protocol/Internet Protocol) models, and understand the functionality of each layer.
- **Data Transmission Techniques**: Students will learn how data is transmitted over networks, including methods such as modulation, multiplexing, and error detection/correction techniques.
- **Network Devices and Equipment**: Students will become familiar with network devices such as routers, switches, hubs, modems, and understand their roles in network communication.

- **Network Protocols**: Students will learn about common network protocols, including TCP, UDP, IP, HTTP, FTP, SMTP, and DNS, and understand their purpose and operation.

- **Local Area Network (LAN) Technologies**: Students will understand LAN technologies, including Ethernet, Wi-Fi (IEEE 802.11), and token ring, and their implementations in network environments.

- **Wide Area Network (WAN) Technologies**: Students will learn about WAN technologies such as leased lines, ISDN, DSL, cable modem, and understand their applications in connecting geographically dispersed networks.

- **Internet and Web Technologies**: Students will understand how the Internet works, including its infrastructure, domain names, IP addressing, and web technologies such as HTTP/HTTPS, HTML, and web browsers.

- **Network Management**: Students will learn about network management principles and tools for monitoring, configuring, and troubleshooting network devices and services.

UNIT-I

Data Communications & Network Models : Data Communications: Components, Data Representation and Data flow; Networks: Distributed Processing, Network Criteria, Network Models, Categories of networks and Internetwork; Internet and Protocols and Standards.

Network Models: Layered tasks, the OSI model, Layers in the OSI Model, TCP/IP protocol Suit, Addressing.

Unit – II

Data and Signals & Digital Transmission : Data and Signals: Analog and Digital Data, Analog and Digital Signals, Periodic and Non periodic Signals, Transmission impairment, Data rate limits and Performance. Transmission modes.

Transmission Media : Guided media (Twisted Pair Cable, Coaxial Cable & Fiber-Optic Cable) and Unguided media :Radio wave, Infrared, Microwave Communication, Satellite, Geosynchronous Satellites Communication and optical fiber communication.

UNIT-III

Multiplexing & Switching : Digital Transmission: Digital to Digital Conversion:- Line coding(Unipolar, Polar & Bipolar), Block Coding, Analog to Digital Conversion: PCM & DM, Digital to analog conversion : ASK,FSK,PSK & QAM, Analog to Analog conversion: Amplitude Modulation, Frequency Modulation & Phase Modulation.

Multiplexing: FDM, WDM, Synchronous TDM and Statistical TDM.

UNIT-IV

Error Detection and Correction : Switching: Circuit switched networks, message switching & packet switching. Datagram networks, Virtual Circuit networks. Error Detection and Correction: Introduction, Block coding: Hamming Distance & Parity bit, linear block codes, cyclic codes: CRC, VRC & LRC, and Checksum.

Data Link Control : Data Link control: Framing, Introduction of Flow and Error Control. Elementary Data Link Protocols: - Simplest Protocol, Stop & Wait Protocol and Simplex protocol for a Noisy channels.

UNIT-V

Networks Layer Functions and Protocols: Routing, Routing algorithms, Network layer protocol of Internet- IP protocol, Internet control protocols.

Transport Layer Functions and Protocols: Transport services, Berkeley socket interface overview, Transport layer protocol of Internet- UDP and TCP. Overview of Application layer protocol:Overview of DNS protocol, Overview of WWW & HTTP protocol.

Recommended Books :

1. Behrouz A Foruzan, Data Communication and Networking; 3rd Edition; Tata McGraw Hill.
2. Behrouz A Foruzan, TCP/IP Protocal Suite; 2nd Edition; Tata McGraw Hill.
3. Stalling William; Data and Computer Communication; 8th Edition Pearson.
4. Tannenbasum; Computer Networks; 4th edition, PHI.
5. Wayne tomasim electronic Communications Systems, Pearson, Education Asia.
6. M.A. Miller, Data and Netowork Communications, Thomosn Kearning
7. Gilbert Held, Understanding Data Communication, Techmedia.
8. Fred Harshal, Data Communications Communications, Networks, Pearson Education Asia.

Practical Subjects in PGDCA Semester II 2025-26

Each practical paper shall be of 3 hours duration on one day and carry 100 marks for the practical examination. The practical examination will involve 3 exercises, each of 20 marks, practical record of 15 marks and viva-voce examination of 25 marks.

COURSE CODE - PGD-50-P-207

1. Java Lab

Recommended Exercises

1. Write a Java Program to find the reverse of a given number. And also check whether it is palindrome or not.
2. Write a Java Program to print Fibonacci sequence (rule: The first two values in the sequence are 0 and 1. Every subsequent value is the sum of the two values preceding it.)
3. Write a program in java to implement Constructors and destructors
4. Write a Java program to find both the largest and smallest number in a list of integers.
5. Write a Java program to find the volume of a box by creating objects.
6. Write a Java program to implement all arithmetic operations with static/class methods for each operation. User must provide values from Keyboard.
7. Write a Java program to implement parameter passing techniques:
   a) call-by-value b) call-by-reference
8. Write a Java program to implement the method overloading.
9. Write a Java program to implement inheritance.
10. Write a Java program to demonstrate method overriding by implementing dynamic method.
11. Write a Java program to create and abstract class called Shape which inherits Rectangle and triangle to calculate area of each shape by implementing abstract method of Shape class by implementing hierarchical inheritance.
12. Write a program in Java to implement Packages

COURSE CODE - PGD-50P-208

2. Web Development Lab

Recommended Exercises:

1. Define the basic Structure of HTML.
2. Implement the Heading Element in HTML.
3. Write a program to insert image in HTML.
4. Write a program to implement Formatting text in HTML.
5. Write a program for changing Background and Style.
6. Write a program to create a Table in HTML.
7. Write a program to implement Lists in HTML.
8. Write a program to implement Frameset in HTML.
9. Write a program to Create Form in HTML
10. Write a program to implement Inline, Internal and External CSS.

Elective – III Lab

1. Python Lab

Recommended Exercises

1. Program to Find the biggest number among three numbers
2. Program to print all the prime numbers with in the given range
3. Python script to display the grade of student using obtained marks.
4. Write a Python script to find the minimum and maximum elements without using built-in operations in the lists
5. Write a Python script to create a list with different data types
6. Write a Python script to find the repeated items of a tuple.
7. Write a Python script to perform Union, Intersection, difference and symmetric difference of given two sets.
8. Write a Python script to concatenate following dictionaries to create a new one.
9. Write a Python script to sort (ascending and descending) a dictionary by value
10. Write a Python script to handle simple errors by using exception handling mechanism
11. Implement python script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it
12. Implement python script that takes a list of words and returns the length of the longest one.
13. Write a python script that read a text file and return those lines which start with a specific letter or specific word.
14. Write a Python script to check that a string contains only a certain set of characters (in this case a-z, A-Z and 0-9).

COURSE CODE - PGD - 50 - P - 209

2. PHP Lab

Recommended Exercises

Part-A

Write PHP script for the following:-

1. Implement simple calculator
2. Find the largest of three numbers
3. The sum of digits of given number
4. Check whether the given number is prime or not
5. The factorial of a given number
6. Fibonacci series of a given number
7. Get name of the user from a form and show greeting text.
8. To check whether given number is palindrome or not.
9. To find largest values of two numbers using nesting of function.
10. Write a Mathematical calculator program.
11. To check whether given number is String palindrome or not.

Part-B

1. Create a PHP page for login page without sql connection.
2. Write a php program to Array manipulation.
3. Create a PHP page for login page with sql connection.
4. Write a php program to Read from existing file.
5. Write a php program to write a file.
6. Write a php program to calculate Date and Time function .
7. Write a php program to design Curriculum Vitae.

COURSE CODE - PGD - 50 - P - 209(A)

3. Digital Marketing Lab

Recommended Exercises

1. Digital Marketing Implementation in Business Scenario
2. Create the Digital Marketing Webpage
3. Conducting the Search Engine Optimization and Search Engine Marketing
4. Using Google Analytics to analyze website performance
5. Creating Promotional banner through Canva
6. Facebook Promotion using banners
7. Creating YouTube Channel for Marketing
8. Twitter Marketing
9. Instagram Marketing
10. Email Marketing

Elective Theory papers for Elective Group-I of PGDCA Second Semester

Elective I (Any One)

COURSE CODE - PGD-50-T-205

Data Structures and Algorithms

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- To design efficient algorithms using various algorithm designing strategies
- To analyze the problem and develop the algorithms related to these problems
- To classify the problem and apply the appropriate design strategy to develop algorithm
- To design algorithm in context of space and time complexity and apply asymptotic notation

Course Outcomes

- **Understanding Data Structures**: Students will understand various data structures such as arrays, linked lists, stacks, queues, trees, graphs, and hash tables, including their operations and applications.

- **Algorithm Design and Analysis**: Students will learn to design, analyze, and implement algorithms to solve computational problems efficiently, including understanding time and space complexity.

- **Sorting and Searching Algorithms**: Students will gain proficiency in sorting algorithms (e.g., bubble sort, selection sort, merge sort, quick sort) and searching algorithms (e.g., linear search, binary search).

- **Graph Algorithms**: Students will learn algorithms related to graphs, such as traversal algorithms (depth-first search, breadth-first search), shortest path algorithms (Dijkstra's algorithm, Bellman-Ford algorithm), and minimum spanning tree algorithms (Prim's algorithm, Kruskal's algorithm).

- **Recursion and Backtracking**: Students will understand recursion and backtracking techniques and use them to solve problems where solutions depend on previous choices.
- **Data Structures for Efficiency**: Students will learn to choose appropriate data structures based on problem requirements to optimize operations such as insertion, deletion, searching, and sorting.
- **Memory Management**: Students will understand memory allocation and deallocation strategies in data structures and algorithms, including dynamic memory management techniques.
- **Complex Data Structures**: Students will explore complex data structures such as AVL trees, B-trees, priority queues, understanding their advantages and applications.
- **Algorithmic Problem Solving**: Students will develop problem-solving skills by applying data structures and algorithms to solve a variety of computational problems efficiently.

UNIT – I

Introduction to Algorithm Design: Algorithm, its characteristics, efficiency of algorithms, analyzing Algorithms and problems.

Linear Structure: Arrays, records, stack, operation on stack, implementation of stack as an array, queue, types of queues, operations on queue, implementation of queue.

UNIT – II

Linked Structure : List representation, Polish notations, operations on linked list - get node and free node operation, implementing the list operation, inserting into an ordered linked list, deleting, circular linked list, doubly linked list, implementation of stack and queues using linked list.

UNIT – III

Tree Structure : Concept and terminology, Types of trees, Binary search tree, inserting, deleting and searching into binary search tree, implementing the insert, search and delete algorithms, tree traversals , Huffman's algorithm.

UNIT – IV

Graph Structure : Graph representation - Adjacency matrix, adjacency list, Warshall's algorithm , adjacency multilist representation. Orthogonal representation of graph . Graph traversals - BFS and DFS. Shortest path, all pairs of shortest paths, transitive closure, reflexive transitive closure.

UNIT – V

Searching and sorting : Searching - sequential searching, binary searching, hashing. Sorting - selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort, efficiency considerations.

Recommended reference books

1. S. Lioschutz: Data Structures, Mc Graw Hill International Edition.
2. A.V. Aho., J.E. Hopcroft, and J.D. Ullman, Data Structures and Algorithms, Pearson Education Asia.
3. R.S. Salaria; Data Structures & Algorithms Using C ;Fourth Edn;Khanna Book Pub.
4. R.B. Patel; Expert Data Structures with C ;Fourth Edn;Khanna Book Pub.

A. Michael Berman: Data Structures via C++, Oxford University Press.

5. Sara Baase and Allen Van Gelder: Computer Algorithms, Pearson Education Asia.

COURSE CODE - PGD-50-T-205 (A)

Cloud Computing

Theory: 3 hours per week

Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- To understand the principles and paradigm of Cloud Computing
- Ability to design and deploy Cloud Infrastructure
- Understand cloud security issues and solutions
- Ability to understand role of Virtualization Technologies
- Design & develop backup strategies for cloud data based on features

Course Outcomes

- **Understanding Cloud Computing Fundamentals**: Students will grasp the basic concepts and principles of cloud computing, including virtualization, elasticity, scalability, and on-demand provisioning of resources.
- **Cloud Service Models**: Students will understand different cloud service models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), and their implications for deployment and management of applications.
- **Cloud Deployment Models**: Students will learn about various cloud deployment models, including public cloud, private cloud, hybrid cloud, and community cloud, and understand their advantages and considerations.

- **Cloud Security and Compliance**: Students will learn about cloud security challenges and best practices, including data encryption, identity and access management (IAM), network security, and compliance with regulatory requirements (e.g., GDPR, HIPAA).
- **Scalability and Performance Optimization**: Students will learn techniques for scaling applications in the cloud to handle varying workloads and optimizing performance using auto-scaling, load balancing, and caching mechanisms.
- **Emerging Trends in Cloud Computing**: Students will stay updated on emerging trends and innovations in cloud computing.

Unit-I

Introduction of Cloud Computing: Cloud computing, Enabling Technology, Vision, Characteristics and components of Cloud Computing. Challenges and Approaches of Migration into Cloud. Types of Clouds, Services models, Cloud Reference Model.

Unit-II

Cloud Computing Architecture: Data center Design and interconnection Network, Architectural design of Compute and Storage Clouds. Cloud Programming and Software: Features of cloud programming, Parallel and distributed programming paradigms-MapReduce, Hadoop, High level Language for Cloud. Service Oriented Architecture – REST and Systems of Systems – Web Services – Publish Subscribe Model

Unit-III

Virtualization Technology: Definition, Understanding and Benefits of Virtualization. Implementation Level of Virtualization, Virtualization Structure/Tools and Mechanisms, Hypervisor VMware, KVM, Xen. Virtualization: of CPU, Memory, I/O Devices, Virtual Cluster and Resources Management, Virtualization of Server , Desktop, Network, and Virtualization of data-center.

Unit-IV

Securing the Cloud: Cloud Information security fundamentals, Cloud security services, Design principles, Policy Implementation, Cloud Computing Security Challenges, Cloud Computing Security Architecture. Legal issues in cloud Computing. Data Security in Cloud: Risk Mitigation, Understanding and Identification of Threats in Cloud, SLA-Service Level Agreements, Trust Management.

Unit-V

Cloud Platforms in Industry: Amazon web services, Google AppEngine, Microsoft Azure Design, Aneka: Cloud Application Platform -Integration of Private and Public Clouds Cloud applications: Protein structure prediction, Data Analysis, Satellite Image Processing, CRM and ERP, Social networking.

Recommended Books:

1. Cloud Computing ,Principle and Paradigms, Edited By RajkumarBuyya, JemesBroberg, A. Goscinski, Pub.- Wiley-2016
2. Kumar Saurabh, "Cloud Computing" , Wiley Pub 2016
3. Distributed and Cloud Computing, Kai Hawang , GeofreyC.Fox, Jack J. Dongarra Pub: Elservier, 2013
4. Krutz , Vines, "Cloud Security " , Wiley Pub,2010
5. Velte, "Cloud Computing- A Practical Approach" ,TMH Pub,2009
6. Katarina Stanoevska-Slabeva, Thomas Wozniak, SantiRistol, "Grid and Cloud Computing – A Business Perspective on Technology and Applications", Springer,2010

COURSE CODE-PGD-50-T-205 (B)

Computer Architecture

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

Upon finishing the course, students will be able to:

- The students will be able to understand digital logic design, including logic elements, and their use in combinational and sequential logic circuit design, the basic architecture of processing, memory and I/O organization in a computer system.
- Understand the design of Sequential and Arithmetic Circuits.
- Understand the functions of CPU and I/O devices.
- Understand the operations and structure of Memory.

Course Outcomes

- **Understanding Boolean Algebra**: Students will learn the basic principles of Boolean algebra, including Boolean variables, Boolean operations (AND, OR, NOT), Boolean laws (commutative, associative, distributive), and De Morgan's laws.

- **Boolean Expressions**: Students will understand how to simplify and manipulate Boolean expressions using algebraic techniques, Karnaugh maps (K-maps), and Boolean algebra theorems to optimize digital circuit designs.

- **Logic Gates**: Students will learn about basic logic gates (AND, OR, NOT), as well as derived gates (NAND, NOR, XOR, XNOR), and understand their truth tables, symbolic representation, and logical operations.

- **Combinational Logic Circuits**: Students will design and analyze combinational logic circuits using logic gates, multiplexers, demultiplexers, encoders, decoders, and understand how data flows through these circuits based on input conditions.

- **Sequential Logic Circuits**: Students will learn about sequential logic circuits, including flip-flops (SR, JK, D, T), registers, counters, and understand their role in storing and processing digital data over time.

- **CPU Architecture**: Students will understand the architecture of the CPU, including its components such as ALU (Arithmetic Logic Unit), registers, control unit, and how instructions are fetched, decoded, and executed.

- **Input/Output Systems**: Students will gain knowledge of I/O subsystems, including interfaces, controllers, DMA (Direct Memory Access), and how data is transferred between peripherals and memory or CPU.

- **Bus Architecture**: Students will understand bus architecture, including different types of buses (e.g., address bus, data bus, control bus), bus protocols, and their role in communication between CPU, memory, and peripherals.

- **Interrupts and Interrupt Handling**: Students will learn about interrupts, interrupt handling mechanisms, interrupt priorities, and how interrupts are used for asynchronous events in computer systems.

- **Virtualization and Memory Management**: Students will understand virtual memory concepts, paging, segmentation, and memory protection mechanisms, as well as virtualization technologies and their impact on system performance.

UNIT- I

Boolean Algebra and Logic Gates: Basic Gates, Basic laws of Boolean algebra, Simplification of Boolean algebra. Combinational Logic Design: Standards representation for logical expression, Minimization of logical functions in terms of Maxterm and Minterm, Simplifications of Boolean equations using K-maps, don't care conditions

UNIT-II

Arithmetic Circuits: Half Adder, Full Adder, Half Subtractor, Full Subtractor, Parallel Binary Adder, Parallel binary Subtractor, Parallel binary adder/Subtractor. Multiplexers, De-Multiplexers, decoders, encoders (Octal to binary, decimal to BCD, priority), BCD to Seven segment decoder.

Sequential Logic: Sequential circuits: Flip-flops, S-R, D, J-K, T, Clocked Flip-flop, Race around condition, Master slave Flip-Flop (truth tables, working)

UNIT-III

Register Transfer And Micro operations : Register Transfer Language, Register transfer, Bus and Memory transfer, Arithmetic Micro-operations, Logic Micro-operations, Shift Micro-operations, Arithmetic Logic Shift Unit.

CPU design : specifying a CPU, design and implementation of a simple CPU (fetching instructions from memory, decoding and executing instructions, establishing required data paths. Design and implementation of a simple micro sequencer.

UNIT-IV

Basic Computer Organization And Design :Instruction Codes, Computer Registers: Common bus system; Computer Instructions: Instruction formats; Instruction Cycle: Fetch and Decode, Flowchart for Instruction cycle; Register reference instructions. I/O & Interrupt, types of Interrupts, Interrupt cycle. Central Processing Unit: Introduction, General Register Organization, Stack Organization: Register stack, Memory stack; Instruction Formats, Addressing Modes

UNIT- V

Memory Organization : Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory. Multiprocessors: Characteristics of multi-processors inter connection structure; inter processor arbitration, inter-processor communication and synchronization.

Recommended Books

1. M, Morris Mano; Computer System Architectures; III Edition, Prentice Hall of India,2008
2. Andrew S. Tanenbaum , Structured Computer Organization,Printice Hall
3. William Stallings, Computer Organization and Architecture , Sixth Edition, Pearson
4. John D. Carpinelli: Computer Systems Organization & Architecture; 3rd Edition; Person Education Asia,2008
5. Malvino B ; Digital Computer Electronics III Edition; TMHL

Elective Theory papers for Elective Group-II of PGDCA Second Semester

Elective II (Any One)

COURSE CODE - PGD-50-T-206

Python Programming

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- To gain a solid foundation in the basics of Python programming, including syntax, data types, control structures, and functions.
- To enhance problem-solving abilities by applying Python programming techniques to solve a variety of computational problems.
- Learn to efficiently use and implement built-in data structures such as lists, tuples, dictionaries, and sets.
- To Utilize Python for data analysis and scientific computing using libraries such as NumPy, Pandas, and Matplotlib.

Course Outcomes

- **Fundamental Programming Concepts**: Students will grasp fundamental programming concepts such as variables, data types, operators, control structures (loops and conditionals), functions, and exception handling in Python.
- **Object-Oriented Programming (OOP)**: Students will understand and apply principles of object-oriented programming (OOP) in Python, including classes, objects, inheritance, polymorphism, encapsulation, and abstraction.
- **File Handling**: Students will gain skills in reading from and writing to files using Python's file handling capabilities, including text files, CSV files, and JSON files, and understand concepts like file modes and buffering.

- **Modules and Packages**: Students will understand how to create, import, and use Python modules and packages to organize and reuse code, enhancing code maintainability and modularity.

- **Functional Programming**: Students will learn functional programming concepts in Python, including lambda functions, map, filter, and reduce functions, and how to apply them to process data efficiently.

- **Regular Expressions**: Students will gain proficiency in using regular expressions (regex) in Python to search, match, and manipulate strings based on specific patterns, enhancing text processing capabilities.

- **Database Connectivity**: Students will learn how to connect Python applications to databases using database APIs (e.g., SQLite, MySQL, PostgreSQL) and execute SQL queries to retrieve, manipulate, and store data.

UNIT-I

Python Concepts: Origin, Comparison, Comments, Variables and Assignment, Identifiers, Basic Style Guidelines, Standard Types, Internal Types, Operators, Built-in Functions, Numbers and Strings. Sequences: Strings, Sequences, String-Operators & functions, Special Features of Strings, Memory Management, programs & examples.

Conditionals and Loops: if statement, else Statement, elif Statement, while Statement, for Statement, break Statement, continue Statement, pass Statement, else Statement

Unit-II

Object and Classes: Classes in Python, Principles of Object Orientation, Creating Classes, Instance Methods, Class variables, Inheritance, Polymorphism, Type Identification, Python libraries(Strings, Data structures & algorithms).

Lists and Sets: Built-in Functions, List type built in Methods, Tuples, Tuple Operators, Special Features of Tuples, Set: Introduction, Accessing, Built-in Methods (Add, Update, Clear, Copy, Discard, Remove), Operations (Union, Intersection, Difference).

Unit-III

Dictionaries : Introduction to Dictionaries, Built-in Functions, Built-in Methods, Dictionary Keys, Sorting and Looping, Nested Dictionaries.

Files: File Objects, File Built-in Function, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules.

Unit-IV

Regular Expression: Regular Expression: Introduction/Motivation, Special Symbols and Characters for REs, REs and Python.

Excetiptons: What Are Exceptions? Exceptions in Python, Detecting and Handling Exceptions, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions.

Unit-V

Database Interaction : SQL Database Connection using Python, Creating and Searching Tables, Reading and storing config information on database, Programming using database connections, Python Multithreading: Understanding threads, Forking threads, synchronizing the threads, Programming using multithreading.

Recommended Books:

1. R. Nageswara Rao, "Core Python Programming", Dreamtech Press, 2nd Edition, 2018
2. Dr. M. Suresh Anand, Dr. R. Jothikumar, Dr. N. Vadivelan, "Python Programming" , Notion Press, 1stEdition, 2020
3. Martin C. Brown, "The Complete Reference Python", McGraw Hill Education, 4thEdition, 2021.
4. Ashok Namdev Kamthane; "Programming and Problem Solving with Python";2nd Edn., MGH,2020
5. Allen B. Downey, "Think Python", O'Reilly Media, 2016
6. SakisKasampalis, Quan Nguyen, Dr Gabriele Lanaro, Ingram, "Advanced Python Programming", short title, 2019
7. David M. Beazley, "Python Essential Reference", Amazon Books, 2010.
8. M. Lutz, "Programming Python, 4th Edition", O'Reilly Media, 2010

COURSE CODE - PGD-50-T-206(A)

PHP Programming

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

In this course, you will learn about:

- Fundamental concepts of PHP scripting language
- The basic structure of a web application
- The request/response cycle
- Basics of MySQL database

- The relationship between the client-side and server-side scripts
- Creating functional websites and web apps in PHP
- PHP web application testing and security
- Creating a PHP web application using a CMS

Course Outcomes

- **Fundamental PHP Concepts**: Students will grasp fundamental programming concepts in PHP, including variables, data types, operators, control structures (loops and conditionals), functions, and arrays.
- **Object-Oriented Programming (OOP) in PHP**: Students will understand and apply object-oriented programming principles in PHP, including classes, objects, inheritance, polymorphism, encapsulation, and abstraction.
- **Web Development with PHP**: Students will learn to create dynamic web pages and web applications using PHP, including handling form data, sessions, cookies, and integrating with databases.
- **Database Connectivity**: Students will gain proficiency in connecting PHP applications to databases such as MySQL, PostgreSQL, or SQLite, executing SQL queries, and managing database interactions securely.
- **Error Handling and Debugging**: Students will learn techniques for error handling and debugging PHP code, including using error reporting settings, logging errors, and troubleshooting common programming errors.
- **Session Management**: Students will understand how to manage user sessions in PHP applications, including session variables, session timeouts, and techniques for maintaining session security.
- **Integration with Front-End Technologies**: Students will learn to integrate PHP with front-end technologies such as HTML, CSS, JavaScript, and AJAX to create interactive and responsive web applications.

UNIT-I

Introduction to PHP: Installation of PHP and MySQL, PHP configuration in IIS & Apache Web Server. Features of PHP, Writing PHP, How PHP code is parsed, Embedding PHP and HTML Executing PHP and viewing in Browser.

Unit - II

Control Structures: Data types, Operators, PHP variables: static and global variables, Comments in PHP, Control Structures, Condition statements, If…Else, Switch, ? operator, Loops, While, Break Statement Continue. Do…While, For, For each, Exit, Die, Return. Arrays: Numeric, Associative and Multidimensional Arrays.

UNIT-III

Strings: Creating and accessing String, Searching & Replacing String, Formatting String, String Related Library function, Pattern matching, Replacing text, Splitting a string with a Regular Expression

Functions: Defining a Function, Calling a Function, Parameter passing, Returning value from function

UNIT-IV

Form Data Handling: $_GET, $_POST, $_REQUEST Variables, Cookies handling, Session Management

Exception Handling: Understanding Exception and error, Try, catch, throw

UNIT-V

File Handling: Opening and closing a file, Copying, renaming and deleting a file
Database Handling: Connection with MySql Database or ODBC, Performing basic database, operation (Insert, Delete, Update, Select, Truncate Alias, Order By), Setting query parameter.

Reference Books:

1. PHP, The CompleteReference, Steven Holzner, TMH
2. Beginning PHP 5.3, Matt Doyle, John Wiley & Sons
3. Core PHP Programming Leon Atkinson Pearson publishers
4. Beginning PHP 5.0 Database Christopher Scollo, Harish,Rawat, Deepak Thomas,Wrox Press

COURSE CODE – 50-T- 206 (B)

Digital Marketing

Theory: 3 hours per week
Examination: Theory Paper – 3 hours; Max. Marks – 100

Note:

1. Candidate has to attempt six questions in all.
2. Question No. 1 covering whole syllabus will consists of 10 short answer questions carrying 2 marks each taking two questions from each unit.
3. Question No. 2 to 6, each of 16 marks, will be framed by taking one question from each unit. There will be an internal choice within the unit.

Course Objectives

By the end of this course, students will be able to:

- To gain a comprehensive understanding of the core concepts and principles of digital marketing, including its evolution and significance in the modern business landscape.
- Develop Digital Marketing Strategies to formulate and implement effective digital marketing strategies that align with business goals and target audience needs.
- Utilize Search Engine Optimization (SEO) to optimize websites for search engines, improving visibility and organic search rankings.
- Understand the principles and techniques of PPC advertising, including the use of platforms like Google Ads to create, manage, and optimize campaigns.
- Gain proficiency in designing, executing, and analyzing email marketing campaigns to nurture leads and drive customer retention.

Course Outcomes

- **Understanding Digital Marketing Fundamentals**: Students will grasp the basic principles and concepts of digital marketing, including its importance in contemporary business environments and its role in reaching target audiences effectively.
- **Digital Marketing Channels**: Students will learn about various digital marketing channels and platforms, such as search engines (SEO), social media (SMM), email marketing, content marketing, pay-per-click (PPC) advertising, and affiliate marketing.
- **Search Engine Optimization (SEO)**: Students will understand the principles of SEO, including keyword research, on-page optimization, off-page optimization (backlinking), technical SEO, and SEO tools and analytics.
- **Social Media Marketing (SMM)**: Students will learn strategies for creating, managing, and optimizing social media campaigns on platforms like Facebook, Twitter, Instagram, LinkedIn, and others, including content creation and community engagement.
- **Email Marketing**: Students will gain knowledge of email marketing strategies, including list building, segmentation, email design, copywriting, A/B testing, automation, and analyzing email campaign performance.
- **Pay-Per-Click Advertising (PPC)**: Students will learn to create and manage PPC campaigns using platforms like Google Ads (formerly AdWords) and Bing Ads, including keyword bidding, ad creation, targeting options, and campaign optimization.
- **Web Analytics**: Students will learn how to use web analytics tools (e.g., Google

Analytics) to track, measure, and analyze digital marketing campaign performance, including traffic sources, user behavior, conversion rates, and ROI.

UNIT – I

Digital Marketing Fundamentals : Marketing v/s Sales, Marketing Mix and 4 Ps, What is Digital Marketing, CRM platform, CRM models, CRM platform, Marketing Automation, Inbound vs Outbound Marketing, Content Marketing, Understanding Traffic, Understanding Leads, Strategic Flow for Marketing Activities.

Unit-II

Website Planning and Structure : Objective of Website and Flow, One Page Website, Google Analytics, Tracking Code, Website Auditing.

Search Engine Optimization: Basic Concepts, how Search Engine works, Keywords, Keywords, titles, meta tags, On page optimization techniques, Off page Optimization techniques, SEO Audit & Future of SEO.

Unit-III

Email Marketing: Content Writing, Contents Writing Techniques and Tools, Email Machine – The Strategy, Email Frequency, Triggers in Email using 4Ps, Sequence of Email Triggers, Email Software and Tools, Importing Email Lists, Planning Email Campaign, Email Templates and Designs, Sending HTML Email Campaigns, Web Forms Lead Importing, Integrating Landing Page Forms Campaign Reports and Insights.

Unit-IV

Google Adwords : Basics, Google Ad Types, Pricing Models, PPC Cost Formula, Ad Page Rank, Billing and Payments, Adwords User Interface, Keyword Planning, Keywords Control, Creating Ad Campaigns, Creating Text Ads, Creating Ad Groups, Bidding Strategy for CPC.

Unit-V

Social Media Optimization (SMO) : Introduction , Advanced Facebook Marketing, Word Press Blog Creation, Twitter Marketing, LinkedIn Marketing, Google Plus Marketing, Instagram, Social Media Analytical Tools, Scheduling Posts, Social media Events, Reply and Message. Social media Ad Campaigns & Components, Youtube Marketing: Channel Links, Channel Keywords, Branding Watermark, Uploading Videos, Featured Contents on Channel

Recommended Books:

1. Ian Dodson, “The Art of Digital Marketing ”, Wiley, 2018
2. Seema Gupta, "Digital Marketing" Mc-Graw Hill, 1st Edition, 2017
3. **References:** Puneet Singh Bhatia, "Fundamentals of Digital Marketing", Pearson, 1st Edition, 2017
4. Vandana Ahuja, "Digital Marketing", Oxford University Press
5. Philip Kotler, "Marketing 4.0: – Moving from Traditional to Digital", Wiley, 2017

डॉ. अरुण कुमार पाण्डेय
उपकुलसचिव
प्रभारी अकादमिक प्रथम