

S. S. JAIN SUBODH P.G. COLLEGE
(AUTOMOMOUS)

(Affiliated to University of Rajasthan, Jaipur)

DEPARTMENT OF COMPUTER SCIENCE



NATIONAL EDUCATION POLICY-2020

SYLLABUS OF UG-BACHELOR OF COMPUTER APPLICATIONS (BCA)

Syllabus for 1st to 8th Semesters

SYLLABUS 2026

BCA Part First (I & II Sem) -2026-27

BCA Part Second (III & IV Sem)-2027-28

BCA Part Third (V & VI Sem) -2028-29

BCA Part Fourth (VII & VIII Sem) -2026-27 onwards

Preamble

Computer Applications has emerged as one of the most influential and rapidly advancing disciplines in the modern era of science and technology. As the world becomes increasingly driven by digital transformation, the role of Computer Applications in shaping innovation, communication, business processes, research, and everyday life has grown immensely. The discipline combines strong theoretical foundations with practical and industry-oriented applications, enabling learners to develop analytical thinking, computational skills, and problem-solving abilities essential for addressing complex real-world challenges.

Today, computers and information technology form the backbone of almost every sector, including business, healthcare, education, finance, governance, engineering, media, and scientific research. This widespread integration of technology has created an ever-growing demand for highly skilled professionals capable of designing, developing, managing, and securing digital systems and solutions. The field of Computer Applications offers diverse and emerging specializations such as Software Development, Artificial Intelligence, Data Science, Machine Learning, Cyber Security, Cloud Computing, Networking, Web and Mobile Technologies, and Data Analytics.

The Bachelor of Computer Applications (BCA) and BCA (Honours) programmes are carefully designed to provide students with a comprehensive understanding of Computer Science, programming methodologies, and modern computing technologies. The programmes focus on blending academic excellence with practical exposure through laboratory work, projects, internships, and industry-oriented learning experiences. Emphasis is placed on developing technical proficiency, innovation, creativity, teamwork, communication skills, and ethical values to prepare students for the highly competitive and continuously evolving global IT environment.

The curriculum is structured to nurture professional competence and adaptability while encouraging students to explore emerging technologies and research-oriented approaches. By integrating conceptual learning with hands-on implementation, the programmes empower students to become competent professionals, entrepreneurs, researchers, and lifelong learners capable of contributing effectively to society and the digital economy.

Graduates of the programme are well-equipped to pursue higher education, research, entrepreneurship, and rewarding careers in areas such as Software Engineering, Web and Mobile Application Development, Artificial Intelligence, Data Analytics, Cyber Security, Cloud Computing, Network Administration, and IT Consultancy. The programme also prepares students to adapt confidently to future technological advancements and global industry requirements.

The Programme Aims to:

- Build a strong academic foundation in Computer Science, Information Technology, and emerging technologies.
- Develop advanced programming, analytical, computational, and problem-solving abilities among students.
- Equip learners with the technical expertise required to design innovative and efficient real-world solutions.
- Foster creativity, research orientation, professional ethics, leadership qualities, and lifelong learning skills.

- Prepare students for higher education, research, entrepreneurship, and successful careers in the rapidly evolving global IT industry.

Eligibility for Admission:

Admission procedure for I semester starts in the month of June/July every year. The admission of the BCA course should be made irrespective of the stream (Arts/Science/Commerce). A candidate must have passed the 10+2 level examination of any recognized board with 50% or more marks (45% for SC/ST/OBC/SOBC category) in aggregate without any approximation.

Examination Scheme:

Each **Theory paper** shall be of 100 marks (70 marks for written examination of 3 hours duration and 30 marks for internal assessment).

Each theory paper shall contain three parts:

Part A

Will contain 12 questions (student will attempt any 10) of very short answer type questions carrying 1 mark each.

Part B

Will contain 4 short descriptive type questions (one from each unit) carrying 5 marks each. All questions are compulsory.

Part C

Will contain 4 long descriptive type questions (one from each unit) carrying 10 marks each. All questions are compulsory with internal choice.

Theory Examination Pattern

Part	Particulars	Marks Distribution	Total Marks
Part A	12 Very Short Questions (Attempt any 10)	10 × 1	10
Part B	4 Short Questions (One from each Unit, Compulsory)	4 × 5	20
Part C	4 Long Questions from each Unit with Internal Choice	4 × 10	40

Each **practical examination** (Maximum Marks: 100) will be of 2-hours duration and shall carry 60 marks for assigned exercise(s), practical file, and viva-voce examination, and 40 marks for internal assessment.

Passing Criteria, Promotion Rules and Examination Regulations

The following rules and regulations shall govern the promotion of students to higher semesters/classes and matters related to examinations for the BCA programme:

Promotion Rules

- Promotion from odd semesters to even semesters shall be automatic, provided the student has registered for the examination by submitting the prescribed examination form and fee.
- For promotion to the next academic year (III/V Semester), a student must pass in at least 50% of the total subjects offered in the combined semesters of the previous academic year.

Minimum Passing Criteria

- A student must secure a minimum of 40% marks in each subject, considering theory examination and internal assessment together.
- In subjects having practical components, a student must obtain a minimum of 40% marks in the practical examination separately to be declared passed in that subject.
- If a student fails either in theory or practical examination, he/she shall be required to reappear only in the respective failed component.
- If a student does not secure the minimum passing marks in the theory examination but clears the practical examination, the practical marks shall be carried forward as per university/ Institution rules.

Examination System

- Semester examinations for odd and even semesters shall be conducted at the end of each respective semester.
- There shall be no provision for supplementary examinations under the semester scheme.
- Students are required to appear in both internal examinations conducted during each semester. No repeat or re-conduct of internal examinations shall be permitted under any circumstances.

Maximum Duration

- A candidate must complete the programme within a maximum period equal to twice the prescribed duration of the course, calculated from the year of first appearance in the main examination.

General Regulations

- A student admitted to the autonomous programme shall not be permitted to pursue any other regular academic programme simultaneously from any other university/institution.
- Students enrolled in the programme shall also not be permitted to undertake any full-time employment during the course of study.

Attendance Requirements

- Every student shall be required to maintain a minimum of 75% attendance separately in theory lectures and practical classes for each subject/paper.
- Students failing to meet the prescribed attendance requirement may not be permitted to appear in the semester examinations as per institutional rules.

Award of Division and Grade:

To award Grade and Grade Points to students the total of two internal assessment & End term semester exam marks in all subjects will be considered. Over all merit / rank of student will be declared after compilation of marks / grade secured by the candidate in all the semesters of that course. The distribution is as follows.

% of Marks	Grade Point	Grade
90.00% and above	10	Outstanding
80.00-89.99%	9	A+
70.00-79.99%	8	A
60.00-69.99%	7	B+
50.01-59.99%	6	B
45.01-49.99%	5	C
40.00-44.99%	4	P
Less than 40%	0	F

Note: Rules for result computation, passing criteria, and the grading system are as per the examination policy of S. S. Jain Subodh P.G. College (Autonomous), Jaipur.

Programme Objectives

The Bachelor of Computer Applications (BCA) programme is thoughtfully designed to nurture highly skilled, innovative, and industry-ready professionals capable of meeting the dynamic demands of the rapidly evolving digital world. The programme emphasizes academic excellence, technical competence, practical exposure, and holistic personality development to prepare students for leadership roles in the field of Information Technology and Computer Science.

The programme aims to achieve the following objectives:

- To build a strong foundation in Computer Science, software development, programming methodologies, and emerging digital technologies.
- To prepare students for successful and rewarding careers in the IT industry through a balanced blend of theoretical knowledge and practical learning.
- To bridge the gap between academia and industry by offering a modern, skill-oriented, and industry-relevant curriculum aligned with current technological trends.
- To provide exposure and specialization opportunities in advanced domains such as Artificial Intelligence, Data Science, Cyber Security, Cloud Computing, Machine Learning, Networking, and Web & Mobile Technologies.
- To develop analytical thinking, logical reasoning, computational skills, and problem-solving abilities required to address complex real-world challenges effectively.
- To impart knowledge of modern tools, technologies, programming frameworks, and information processing techniques widely used in the software and technology industry.
- To encourage innovation, creativity, research aptitude, entrepreneurship, and a spirit of continuous learning among students.
- To enable students to design, develop, and implement efficient, secure, and scalable computing solutions with confidence and professional competence.
- To cultivate leadership qualities, teamwork, communication skills, ethical values, and social responsibility for holistic professional development.
- To prepare students for higher education, research, competitive examinations, and advanced studies in Computer Science, Information Technology, and allied disciplines.
- To empower learners with adaptability and lifelong learning skills necessary to excel in the ever-changing global technological environment.

Programme Outcomes (POs): BCA (3/4-Year Degree)

Upon completion of the three-year BCA program, students will possess the following capabilities:

PSO1: Demonstrate the ability to apply theoretical knowledge to various fields.

PSO2: Develop language proficiency to meet the demands of corporate communication.

PSO3: Prepare students in various technology disciplines, including computer applications, computer networking, software engineering, JAVA, database concepts, and programming.

PSO4: Introduce the concept of project development using the technologies learned during the semester to enhance programming skills in young IT professionals.

PSO5: Improve logical ability and programming concepts through practical implementation in the programming lab.

PSO6: Prepare students for the future by fostering creativity, social awareness, and general knowledge.

PSO7: Encourage students to transform their start-up ideas into reality through implementation.

PSO8: Ability to understand changes and future trends in the field of computer applications.

PSO9: Identify, formulate, analyze, and solve programming problems using different programming languages.

Scheme of Examinations & Syllabus
w. e. f. session 2026-27

BCA-I Semester						
Sem	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
I	PBCA101	Programming in C	CC(Major)	Theory	4	4
	PBCA102	Web Application Development	CC(Major)	Theory	4	4
	PBCA103	Computer Fundamentals & Office Management Tools	CC(Major/Minor)	Theory	4	4
	PBCA151	Programming in C Lab	CC(Major)	Practical	4	2
	PBCA152	Web Application Development Lab	CC(Major)	Practical	4	2
	PBCA153	Office Management Tools Lab	CC(Major/Minor)	Practical	4	2
		General English	AEC	Theory	2	2
		SEC1 (Mathematics for Computing)	SEC	Theory	2	2
		VAC1 (Choice Based)	VAC	Theory	2	2
			Total		30	24
II	PBCA201	Operating Systems	CC(Major)	Theory	4	4
	PBCA202	Database Management System	CC(Major)	Theory	4	4
	PBCA203	Computer Organization & Architecture	CC(Major)	Theory	6	6
	PBCA251	Operating Systems Lab	CC(Major)	Practical	4	2
	PBCA252	DBMS Lab	CC(Major/Minor)	Practical	4	2
		General Hindi	AEC	Theory	2	2
		SEC2 (Effective Communication Skills)	SEC	Theory	2	2
		VAC2 (Choice Based)	VAC	Theory	2	2
			Total		28	24

Sem	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
III	PBCA301	DataStructuresand Algorithms	CC(Major)	Theory	4	4
	PBCA302	ObjectOrientedProgram ming Through C++	CC(Major)	Theory	4	4
	PBCA303	SoftwareEngineering	CC(Major/Minor)	Theory	6	6
	PBCA351	DataStructuresLabUsin g C	CC(Major)	Practical	4	2
	PBCA352	OOPsLab	CC(Major)	Practical	4	2
		Introduction to Cyber Security	SEC	Theory	2	2
		VAC3 (Choice Based)	VAC	Theory	2	2
		MDC (Choice Based)	MDC	Theory	4	4
			Total		30	26
IV	PBCA401	PHP Programming	CC(Major)	Theory	4	4
	PBCA402	ObjectOrientedConcept s Using Java Programming	CC(Major)	Theory	4	4
	PBCA403	Mathematics&Statistics	CC(Major/Minor)	Theory	6	6
	PBCA451	PHPLab	CC(Major)	Practical	4	2
	PBCA452	JavaLab	CC(Major)	Practical	4	2
		Environmental Management	SEC	Theory	2	2
		VAC4 (Choice Based)	VAC	Theory	2	2
		MDC (Choice Based)	MDC	Theory	4	4
			Total		30	26

Sem	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
V	PBCA501	ArtificialIntelligence& Machine Learning	CC(Major)	Theory	4	4
	PBCA502	Python Programming	CC(Major)	Theory	4	4
	PBCA503	DataCommunication& Computer Networks	CC(Major/Minor)	Theory	6	6
	PBCA551	MachineLearningLab	CC(Major)	Practical	4	2
	PBCA551	Python Lab	CC(Major)	Practical	4	2
		VAC5 (Choice Based)	VAC	Theory	2	2
			Total		24	20
VI	PBCA601	.NETFrameworkwithC#	CC(Major)	Theory	4	4
	PBCA602	IntroductiontoDataScience	CC(Major)	Theory	4	2
	PBCA603	Cloud Computing	CC(Major/Minor)	Theory	6	6
	PBCA651	.NETwithC# Lab	CC(Major)	Practical	4	2
	PBCA652	Industrial Project	CC(Major)	Practical	4	2
		VAC6 (Choice Based)	VAC	Theory	2	2
			Total		24	18

Sem	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
VII	PBCA701	Data Visualization and Interpretation	CC	Theory	4	4
	PBCA702	Digital Image Processing	CC	Theory	4	4
	PBCA703	Adv. Java Programming	CC	Theory	4	4
	PBCA704	Data Warehousing & Data Mining	CC	Theory	4	4
	PBCA705	Deep Learning	CC	Theory	4	4
	PBCA751	Data Visualization and Interpretation Lab	CC	Practical	4	2
	PBCA752	Digital Image Processing Lab	CC	Practical	4	2
	PBCA753	Adv. Java Lab	CC	Practical	4	2
		Total Credits of VII Sem			32	26
VIII	PBCA801	Big Data Analytics	CC	Theory	4	4
	PBCA802	IOT	CC	Theory	4	4
	PBCA803	Network Security and Cryptography	CC	Theory	4	4
	PBCA804	Computer Graphics	CC	Theory	4	4
	PBCA851	Big Data Lab	CC	Practical	4	2
	PBCA852	IOT Lab	CC	Practical	4	2
	PBCA853	Network Security and Cryptography Lab	CC	Practical	4	2
		Total Credits of VIII Sem			28	22

Semester:I

CorseCode: PBCA101	CourseTitle: Programmingin C
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts and implementation techniques of programming languages.
- Develop the ability to learn and adapt to new programming languages efficiently.
- Gain an understanding of the concepts and features of functional programming languages.
- Acquire the skills necessary to design and write simple programs in various programming languages.

CourseOutcomes(COs)

After successfully completing this course, students will be able to:

- Understand the basic programming concepts, structure, and syntax of the C programming language.
- Design algorithms and develop solutions for simple computational problems.
- Read, analyze, and trace the execution of C programs.
- Write, compile, debug, and execute C programs for a variety of applications.
- Perform input and output operations effectively using C programs.
- Implement programs involving arrays and other fundamental data structures.
- Use Integrated Development Environments (IDEs) and programming tools for efficient program development.

PBCA101: Programming in C

Unit-I

Fundamentals of C: Programming Concepts, Pseudocode, Algorithm, Flowchart.

Introduction: History and importance of C, Basic Structure and Execution of C Program, Constants, Variables, and Data Types, Operators and Expressions, Operator Precedence and Associativity, Managing Input and Output Operations,

Decision making and Branching: if statement, if...else statement, nesting of if else statement, else if ladder, switch statement, Goto statement.

Unit-II

Iteration: while, do...while, for loop, nested loops, break & continue.

Arrays and Strings: One-dimensional arrays, Declaration and Initialization, Two-dimensional arrays, Declaration and Initialization, Character arrays.

String: Reading and Writing strings, String-handling functions – strlen, strcpy, strcat, strcmp.

Unit-III

Functions: Need, Elements of user-defined functions, Definition of Functions, Function call and declaration, Category of Functions, Parameter Passing, Recursion, Passing arrays to functions, Passing strings to functions.

Storage Classes: Scope, visibility and life time of variables.

Understanding Pointers: Accessing the address of a variable, declaration and initialization of pointer variables, accessing a variable through its pointer, Array through pointers.

Unit-IV

Structures and Unions: Defining structure, Declaring structure variables, Accessing structure members, Structure initialization, Operation on individual members, Array of structures, Union.

File Management in C: Introduction of File handling, Need of File Handling, Defining, Opening and Closing a File, Input/Output Operations on Files.

ReferenceBooks:

- BalagurusamyE;ProgramminginANSIC;FifthEdn;McGraw Hill,2011.
- KanetkarY.;LETUS C;XEdition, BPB,2010.
- DeitelHM&DeitelJP;CHowtoprogram;5thEdn;PearsonPub
- GottfriedB;ProgrammingwithC:SchaumQutlines;McGrawHillEdition.

CorseCode: PBCA102	CourseTitle: WebApplication Development
CourseCredit: 04	Hours/Week: 04

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandthe fundamentalconceptsandtechnologiesusedinwebdevelopment.
- Identifyandapply appropriatetechnologiestoaddressweb client-serverchallenges.
- Designanddevelopreal-time,interactivewebapplications.
- UseJavaScriptto create dynamicwebpagesandperformclient-sideformvalidation.
- Analyze and select suitable client-side and server-side technologies for various webapplications.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Applyappropriatewebsitechnologiestosolveclient-servercomputingproblems.
- Analyze,design,anddevelopreal-timewebapplications.
- Implementdynamicweb pagefeaturesusingJavaScript.
- Validateuserinputefficientlythroughclient-side scripting.
- Selectandintegratesuitablefront-endandback-endtechnologiesforweb-basedsolutions.
- AnalyzetoUseappropriateclient-sideandServer-sideapplicationtechnology

PBCA102: WebApplicationDevelopment

Unit – I

Basic of internet, file transfer, telnet, usenet. Introduction to Internet Protocols-HTTP, FTP, SMTP protocols.

World Wide Web: Elements of the Web, Web browser and its architecture, the web server, the proxy server, Microsoft internet explorer, viewing pages with a browser, using a browser for Mail, News and chat, Security and Privacy issues (cookies, firewalls, Data Security, executable Applets and scripts, blocking system).

Unit– II

HTMLFundamentals: IntroductiontoHTML,HTMLElements,HTMLSemantics,HTML 5 Doc Types, New Structure Tags, Section, Nav, Article, Aside, Header, Footer, HTML Attributes, Headings, Paragraphs, Styles, Quotations, Blocks, Classes, Layout, Iframes, Creating HTML Pages, incorporating Horizontal Rules and Graphical Elements, Hyper-links, Creating HTML Tables, Creating HTML Forms, HTML and Image Techniques, HTML and Page,DevelopmentofWebsiteandWebpage(Planning,NavigationandThemes,Elements of a Web page, steps of creating a site, publishing and publicizing site structuring web site.

Unit–III

CascadingStyleSheets: Understanding StyleSheets, CSS Syntax and Applying StyleSheets to HTML document, Developing StyleSheets: inline, internal and external. CSS Selectors, <DIV> tag, Using class and ID, Styling Backgrounds, Styling borders, Styling Text, Styling Fonts, Styling Links, Styling Lists, Styling Tables, Margin, Flex and Grids. Bootstrap & Web page design: CMS, Banks of CMS, Joomla/WordPress-Installation, Design and development of websites.

Unit–IV

Java script: Introduction to scripting language, Client-Side Scripting, memory concepts, arithmetic decision making. Java script control structures, Java script functions, JS Popup Boxes, events, program modules in java script, function definitions duration of identifiers, scope rules, Controlling Programming Flow, recursion java script global functions. Arrays handling in Java script, The Java Script Object Model, Developing Interactive Forms, Validation of Forms, Cookies and Java Script Security Controlling Frames in Java Script, Client – Side Java Script Custom.

ReferenceBooks:

- The Complete Reference: HTML & XHTML; Thomas A. Powell, 4th Edn.
- Mastering HTML 4.0 by Deborah S. Rayan Eric J. Ray From BPB
- Mastering JavaScript, BPB publication.
- Internet and web technology by Raj Kamal, TMH Publication Steven Holzner,
- The Complete Reference JavaScripts, Tata McGraw– Hill, 3rd Edn.
- JavaScript, Don Gosselin, Vikas publications

CorseCode: PBCA103	CourseTitle: Computer Fundamentals& Office Management Tools
CourseCredit:04	Hours/Week: 04

Course Objectives(COs)

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

- Understand the fundamental concepts of computing, including computer hardware, software, operating systems, and networking.
- Develop proficiency in the effective use of office productivity tools such as word processors, spreadsheets, presentation software, and database management systems.
- Apply information technology concepts and tools to solve business problems and enhance organizational productivity.
- Acquire practical skills in internet usage, electronic communication, and multimedia applications for professional and academic purposes.

Course Outcomes(COs)

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

- Demonstrate an understanding of computer fundamentals, including the classification, architecture, and components of computers.
- Use internet services effectively, including web browsing, email communication, online search, and domain-related services.
- Perform office automation tasks using word processing software.
- Create, analyze, and manage data using spreadsheet applications.
- Design effective presentations using presentation software.
- Develop and manage simple databases using database management software.
- Perform basic multimedia editing involving images, audio, and video.
- Configure and manage operating system settings and essential system utilities.

PBCA103: Computer Fundamentals& Office Management Tools

UNIT- I

Introduction to Computers: Characteristics of computers, Evolution of computers, generation of computers, Block diagram of computer & role of each block, classification of computers. Input and Output Devices

Primary and Secondary Memory: Memory hierarchy, Random access memory (RAM), types of RAM, Read only memory (ROM), types of ROM. Classification of secondary storage devices, magnetic tape, magnetic disk, optical disk.

Number Systems: Introduction to number system, Binary, Octal, Hexadecimal, conversion between number bases, Arithmetic operations on binary numbers, Alphanumeric- BCD, ASCII.

UNIT-II

Computer Software: software categories, system software, application software, utility software. Classification of system software, Computer Languages:

Introduction, classification of programming languages, generations of programming languages, features of a good programming language.

Internet Basics: Internet applications, Services of Internet, Logical and Physical addresses, Internet Service Providers, Domain Name System.

UNIT-III

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.

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, Autosum, copying formulas, absolute and relative addressing, formatting of worksheet, previewing & printing worksheet, Graphs and Charts, Database, macros, multiple worksheets-concepts.

UNIT-IV

PowerPoint: Creating and viewing 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.

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:

- Sanjay Saxena; A First Course in Computers 2003 Edition; Vikas Pub.
- Computer Fundamentals by P.K. Sinha, BPB Publication.
- Computer Fundamentals and Programming in C, Reema Thareja, OXFORD University Press.
- Microsoft; 2007/2010 Microsoft Office System; PHI.
- Microsoft; Microsoft Office 2007/2010: Plain & Simple; PHI.
- MS-Office, Dr. S.S. Shrivastava, Published by Laxmi Publication.
- Office 2021: In Easy Steps, Michal, BPB Publication.

CorseCode: PBCA151	CourseTitle: Programming in C Lab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

Part A

1. Write a program to calculate the area and circumference of a circle for a given radius.
2. Write a program to read three numbers and determine the largest among them.
3. Write a program to demonstrate the use of library functions available in math.h
4. Write a program to check whether a given number is prime.
5. Write a program to generate the first n prime numbers.
6. Write a program to find the sum of digits of a number, reverse the number, and check whether it is a palindrome.
7. Write a program to continuously accept numbers until the user enters 999, and calculate the sum of all positive numbers entered.
8. Write a program to read the percentage of marks and display the appropriate grade using an else-if ladder.
9. Write a program to read marks of n students and calculate the average using a one-dimensional array.
10. Write a program to remove duplicate elements from a one-dimensional array.
11. Write a program to perform addition and subtraction of two matrices.

Part B

1. Write a program to find the length of a string without using any built-in string function.
2. Write a program to demonstrate commonly used string handling functions.
3. Write a program to demonstrate the use of pointers in C.
4. Write a program to check whether a number is prime using a user-defined function isPrime().
5. Write a program to read, display, and find the trace of a square matrix.
6. Write a program to read, display, and add two matrices.
7. Write a program to count alphabets, digits, vowels, consonants, spaces, and special characters in a given string.
8. Write a program to reverse a string using pointers.
9. Write a program to swap two numbers using pointers.
10. Write a program using structures to read and display records of n students.
11. Write a program to demonstrate the difference between a structure and a union.

CorseCode: PBCA152	CourseTitle: WebApplicationDevelopmentLab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Create web pages demonstrating basic HTML elements, attributes, formatting tags, and hyperlinks.
2. Design web pages using images, tables, and various form elements.
3. Develop multimedia web pages incorporating HTML5 audio, video, and advanced input types.
4. Implement CSS syntax and attribute selectors for styling web pages.
5. Apply CSS properties such as fonts, backgrounds, colors, links, and lists.
6. Demonstrate the CSS Box Model, display properties, opacity, float, and clear.
7. Design responsive layouts and navigation bars using CSS.
8. Create visually appealing web pages using rounded corners, border images, transitions, and animations.
9. Write programs demonstrating output methods, variable declarations, operators, arithmetic operations, data types, and assignments.
10. Implement JavaScript functions, Boolean logic, comparison operators, and conditional statements.
11. Develop programs using switch statements, loops, break statements, and type checking.
12. Create and manipulate JavaScript objects, and demonstrate variable scope.
13. Perform string operations using various string methods.
14. Implement numerical computations using number methods, mathematical functions, and date manipulation.
15. Handle events and perform form validation using JavaScript and Web APIs.
16. Develop interactive web applications using the Document Object Model (DOM), Browser Object Model (BOM), and advanced validation techniques.

CorseCode: PBCA153	CourseTitle: OfficeManagementToolsLab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

Students shall perform practical exercises based on the following office automation tools:

1. Word Processing (MS Word equivalent)
 - Create, edit, format, and print professional documents.
 - Use tables, mail merge, templates, headers, footers, and page formatting features.
2. Spreadsheet (MS Excel equivalent)
 - Perform calculations using formulas and functions.
 - Create charts, sort and filter data, and generate reports using worksheets.
3. Presentation Software (MS PowerPoint equivalent)
 - Design professional presentations using themes, animations, transitions, and multimedia elements.
4. Database Management (MS Access equivalent)
 - Create databases, tables, forms, queries, and reports for managing structured data efficiently.

Semester:II

CorseCode: PBCA201	CourseTitle: OperatingSystems
CourseCredit: 04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand and design key operating system components, including system calls, process schedulers, memory management, virtual memory, and paging systems.
- Evaluate and compare operating system components using appropriate performance analysis techniques and tools.
- Analyze device management and resource allocation techniques in time-sharing and distributed systems.
- Develop and analyze simple concurrent programs using concepts such as message passing, synchronization, and transactional memory.

CourseOutcomes(COs)

After completing this course, students will be able to:

- Explain fundamental operating system concepts, including processes, threads, files, semaphores, inter-process communication, and shared memory.
- Analyze and apply important operating system algorithms, such as process scheduling and memory management algorithms.
- Classify and evaluate various resource management, deadlock handling, and memory management techniques.
- Perform basic system administration and command-line operations in a Linux environment.

PBCA201:OperatingSystems

Unit – I

Concepts: Operating System & its need, functions of OS, Types of OS: Simple Batch Systems, Multi-programmed Batched Systems, Time-Sharing Systems, Parallel Systems, Distributed Systems and Real-Time Systems.

Operating-System Structures: System Components, Operating System Services, System Calls, System Structure, Virtual Machines, Process Management.

Unit– II

CPU Scheduling Algorithms: Basic Concepts, Scheduling Criteria, FCFS, SJF, Priority, Round-Robin, Multilevel Queue, Multilevel Feedback Queue, Multiple-Processor Scheduling.

Process Synchronization & Deadlocks: The Critical section problem,

synchronization

semaphores, Classical problems of synchronization, System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

Unit-III

Memory Management: Functions, Logical versus Physical Address space, Swapping, Contiguous allocation (fragmentation), Paging, Segmentation. Virtual Memory, Demand Paging, Page-replacement Algorithms (FIFO, Optimal, LRU).

File Management: File Concepts (Operations & Attributes), Access Methods, Directory Structure, File System Structure, Allocation Methods (Contiguous Allocation, Linked Allocation, Indexed Allocation).

Device Management: General device characteristics, device controllers, device drivers, Interrupt Driven I/O, Memory Mapped I/O, Direct Memory.

Unit-IV

Introduction to Linux, Evolution of Linux, Linux Architecture, Linux file system (inode, Super block, Mounting and Unmounting), Essential Linux Commands and Shell Scripts (Internal and External Commands), Kernel, Process Management in Linux.

Reference Books:

- A. Silberschatz and P. Galvin, "Operating System Concepts", Addison-Wesley, 5th Ed., 2001.
- Gary Nutt: Operating Systems - A Modern Perspective (Second Edition), Pearson Education, 2000.
- Tanenbaum A. S., Modern Operating Systems, PHI Publ.
- Peterson Richard, "The Complete Reference Linux" Tata McGraw Hill.
- Simitabha Das, "Unix/Linux Concepts & Applications". Tata McGraw Hill
- Achyut S. Godbole: Operating Systems, Tata McGraw Hill Publishing Company Limited, 2000.
- Harvey M. Deitel, Operating Systems, Pearson Education, 2001.

CorseCode: PBCA202	CourseTitle: DatabaseManagementSystems
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts, architecture, and components of MySQL database systems.
- Explore various database models, including relational, document-oriented, key-value, column-oriented, and graph databases.
- Design database objects, load and manipulate data, and optimize database performance.
- Develop the ability to manage and query large volumes of data efficiently using SQL.

CourseOutcomes(COs)

After completing this course, students will be able to:

- Demonstrate a clear understanding of database design, management, and related terminology.
- Analyze and compare different database models and their applications.
- Evaluate database schemas for normalization and eliminate design anomalies.
- Design, implement, and manage relational databases using MySQL.
- Write efficient SQL queries for data retrieval, manipulation, and administration.

PBCA202:DatabaseManagement Systems

UNIT- I

Database System Concepts & Architecture: Overview of DBMS, Basic DBMS terminology, database system v/s file system, Advantages and dis-advantages of DBMS, Codd's rules, data independence. Architecture of a DBMS, Schemas, Instances, Database Languages, Database Administrator, Data Models.

UNIT-II

Data Modeling: Data modeling using the Entity Relationship Model: ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation.

Relational Model: Concepts, Constraints, Languages, Relational database design by ER & EER mapping, Relational algebra relational calculus. Relational Algebra, Fundamental operations of Relational Algebra.

UNIT-III

Database Design: Functional dependencies, loss less decomposition, Normalization: 1-NF, 2-NF, 3-NF and BCNF. Transaction Management: Transactions: Concepts, ACID Properties, States of Transaction, Serializability, Isolation, Checkpoints, Deadlock Handling.

Recovery System & Security: Failure Classifications, Recovery & Atomicity, Log Base Recovery, Recovery with Concurrent Transactions, Introduction to Security & Authorization.

UNIT-IV

Introduction to SQL: Characteristics of SQL, Advantages of SQL, SQL data types and literals, Types of SQL commands, SQL operators and their procedure, Tables, views and indexes, Queries and sub queries, Aggregate functions, insert, update and delete operations, Joins, Unions, Intersection, Minus in SQL.

ReferenceBooks:

- KorthHFandSilberschatazA, SystemConcepts, SixthEdition; McGraw Hill, 2010
- Leon, andLeon, SQL TataMcGrawHillPub.Co. Ltd.
- IvanBayross; SQL/PL4thEdn: BPB, 2009
- Navathe S.B. Elmasri R; Fundamentals of Database Systems, Fifth Edition, Pearson 2011.
- Ramakrishnan and Gharke, Database Management Systems, 3rdEd, Tata McGraw Hill, 2007.
- SinghS.K.; DatabaseSystems; I Edition; Pearson, 2006.

CorseCode: PBCA203	CourseTitle: ComputerOrganization&Architectu re
CourseCredit:06	Hours/Week: 06

CourseObjectives(COs)

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

- Understand the fundamentals of digital logic design, including logic gates, combinational circuits, and sequential circuits.
- Design and analyze sequential and arithmetic circuits used in computer systems.
- Comprehend the basic architecture and organization of computer systems, including CPU, memory, and input/output devices.
- Understand the structure, operation, and hierarchy of memory systems.

CourseOutcomes(COs)

After completing this course, students will be able to:

- Apply Boolean algebra and data representation techniques in digital system design.
- Design and analyze combinational, sequential, and arithmetic circuits.
- Explain the organization and operation of microprocessors and computer architecture.
- Describe the functions and working of CPU and input/output devices.
- Analyze the structure, hierarchy, and operations of memory systems.
- Understand the concepts of parallel processing, pipelining, and their significance in modern computer systems.

PBCA203: Computer Organization & Architecture

UNIT- I

Boolean Algebra and Logic Gates: Logic Gates, Basic laws of Boolean algebra, Simplification of Boolean algebra.

Combinatorial Logic: Multiplexers, Decoders, Encoders, Adder & Subtractors, Parallel Binary Adder, Parallel binary Subtractor.

UNIT-II

Sequential Logic: Sequential circuits: Flip-flops, S-R, D, J-K, T, Clocked Flip-flop, Race around condition, Master slave Flip-Flop.

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.

UNIT-III

Basic Computer Organization and Design: Instruction Codes, Computer Registers;

Common bus system; Computer Instructions; Instruction formats; Instruction Cycle; Register reference instructions, Addressing Modes.

CPU Design: Specifying a CPU, design and implementation of a simple CPU (fetching instructions from memory, decoding and executing instructions).

UNIT-IV

Input-Output Organization: Input-output Interfaces, Asynchronous Data Transfer, Mode of Transfer - Programmed I/O, Interrupt I/O, Direct Memory access (DMA).

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory, I/O Interrupt, types of Interrupts.

Reference Books

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

CorseCode: PBCA251	CourseTitle: OperatingSystemsLab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Perform installation, configuration, and basic system settings in a Linux environment.
2. Practice directory navigation and file system management using Linux commands.
3. Execute commonly used Linux commands for file handling, process management, and system monitoring.
4. Implement pattern matching and text processing using commands such as grep, sed, and awk.
5. Create, edit, and manage files using text editors such as vi and nano.
6. Develop and execute shell scripts to automate simple tasks and demonstrate practical applications.

CorseCode: PBCA252	CourseTitle: DBMS Lab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Analyze a real-world organization and identify its entities, attributes, and relationships.
2. Determine primary keys, candidate keys, and partial keys for the identified entities.
3. Establish relationships among entities and define appropriate cardinalities. Identify strong and weak entities wherever applicable.
4. Represent entities and relationships using Entity-Relationship (ER) diagrams and corresponding relational tables.
5. Practice Data Definition Language (DDL) commands such as CREATE, ALTER, DROP, TRUNCATE, and RENAME.
6. Create and manage databases and tables, including modification and deletion operations.
7. Practice Data Manipulation Language (DML) commands on the created database.
8. Perform data operations using commands such as SELECT, INSERT, UPDATE, and DELETE.
9. Write SQL queries using subqueries and operators such as ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, and INTERSECT, along with integrity constraints.
10. Implement queries using aggregate functions such as COUNT, SUM, AVG, MAX, and MIN, along with GROUP BY, HAVING, and the creation and deletion of views.

Semester:III

CorseCode: PBCA301	CourseTitle: DataStructuresandAlgorithms
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Designefficientalgorithmsusingvariousalgorithmdesignstrategies.
- Analyzecomputationalproblemsanddevelopappropriatealgorithmicsolutions.
- Classifyproblemsandselectsuitablealgorithmdesigntechniquesforsolvingthem.
- Evaluatealgorithmsintermsof timeandspacecomplexityusingasymptoticnotations.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Understandthefundamentalconceptsandsignificanceofdatastructuresincomputing.
- Implementandapplycommondatastructuresuchasarrays,linkedlists,stacks, queues, trees, and graphs.
- Analyzethetimeandspacecomplexityofalgorithmsassociatedwithvariousdata structures.
- Utilize appropriate data structures and algorithms to solve real-world computationalproblems efficiently.

PBCA301:DataStructuresandAlgorithms

UNIT – I

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

Linear Structure Array: Introduction, traversal, insertion, deletion operation on array.

Searchingandsorting: Searching- sequentialsearching,binarysearching,hashing.Sorting- selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort, efficiency considerations.

UNIT–II

Linked List Structure: Linked List, Array v/s Linked List, Linked List representation, operations on linked list – traversal, insertion, deletion types of linked list – singly, doubly, circular linked list.

UNIT – III

Stack: Introduction, implementation of stack as an array and linked list, operation on stack, Polish notations and arithmetic expressions.

Queue: Introduction of queue, implementation of queue as an array and linked list, types of queues, operations on queue.

UNIT – IV

Tree Structure: Concept and terminology, Types of trees, binary search tree, array and linked representation of binary search tree, tree traversals, inserting, deleting and searching into binary search tree.

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, transitive closure.

Referencebooks:

- S.Lioschutz:DataStructures,McGrawHill InternationalEdition.
- A.V.Aho.,J.E.Hopcroft,andJ.D.Ullman,Data StructuresandAlgorithms, Pearson.
- A.MichaelBerman:DataStructuresviaC++,OxfordUniversityPress.
- SaraBaase andAllenVanGelder:ComputerAlgorithms,PearsonEducation Asia.

CorseCode: PBCA302	CourseTitle: ObjectOrientedProgramming Through C++
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts and principles of Object-Oriented Programming (OOP) and distinguish it from procedural programming.
- Learn the syntax and semantics of the Java programming language, including data types, operators, and control structures.
- Design and implement classes and objects to encapsulate data and behavior effectively.
- Apply inheritance and polymorphism to create flexible, reusable, and maintainable software solutions.
- Utilize encapsulation, abstraction, abstract classes, and interfaces in program design.
- Implement robust error handling and exception management techniques in C++ applications.

CourseOutcomes(COs)

After successfully completing this course, students will be able to:

- Demonstrate a clear understanding of object-oriented programming concepts and principles.
- Design and develop C++ programs using classes, objects, inheritance, and encapsulation.
- Apply polymorphism, abstraction, and interfaces to create reusable and extensible applications.
- Implement exception handling mechanisms to develop reliable and robust software.
- Develop efficient, modular, and maintainable C++ applications using advanced programming techniques.

PBCA302: Object Oriented Programming Through C++

UNIT–I

Introduction to Object Oriented Concepts: Evolution of OOP, OOP Paradigm, advantages

of OOP, comparison between functional programming and OOP approach, characteristics of object-oriented language – objects, classes, inheritance, reusability, user defined data types, polymorphism, overloading.

UNIT–II

Introduction to C++: C++ tokens, data types, C++ operators, type conversion, variable

declaration, arrays, statements, expressions, conditional statements, Jumping statements, loops, functions, pointers, structures.

Classes and Objects: Classes, objects, defining member functions, arrays of class objects, pointers and classes, passing objects, constructors, types of constructors, destructors, this pointer, access specifiers, friend functions and friend class, inline functions.

Unit– III

Inheritance: Introduction, Importance of Inheritance, types of inheritance, Constructor and Destructor in derived classes., member access control.

Polymorphism: Functions Overloading, Operator Overloading, early binding polymorphism
with pointers, Unary and Binary Operator Overloading, Overload Assignment Operator, Copy Constructor.

Unit–IV

Virtual Function: Virtual Function, late binding, pure virtual functions, abstract classes, Generic Programming with Templates, Overloaded Function Templates, Multiple Arguments function Template.

File Management: Handling Data files (sequential and random), Opening and closing of files, stream state member functions, Operations on File, Exception Handling.

Recommended Books:

- Deitel HM & Deitel JP; C/C++ How to program; 5th Edn; Pearson Pub.
- Balagurusamy; Object Oriented Programming in C++; 4th Edition TMH.
- Venugopal, Rajkumar; Mastering C++; Tata Mcgrow Hill.
- Kanetkar Y.: LETUS C++; BPB;
- Byron Gottfried; Programming with C; TMH.

CorseCode: PBCA303	CourseTitle: Software Engineering
CourseCredit:06	Hours/Week: 06

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandthefundamentalprinciples,concepts,andpracticesofSoftware Engineering.
- Apply standard software development processes and methodologies to build reliable software systems.
- Performrequirementanalysisanddesignefficientsoftwaresolutions.
- Develop software using appropriate programming languages, tools, and development environments.
- Apply software testing techniques and quality assurance practices to ensure software reliability and performance.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- ExplaincoreconceptsandprinciplesofSoftware Engineering.
- Applyappropriatesoftwaredevelopmentlifecycle(SDLC)modelsandmethodologies.
- Analyzeuserrequirementsanddesignstructuredandefficientsoftwaresystems.
- Implementsoftwaresolutionsusingsuitabletoolsandtechnologies.
- Perform software testing and apply quality assurance techniques to deliver robust and maintainable applications.

PBCA303:SoftwareEngineering

Unit-I

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

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

Unit-II

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, modularity, functional independence, refinement, and design classes), The Design Model (data design elements, architectural design elements, interface design elements, component-level design elements, deployment-level design elements).

Testing Strategies & Tactics: A strategic approach to software testing, Strategic issues,

Software testing fundamentals, Test characteristics, Test Strategies for conventional software are: Unit Testing, Integration testing, Validation Testing, System testing, Black-Box testing, White Box testing.

Unit-IV

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

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

Recommended Books:

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

CorseCode: PBCA351	CourseTitle: DataStructuresLabUsingC
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Implement Linear Search and Binary Search to locate the element 7 in the array {4, 7, 3, 2, 1, 7, 9, 0} and display its first occurrence.
2. Sort the array {5, 3, 1, 6, 0, 2, 4} in ascending order using the Bubble Sort algorithm.
3. Sort the array {75, 8, 1, 16, 48, 3, 7, 0} in descending order using both Insertion Sort and Selection Sort.
4. Sort the array {5, 3, 1, 6, 0, 2, 4} in ascending order using the Quick Sort algorithm.
5. Sort the array {5, 3, 1, 6, 0, 2, 4} in ascending order using the Merge Sort algorithm.
6. Create a singly linked list by inserting the elements {61, 16, 8, 27}. Delete the elements 8, 61, and 27, displaying the list after each insertion and deletion.
7. Implement a linear queue by inserting the elements {61, 16, 8, 27}. Delete three elements and display the queue after each operation.
8. Implement a circular queue using an array by inserting the elements {61, 16, 8, 27}. Delete all elements one by one, displaying the queue after each operation.
9. Create an ordered singly linked list using the elements {61, 16, 8, 27}. Delete the elements 8, 61, and 27, displaying the list after each operation.
10. Add the polynomials $6x^3 + 10x^2 + 5$ and $4x^2 + 2x + 1$ using linked list representation.
11. Implement a stack by pushing the elements 5, 9, 34, 17, 32. Pop three elements and display the popped values.
12. Write a recursive program to find the Greatest Common Divisor (GCD) of the numbers 4, 6, and 8.
13. Implement a circular queue using a linked list. Insert the elements {5, 7, 0, 6, 3, 9} and delete the elements 6, 9, and 5.
14. Create a Binary Search Tree using the elements {18, 15, 40, 50, 30, 17, 41}. Insert 45 and 19, then delete 15, 17, and 41, displaying the tree after each operation.
15. Create a Binary Search Tree using the elements {2, 5, 1, 3, 9, 0, 6} and perform in order, pre order, and post order traversals.
16. Sort the elements {9, 16, 32, 8, 4, 1, 5, 8, 0} using the Heap Sort algorithm.
17. Implement Depth First Search (DFS) traversal for a graph.
18. Implement Breadth First Search (BFS) traversal for a graph.

CorseCode: PBCA352	CourseTitle: OOPs Lab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Develop simple C++ programs to demonstrate the creation and use of classes, objects, and references.
2. Write programs to perform various operations on arrays and strings in C++.
3. Implement applications demonstrating different types of inheritance.
4. Develop programs illustrating function overloading.
5. Write programs to demonstrate operator overloading.
6. Implement applications using virtual functions and runtime polymorphism.
7. Develop generic programs using templates.
8. Write programs to demonstrate exception handling mechanisms.
9. Perform file handling operations, including reading from and writing to files.

Semester:IV

CorseCode: PBCA401	CourseTitle: PHP Programming
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts and syntax of the PHP scripting language.
- Comprehend the architecture and basic structure of web applications.
- Understand the HTTP request-response cycle and the interaction between client-side and server-side scripts.
- Learn the fundamentals of MySQL database design and management.
- Develop dynamic, interactive, and database-driven web applications using PHP and MySQL.
- Apply testing, debugging, and security practices in PHP web application development.
- Create and manage PHP-based websites using Content Management Systems (CMS).

CourseOutcomes(COs)

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

- Demonstrate proficiency in Core PHP as a server-side scripting language.
- Design and develop dynamic, interactive, and responsive web pages and websites.
- Execute PHP scripts on a web server and process user requests effectively.
- Integrate PHP with MySQL to create database-driven web applications.
- Implement secure, efficient, and maintainable PHP-based web solutions.

PBCA401:PHPProgramming

UNIT-I

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

Unit-II

Control Structures: Data types, Operators, PHP variables: local, static and global variables, Comments in PHP, Control Structures, Condition statements, if...Else, Switch,? operator, Loops, While, Break Statement Continue. Do... While, For, foreach, 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, defaults arguments in function, dynamic function calling.

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

UNIT-IV

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

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.

Recommended Books:

- PHP, The Complete Reference, Steven Holzner, TMH
- Beginning PHP 5.3, Matt Doyle, John Wiley & Sons
- Core PHP Programming Leon Atkinson Pearson publishers
- Beginning PHP 5.0 Database Christopher Scollo, Harish, Rawat, Deepak Thomas, Wrox Press

CorseCode: PBCA402	CourseTitle: ObjectOrientedConcepts Using Java Programming
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandand applyobject-orientedprogrammingconceptsusingJava.
- Develop,compile,debug,andexecuteJava applications efficiently.
- ApplyadvancedJavaconceptssuchasinheritance,polymorphism,exceptionhandling, multithreading, and interfaces.
- Designanddevelopbothconsole-basedandGUI-basedapplicationsusingJava.
- Workeffectivelyonindividualandteam-basedsoftwaredevelopmentprojects.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Explainthefeaturesof Javaandthe architectureof the JavaVirtualMachine(JVM).
- Write,compile,andexecuteJavaprogramsusingdatatypes,operators,control structures, and type casting.
- Designclassesandobjects,andapplyinheritanceandpolymorphismtosolve programming problems.
- Developapplicationsusinginterfaces,multithreading,andexception handling.
- CreateGUI-basedand event-drivenapplicationsusingJava.
- ImplementfilehandlingoperationsinJava applications.

PBCA402:ObjectOrientedConceptsUsingJavaProgramming

UNIT-I

Introduction: Object Oriented Concepts, Introduction to Java, Features of Java features, Java virtual machine.

Programming Fundamentals of Java: Java Tokens, Operators, Data types, Control Structures, Array.

UNIT-II

Classes and Objects: Introduction to class and objects, Class definition, Object creation, Method Overloading, Constructors, Constructor Overloading, Inheritance. Static, Final, Abstract Keywords.

Packages and Interfaces: Access Specifiers, Package Creation, Use of packages, Basics of Interfaces, Use of Interfaces.

String Handling: String and StringBuffer class, Class, String methods.

UNIT-III

Exception handling in Java: Exception classes, Exception raising & handling, use of Try, Catch and finally, Throwing Exceptions, User Defined Exceptions.

Threading in Java: Concepts of Multi-threading, Thread Life Cycle, Thread class, Runnable Interface, Thread Management in Java.

UNIT-IV

Applets: Introduction to Applets, Applet life cycle, Creating Applets, AppletViewer
Graphics: Graphics Components, Color, Font, Drawing Objects.

JDBC: JDBC Drivers, Two Tier and Three Tier client server Architecture, Setting up a connection to database, Creating and executing SQL statements

ReferenceBooks:

- Masteringjava2,BPB Publications
- ProgrammingwithJavaAPrimer,E. BalagurusamyTataMcGrawHillCompanies
- JavaProgrammingJohnP.FlyntThomson2nd
- ThecompleterefereceJAVA2,Herbert schildt.TMH
- Arnold,Gosling,“TheJavaProgrammingProfessional2000”,AddisonWesley Publication
- C.Thomaswu,“AnintroductiontooopwithJava”, TMH

CorseCode: PBCA403	CourseTitle: Mathematics&Statistics
CourseCredit:06	Hours/Week: 06

CourseObjectives(COs)

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

- Develop strong logical reasoning and problem-solving skills.
- Gain a thorough understanding of fundamental mathematical concepts and theories.
- Acquire a solid foundation in elementary statistics and data analysis.
- Communicate mathematical and logical ideas effectively in written form.
- Apply mathematical concepts in computational and programming contexts.
- Explore various branches of mathematics, including discrete mathematics, linear algebra, statistics, and their applications in computing.

CourseOutcomes(COs)

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

- Understand and apply the concepts of mathematical reasoning, sets, relations, and functions.
- Construct and evaluate logical arguments and mathematical proofs.
- Apply fundamental counting principles and combinatorial techniques.
- Analyze and solve problems involving matrices and determinants.
- Understand the principles of data collection, classification, and presentation.
- Compute and interpret measures of central tendency and dispersion.
- Analyze relationships between variables using correlation and regression techniques.

PBCA403:Mathematics&Statistics

UNIT – I

Sets: Definition of sets, representation of sets, type of sets, Operations on sets, Subsets, Power set, Universal set, Complement of a set, Union and Intersection of two sets, Venn diagrams, Principles of Inclusion and Exclusion. **Relations:** Cartesian product of sets, Definition of relation, Types of relations- reflexive, symmetric, anti-symmetric, transitive, equivalence.

Functions: Definition, Domain & Range of a functions, one to one and onto functions, Bijective functions, composite functions, inverse of functions.

UNIT–II

Logic and Proofs: Proposition, Conjunction, Disjunction, Negation, Compound proposition, De Morgan's laws, Tautology and Contradiction.

Matrices: Definition and Types of Matrices, Addition, Subtraction and Multiplication of Matrices, Non-commutativity of multiplication of matrices, Scalar Multiplication, Transpose of a Matrix.

Determinant: Determinant of a square matrix (upto 3×3 matrices), properties of determinants, minors, cofactors, expansion of determinants, application of determinants in finding the area of a triangle. Adjoint and Inverse of a matrix, Solution of system of linear equations by Cramer's Rule.

UNIT-III

Statistics: Data collection methods, Data classification, Frequency Distribution, Graphical representation of frequency distribution. Measures of Central Tendency- Mean, Median, Mode, Measures of Dispersion- Mean Deviations, Standard Deviations, Variance

UNIT-IV

Correlation Analysis: Correlation, Types of Correlations, Methods of Studying Correlations, Measure of Karl Pearson's coefficient of correlation, Rank Correlation Coefficient.

Regression Analysis: Regression, Use of regression analysis, Difference between Correlation and Regression Analysis, Regression Lines Equations, Properties of regression lines.

Reference Books:

- C.L.Liu: Elements of Discrete Mathematics, Tata Mc-Graw Hill Publishing Company Ltd., 2000
- Seymour Lipschutz; Discrete Mathematics; TMH
- Kenneth H Rosen; Discrete Mathematics & Its Applications; 6 Edition, MGH
- Richard Johnsonbaugh: Discrete Mathematics, Pearson Education, Asia, 2001
- John Truss: Discrete Mathematics for Computer Scientists, Pearson Education, Asia, 2001.
- Basic Mathematics, R.D. Sharma
- B.L. Agrawal; Basic Statistics; Khanna Pub
- S.P. Gupta; Statistical Methods; Sultan Chand & Sons
- S.C. Gupta, V.K. Kapoor; fundamental of statistics; Sultan Chand & Sons

CorseCode: PBCA451	CourseTitle: PHP Lab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Install and configure the XAMPP server environment for PHP development.
2. Write PHP programs demonstrating variables, datatypes, constants, operators, and control structures.
3. Develop programs using user-defined and built-in PHP functions.
4. Implement various operations on indexed, associative, and multidimensional arrays.
5. Perform string manipulation using commonly used PHP string functions.
6. Create web forms and implement form handling using PHP, including the use of include and require statements.
7. Develop database-driven web applications by integrating PHP with MySQL for data insertion, retrieval, updating, and deletion.

CorseCode: PBCA452	CourseTitle: Java Lab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Develop simple Java programs to demonstrate class creation, object instantiation, and object references.
2. Write programs to perform various operations on arrays in Java.
3. Implement programs for string creation, manipulation, and commonly used string methods.
4. Develop applications demonstrating runtime and compile-time polymorphism.
5. Create and use built-in Java packages in application development.
6. Design and implement user-defined packages in Java.
7. Develop programs illustrating different types of inheritance.
8. Implement applications using interfaces and multiple inheritance through interfaces.
9. Create multithreaded applications using Java threads and thread synchronization.
10. Develop programs demonstrating exception handling and custom exceptions.

Semester: V

CorseCode: PBCA501	CourseTitle: ArtificialIntelligence& Machine Learning
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts, components, and application areas of Artificial Intelligence.
- Apply logical reasoning techniques for knowledge representation and intelligent problem-solving.
- Describe the architecture and functioning of knowledge-based and expert systems.
- Understand the principles and types of Machine Learning, including supervised, unsupervised, and reinforcement learning.
- Implement machine learning algorithms such as Linear Regression, Logistic Regression, Decision Trees, Support Vector Machines, K-Nearest Neighbors, and Neural Networks.
- Apply data preprocessing techniques, including data cleaning, normalization, and feature engineering.
- Evaluate and interpret machine learning models using performance metrics such as Accuracy, Precision, Recall, F1-Score, and ROC-AUC.

CourseOutcomes(COs)

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

- Explain the fundamental concepts, techniques, and applications of Artificial Intelligence and Machine Learning.
- Apply AI and ML algorithms to solve real-world problems effectively.
- Design and develop intelligent systems using appropriate tools, frameworks, and programming libraries.
- Evaluate, optimize, and interpret the performance of AI and ML models.
- Distinguish between supervised, unsupervised, and reinforcement learning approaches.
- Differentiate between linear and non-linear classification techniques.
- Explain the theoretical foundations and practical applications of Support Vector Machines (SVM).

PBCA501: Artificial Intelligence & Machine Learning

UNIT – I

General Issues and overview of AI: Concept of Intelligence, AI intelligent agents, Characteristics of AI, Comparison of AI. Defining problems as a State Space Search, Search and Control Strategies, Production systems, Problems – Water Jug problem, Block words Problem, Monkey & Banana problem, Applications of AI. Ethical considerations in AI development & deployment and MYCIN.

Unit-II

Searching- Searching for solutions, uninformed search strategies – Breadth first search, depth first Search. Informed search strategies (Heuristic search) Generate-and-test, Hill climbing, Best First Search, Constraint Satisfaction, A*, AO* Algorithms, Problem reduction, Game Playing-Adversarial search.

Knowledge Representation: Definition of Knowledge, Types of knowledge (Procedural and Declarative knowledge), Approaches to Knowledge Representation, Knowledge representation using Propositional and Predicate logic, Conversion to clause form, Resolution in Propositional logic, Resolution in Predicate logic.

Unit-III

Concepts: Machine Learning, Foundations-Overview, Applications, Types of Machine Learning, Basic Concepts in Machine Learning – Examples of Machine Learning.

Supervised Learning: Introduction, Linear Models of Classification – Decision Trees, Naïve Bayes Classification, Linear Regression – Logistic Regression – Bayesian Logistic Regression – Probabilistic Models.

Unit-IV

Unsupervised Learning: Clustering, Association rule mining, K-Means Clustering, EM (Expectation Maximization), Mixtures of Gaussians, EM algorithm in General, The Curse of Dimensionality, Dimensionality Reduction, Factor Analysis, Principal Component Analysis.

Recommended Books:

- Elaine Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw Hill, 3rd edition, 2012.
- Dan W. Patterson, “Introduction to Artificial Intelligence and Expert Systems”, Prentice Hall of India, 1st edition, 2012
- Winston, Patrick, Henry, “Artificial Intelligence”, Pearson Education, 3rd edition, 2014
- Subhasree Bhattacharjee, “Artificial Intelligence for Student” Shroff Publishers and Distributors Pvt. LTD., 1st Edition, 2016
- Stuart Russell, Peter Norving, “Artificial Intelligence: A Modern Approach”, Pearson

Education, 3rd edition, 2010.

- Mitchell T. M., Machine Learning, McGraw Hill
- Bishop C., Pattern Recognition and Machine Learning, Springer-Verlag
- Joel Grus, “Data Science from Scratch-First Principles with Python”, O’Reilly, 2015
- M. Gopal, “Applied MACHINE LEARNING”, McGraw-Hill, 2018
- Dr. Mahaveer Kumar Sain, “Introduction to Machine Learning”, Akinik Publications-New Delhi, 2021.

CorseCode: PBCA502	CourseTitle: Python Programming
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- DevelopastrongfoundationinPythonprogramming,includingsyntax,datatypes, operators, control structures, and functions.
- Apply Python programming techniques to solve a variety of computational and real-world problems.
- Effectivelyusebuilt-indatastructuresuchaslists,tuples,dictionaries,andsets.
- Designmodular,efficient,andreusablePythonprograms.
- UtilizePythonlibrariesuchasNumPy,Pandas,andMatplotlibfordataanalysis, visualization, and scientific computing.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Understandthe fundamentalconceptsofprogrammingusingPython.
- Design,develop,document,anddebugmodularPython applications.
- Applyappropriateprogrammingconstructsandbuilt-indatastructurestosolve computational problems.
- UsevariousPythondata objectsandlibrarieseffectively.
- Implementobject-orientedprogrammingconceptsusingclassesand objects.
- Performfilehandlingoperationsforreading,writing,andmanagingdata files.

PBCA502:PythonProgramming

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).

ListsandSets:Built-

inFunctions,ListtypebuiltinMethods,Tuples,TupleOperators,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

Exceptions: Concepts of Exceptions, Exceptions in Python, Detecting and Handling Exceptions, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions.

Recommended Books:

- R. Nageswara Rao, "Core Python Programming", Dreamtech Press, 2nd Edition, 2018
- Dr. M. Suresh Anand, Dr. R. Jothikumar, Dr. N. Vadivelan, "Python Programming", Notion Press, 1st Edition, 2020
- Martin C. Brown, "The Complete Reference Python", McGraw Hill Education, 4th Edition, 2021.
- Ashok Namdev Kamthane, "Programming and Problem Solving with Python", 2nd Edn, MGH, 2020
- Allen B. Downey, "Think Python", O'Reilly Media, 2016
- Sakis Kasampalis, Quan Nguyen, Dr. Gabriele Lanaro, Ingram, "Advanced Python Programming", short title, 2019

CorseCode: PBCA503	CourseTitle: DataCommunication& Computer Networks
CourseCredit:06	Hours/Week: 06

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandthefundamentalconceptsofdatacommunication,computernetworks, and various types of networks.
- ExplainthearchitectureandfunctioningoftheOSIandTCP/IPPreferencemodels, including the roles of their respective layers.
- Identifyanddescribenetworkinghardwarecomponents,transmissionmedia,and communication protocols.
- Analyzetheoperationofnetworkdevicesandprotocolsinmoderncommunication systems.
- Understandkeynetworksecurityconcepts,threats,vulnerabilities,andprotection mechanisms.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Demonstrateathoroughunderstandingofdatacommunicationandcomputernetw orking principles.
- Analyze, design, and evaluate network architectures, topologies, and communication models.
- Configure,manage, andtroubleshootbasicnetworkdevicesandprotocols.
- Implement appropriate network security techniques to safeguard data and communication channels.
- Recognize emergingtrends,technologies,andapplicationsinthe fieldof computer networks.

PBCA503:DataCommunication&ComputerNetworks

UNIT-I

Introduction: Network definition, Network topologies, Types of Networks, Layered network architecture, Categories of Network, protocol, Standards and interface.

Network Models: client-server, peer-to-peer, OSI reference model, Architecture and functions of layers. TCP/IP protocol suite.

UNIT-II

Data Communication Fundamentals: Analog and digital signal, Data-rate limits, Digital to digital & Digital to analog modulation. Guided and Unguided Transmission media

Data Link Layer and Network Devices Data link layer: framing, error detection and Corrections, flow control, Network devices: switches, routers, bridges, etc., MAC addressing and Ethernet standards.

UNIT-III

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

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

UNIT-IV

Circuit Switching: Simple Circuit Switching, Circuit Switching Networks, Packet Switching concepts and principles. Space Division switching, Time Division Multiplexing, Routing in Switching Networks, Control Signals & Channels.

Network security concepts: encryption, firewalls, VPN, Wireless networks and technologies.

Recommended Books:

- Behrouz A. Forouzan, "Data Communication and Networking", 4th edition, Tata McGraw Hill.
- A.S. Tanenbaum, "Computer Networks", Pearson Education Asia, 4th Ed.
- William Stallings, "Data and computer communications", Pearson Education Asia, 7th Ed.
- "Computer Networking: A Top-Down Approach" by James F. Kurose and Keith W. Ross.

CorseCode: PBCA551	CourseTitle: MachineLearningLab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Install and configure Python along with the essential libraries such as NumPy, Pandas, Matplotlib, and Scikit-learn.
2. Explore the features and applications of Scikit-learn as a machine learning library.
3. Set up the Python environment and verify the installation of all required machine learning packages.
4. Write a program to load and explore datasets from CSV and Excel files using Pandas.
5. Visualize datasets using Matplotlib or Seaborn by creating scatter plots, bar charts, histograms, and other appropriate visualizations.
6. Perform data preprocessing tasks, including handling missing values, encoding categorical variables, and feature scaling.
7. Implement a K-Nearest Neighbors (KNN) classifier using Scikit-learn, train it on a dataset, and evaluate its performance.
8. Develop a Linear Regression model for predictive analysis using continuous target variables.
9. Implement a Decision Tree classifier, train the model, and visualize the decision tree structure.
10. Apply K-Means clustering to a dataset and visualize the resulting clusters.

CorseCode: PBCA552	CourseTitle: Python Lab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Write a Python program to demonstrate the use of basic data types.
2. Create a list and perform various operations such as insert(), append(), remove(), pop(), clear(), and determine its length using len().
3. Create a tuple and demonstrate operations such as accessing elements, checking membership, determining length, and tuple concatenation.
4. Create a dictionary and perform operations including accessing items, updating values, retrieving values using get(), displaying all key-value pairs, and finding its length.
5. Develop a menu-driven Python program to perform arithmetic operations (addition, subtraction, multiplication, and division) using functions with arguments.
6. Write a Python program to determine whether a given number is positive, negative, or zero using conditional statements.
7. Use the filter() function to extract even numbers from a given list.
8. Write a program to display the current date and time using Python's datetime module.
9. Write a program to add a specified number of days to the current date and display the resulting date.
10. Write a program to count the occurrences of each character in a string and store the results in a dictionary.
11. Write a program to calculate the frequency of characters in a text file.
12. Using NumPy, create an array and display its type, dimensions, shape, and data type of elements.
13. Write a program to concatenate two Pandas DataFrames.
14. Read a CSV file using Pandas and display the first five and last five records.
15. Write a Python program to calculate the area of a circle for a given radius using the math module.
16. Load a sales dataset from a CSV file and create a line plot showing monthly profit using appropriate labels, title, legend, and grid.

Semester:VI

CorseCode: PBCA601	CourseTitle:.. NETFrameworkwith C#
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandthearchitecture,components,andfeaturesofthe.NETFrameworkandthe .NETecosystem.
- Developdynamic,interactive,andscalableweb applicationsusingASP.NET.
- ExploreWebForms,MVCarchitecture,andWebAPIsformodernapplicationdevelopment.
- ApplyADO.NETtechniquesfordatabaseconnectivity,damanipulation,and transaction management.
- Design,develop,test,anddeployenterprise-levelwebapplicationsusing.NET technologies.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Demonstrateacomprehensiveunderstandingofthe.NETFrameworkanditscore components.
- DevelopWindows-basedandweb-basedapplicationsusingASP.NETandrelated technologies.
- Designandimplementdatabase-drivenapplicationsusingADO.NET.
- Createprofessionalwebapplications,reports,anduserinterfacesusing.NETtoolsa nd frameworks.
- Deploy,maintain,andmanage.NETapplicationsinreal-worldenvironments.

PBCA601:..NETFrameworkwithC#

Unit-I

Introduction to .Net framework: Managed Code and the CLR Intermediate Language, Metadata and JIT Compilation Automatic Memory Management,The Framework Class Library: .Net objects- ASP .NET, NET web services, Windows Forms.

Program control structures: variables and constants, data types, identifiers, declaration and initialization,operatorsandtheirprecedence, expressions,consoleinputandoutput,decision-making statements including if, if-else, nested if, and switch-case; iterative statements including while, do-while, for, and foreach; jump statements such as break, continue, and return.

Unit-II

Types: ValuedatatypesStructures,Enumerations,Referencedatatypes, arrays.

Windows Programming: Creating windows forms windows controls, Button, check box, Combo box, Label, List box Radio Button, Text box, Events, Click, close deactivate, Load, mouse move, mouse down, mouse up.

Menus and Dialog Boxes: Creating menus, menu items, context menu, using dialog boxes, show dialog () method.

Unit– III

ADO.NET: Architecture of ADO.NET, ADO.NET providers, Connection, Command, Data Adapter, Dataset, Connecting to Data Source, Accessing Data with Data set and Data reader, Create an ADO.NET application, Using Stored Procedures.

ASP.NET Features: Application ofStates and Structure; ChangetheHomeDirectoryin IIS-Add a Virtual Directory in IIS- Set a Default Document for IIS – Change Log File Properties for IIS-Stop, Start, or Pause a Web Site.

Unit-IV

CreatingWebControls:WebControls,HTMLControls,UsingInternistControl,UsingInput Validation Controls, Selecting Controls for Applications, Data Controls and Adding web controls to a page.

CreatingWebForms:ServerControls,TypesofServerControls,AddingASP.NETCode to a page.

Web Services and WCF: Web Services protocol and standards – WSDL Documents-Visual Studio.NET Architecture of WCF, WCF Client

RecommendedBooks:

- MathewMacDonald:Beginning ASP.NET4.0inC# 2010,3rdEdition,APres.
- BillEvjenScott Hanselman,DevinRader:ProfessionalASP.NET4,2010,Willey.
- GeorgeShepherd:MicrosoftASP.NETStepbystep,2010MicrosoftPress.
- ImarSpaanjaars:BeginningASP.NET4:inC#andVB(Wrox Programmingto Programmer), 2010 Wiely Publishing.
- StevenHolzner;ASP.NET4.0(CoverC#&VB) BlackBook;DreamtechPress.
- StevenHolzner;.NETProgrammingBlackBook;DreamtechPress.

CorseCode: PBCA602	CourseTitle: IntroductiontoData Science
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandthefundamentalconcepts,processes, andapplicationsofData Science.
- Performdatacollection, cleaning,transformation,andanalysisusingRprogramming.
- CreatemeaningfulvisualizationsandinterpretdataeffectivelyusingRlibraries.
- Applystatisticaltechniques,machinelearningalgorithms,anddataminingmethodsto solve real-world problems.
- Designandimplementend-to-enddata scienceprojectsusingRandcommunicate insights effectively.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- DemonstrateastrongunderstandingofcoreDataScienceconceptsandmethodologies.
- Performdatamanipulation,preprocessing,andexploratorydataanalysisusingR.
- Analyzeandvisualizecomplexdatasetstoderiveactionableinsights.
- Buildandevaluatestatisticalandmachinelearningmodelsforpredictive analytics.
- Presentdata-drivenfindingseffectivelythroughvisualization,reporting,and storytelling.

PBCA602:IntroductiontoDataScience

Unit-I

IntroductiontoDataScience: Concept of Data Science, Need for Data Science, Components of Data Science, Big data, Facets of data: Structured data, Unstructured data, Machine-generated data, Graph-based or network data, Audio, image and video, Streaming data, The need for Business Analytics, Data Science Life Cycle, Applications of data science.

Unit-II

DataScienceProcess: Overview of data science process, setting the research goal, retrieving data, Cleansing, integrating and transforming data, Exploratory data analysis. Data Modelling, Presentation and automation

Data Analytics: Types of Analytics, Data Analytics Lifecycle: Overview - Discovery - Data Preparation - Model Planning - Model building, Regression analysis, Classification techniques, Clustering, Association rules analysis.

Unit-III

Statistics: Basic terminologies, Population, Sample, Parameter, Estimate, Estimator, Sampling distribution, Standard Error, Properties of Good Estimator, Measures of Central tendency, Measures of Spread, Probability, Normal Distribution, Binary Distribution, Hypothesis Testing, Chi-Square Test.

Unit-IV

Data Science Tools: Basic Data Science languages- R, Python, Knowledge of Excel, SQL Database, Introduction to Weka.

Recommended Books:

- Samuel Burns, “Fundamentals of Data Science: Take the first Step to Become a Data Scientist”, Amazon KDPP Printing and Publishing, First Edition, 2019
- Davy Cielen, Arno D. B. Meysman, Mohamed Ali, “Introducing Data Science”, Manning Publications, 2016
- Cathy O’Neil and Rachel Schutt, “Doing Data Science, Straight Talk from The Frontline”, O’Reilly. 2014.

CorseCode: PBCA603	CourseTitle: CloudComputing
CourseCredit:06	Hours/Week: 06

CourseObjectives(COs)

Bytheendofthiscourse, studentswillbeableto:

- Understandthefundamentalconcepts,principles,andservicemodelsofCloud Computing.
- Analyzeanddesigncloud-basedinfrastructureanddeploymentmodelsforvarious applications.
- Explorevirtualization technologiesand theirroleincloud environments.
- Identifycloudsecuritychallengesandimplement appropriatesecuritysolutions.
- Design effective backup,recovery, and data management strategies for cloud-basedsystems.

CourseOutcomes(COs)

Uponsuccessful completionofthiscourse,studentswillbeableto:

- Explainthekeyconcepts,benefits,challenges,andapplicationsofCloudComputing.
- Describe the relationship between parallel computing, distributed systems, and cloud computing.
- Implementvirtualizationtechnologiesandservice-orientedarchitecturesincloudenvironments.
- Selectappropriatecloudplatforms,services,anddeploymentmodelsforspecific applications.
- Install,configure,andutilizemoderncloudcomputingtoolsandtechnologies effectively.

PBCA603:CloudComputing

Unit-I

Introduction of Cloud Computing: Definition, Historical Developments, Enabling Technology, Vision, Essential Characteristics of Cloud Computing, Components of Cloud Computing. Challenges and Approaches of Migration into Cloud, Cloud Applications: Health care, CRM and ERP, Social Networking, Media Applications and Multiplayer Online Gaming. Benefits for the Market, Enterprise, End user and Individuals.

Unit-II

Cloud Computing Architecture: Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Interoperability and Standards, Scalability and Fault Tolerance.

Unit-III

Virtualization: Introduction, Characteristics of Virtualized Environment, Taxonomy of Virtualization Techniques, Virtualization and Cloud computing, Virtualization: of CPU, Memory, I/O Devices, Server, Desktop, Network, and data-center. Pros and Cons of Virtualization, Technology Examples-VMware and Microsoft Hyper-V, KVM, Xen.

Introduction of Cloud security services, Design Principles, Policy Implementation, Cloud Computing Security Challenges, Cloud Computing Security Architecture. Risk Mitigation, Understanding and Identification of Threats in Cloud, SLA-Service Level Agreements.

Unit-IV

Cloud Platforms in Industry: Amazon Web Services- Compute Services, Storage Services, Communication Services and Additional Services. Google AppEngine- Architecture and Core Concepts, Application Life-Cycle, cost model.

Recommended Books:

- Cloud Computing Principle and Paradigms, Edited by Rajkumar Buyya, James Broberg, A. Goscinski, Pub.- Wiley-2016
- Kumar Saurabh, "Cloud Computing", Wiley Pub 2016
- "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood
- "Cloud Computing: A Practical Approach" by Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter
- Mastering Cloud Computing by Raj kumar Buyya, Christian Vecchiola, S. Thamarai Selvi from TMH 2013.
- Distributed and Cloud Computing, Kai Hawang, Geoffrey C. Fox, Jack J. Dongarra Pub: Elsevier, 2013

CorseCode: PBCA651	CourseTitle: .NET withC# Lab
CourseCredit: 02	Hours/Week: 04

Recommended Practical Exercises

1. Develop a simple Windows Forms application using C# and .NET.
2. Implement array operations and control structures in C#.
3. Create applications demonstrating basic data structures such as stacks, queues, and lists.
4. Develop programs illustrating method overloading and constructor overloading.
5. Design user interfaces using various Windows Forms controls.
6. Implement multiple classes and interfaces in a C# application.
7. Handle user-generated events using event-driven programming techniques.
8. Create database applications using data controls and ADO.NET.
9. Implement dynamic data binding in Windows and web applications.
10. Design forms using validation controls to ensure proper user input.
11. Create and manage forms, dialog boxes, and message boxes.
12. Develop web applications using standard ASP.NET web controls.
13. Create and implement reusable User Controls in ASP.NET.
14. Design and deploy a complete ASP.NET website.
15. Implement session management and state management techniques.
16. Perform asynchronous operations and exception handling using AJAX.
17. Develop and consume Web Services and Windows Communication Foundation (WCF) services.

CorseCode: PBCA652	CourseTitle: Industrial Project
CourseCredit:02	Hours/Week: 04

The Project Training for the students is mandatory and it is an in-house training. The trainer will be from the industry to train them on the respective technologies and so it will be a paid training.

Two typed and duly bound copies of project report shall be submitted at least 3 weeks before commencement of the Theory/Practical examination which ever commences earlier.

Guidelines:

- It is a team work - team consisting of preferably two (in no case more than three) students. In special cases, a single student team is acceptable.
- Each team will be allotted a faculty member who will be their mentor.
- The topic will be allotted by the mentor at the beginning of the course.
- There will be three monthly presentations of 20 min. each and one ESE presentation of 40 minutes.
- It is advisable that the finalization of topic and major milestones is completed within 20 days from the date of start of the semester.
- The mentors will assess the progress of the students allocated to them on ongoing basis.

Semesters VII and VIII
2026-27
(Honours)

Semester: VII

Course Code: PBCA701	Course Title: Data Visualization and Interpretation
Course Credit: 04	Hours/Week: 04

Course Objectives (COs)

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

- Understand the principles of data visualization, visual perception, and effective data representation.
- Apply visualization techniques to communicate insights clearly and effectively.
- Analyze different types of data and select appropriate visualization methods.
- Perform statistical analysis and data manipulation using Python.
- Design interactive dashboards and reports using modern tools such as Power BI and Tableau.
- Apply storytelling techniques to present data-driven insights for decision-making.
- Evaluate the effectiveness of visualizations based on usability, clarity, and impact.

Course Outcomes (COs)

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

- Explain various data visualization tools, techniques, and best practices.
- Perform statistical analysis, hypothesis testing, and data preprocessing using Python.
- Create interactive and dynamic dashboards using Power BI and Tableau.
- Apply descriptive analytics and visualization techniques to solve real-world business problems.
- Communicate analytical findings effectively through visual storytelling and data-driven presentations.

PBCA701: Data Visualization and Interpretation

Unit-I

Introduction to Data Visualizations and Perception:

Definition, history, and importance of data visualization, Introduction of visual perception, visual representation of data, Data representation, Data Presentation, data visualization tools. Gestalt principles, Information overload. Data Ethics and Bias in Visualization, misleading Visualizations with Real-world Examples, Introduction to Data Storytelling.

Unit-II

Visual Representations: Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications. Chart types: Bar charts, histograms, scatter plots, box plots, heatmaps, treemaps. Design principles: Tufte's principles, color theory, spatial layout, and density. Dashboard design principles, Accessibility (color blindness, UX).

Unit-III

Classification of Visualization Systems & Tools: Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents. Python: Matplotlib, Seaborn, Plotly, Business Intelligence Tools: Power BI, Introduction to Tableau (Overview only). Visualization and Dashboards- Inbuilt visuals, Custom visuals, learn from existing reports, Visualization as a Tool tip, Final dashboard – putting it together Filter, slicer, bookmarks, buttons.

Unit-IV

Visualization, Interpretation and Analysis: Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization. Various visualization techniques, data structures used in data visualization. Exploratory Data Analysis (EDA) using Python (NumPy/Pandas). Analyzing time series data and geographical data. KPI Dashboard Design, Business Data Storytelling Techniques, Case Studies using Real-world Datasets.

References Books:

- Interactive Data Visualization: Foundations, Techniques, and Applications, Ward, Grinstein, Keim, Natick, 2nd edition, A K Peters, Ltd. 2015.
- Visualization Analysis & Design, Tamara Munzner, 1st Edition, A K Peters Visualization Series 2014
- Microsoft Power BI cookbook, Brett Powell, 2nd edition.
- Visual Analytics with Tableau, Alexander Loth, Wiley, 2019
- Python for Data Analysis, Wes McKinney, 3rd Edition, O'Reilly Media Inc., August 2022
- Python Data Analysis, Avinash Navlani, Armando Fandango, Ivan Idris, 3rd Edition, Packt Publishing, February 2021
- Murach's Python for Data Analysis, Scott McCoy, Murach, August 2021
- Data Analysis and Visualization Using Python, Dr. Ossama Embarak, Apress, November 2018

CourseCode: PBCA702	CourseTitle: DigitalImageProcessing
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts, components, and applications of Digital Image Processing.
- Analyze image transformation techniques and their practical applications.
- Apply image enhancement methods in both spatial and frequency domains.
- Explore image restoration, segmentation, representation, and compression techniques.
- Implement edge detection and image recognition algorithms for various applications.
- Understand the principles of image security, watermarking, and secure image transmission.

CourseOutcomes(COs)

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

- Explain the core concepts, techniques, and applications of Digital Image Processing.
- Implement image enhancement, filtering, restoration, and transformation algorithms.
- Apply image segmentation, representation, compression, and recognition methods to solve practical problems.
- Utilize edge detection techniques and security mechanisms in image processing applications.
- Analyze and evaluate digital image processing solutions for real-world scenarios.
- Demonstrate the foundation required to pursue advanced studies and research in Digital Image Processing.

BCA702: Digital Image Processing

Unit-I

Digital Imaging Fundamentals: Digital Imaging Fundamentals: Basic concept of digital image, image datasets and formats (JPEG, PNG, BMP), introduction to OpenCV, image formation in human eye, pixel and basic mathematical operations on digital images, sampling and quantization, applications of digital image processing.

Transform of Digital Images: Importance of image transforms, Fourier Transform (DFT),

Inverse Fourier Transform (IDFT), Fast Fourier Transform (FFT) and its applications in image processing.

Unit-II

Digital Image Enhancement: Introduction and importance of image enhancement, techniques in spatial and frequency domains, histogram processing and histogram equalization, basic filtering techniques including mean and median filters.

Frequency Domain Enhancement: Concept of frequency domain filtering, low pass, high pass and band pass filters and their applications in image processing.

Unit-III

Digital Image Compression: Digital Image Compression: Importance and types of image

compression, basic concepts of lossless and lossy compression, image compression standards, overview of compression techniques such as Huffman coding, DCT, and wavelet-based compression.

Image Restoration: Degradation model, basic restoration techniques including inverse filtering, Segmentation of Digital Images: Importance and applications of segmentation, thresholding, edge-based segmentation, watershed algorithm.

Unit-IV

Edge detection in Digital Image Processing: Edge Detection: Importance of edge detection in image processing, basic edge detection techniques and commonly used operators.

Security in Digital Image Processing: Introduction to image security, concepts of watermarking, image encryption, and steganography.

Introduction to Computer Vision Applications: Basic concepts of face detection using OpenCV and introduction to image classification

Reference Books:

- Fundamentals of Digital Image Processing, S. Annadurai, R. Shanmugalakshmi, Pearson Education, 2006.
- Digital Image Processing Using MATLAB, Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, Third Edition Tata Mc Graw Hill Pvt. Ltd., 2011.
- Digital Image Processing - Rafael C. Gonzalez, Richard E. Woods, 3rd Edition, Pearson, 2008
- Digital Image Processing - S. Jayaraman, S. Essakirajan, T. Veerakumar - TMH, 2010.

- Introduction to Digital ImageProcessing with Matlab, Alasdair McAndrew, Thomson Course Technology.
- Fundamentalsof DigitalImageProcessing-A.K.Jain, PHI, 2011.
- DigitalImageProcessingusingMatlab, RafealC.Gonzalez, RichardE.Woods, Steven L.Eddins, Pearson Education.
- DigitalImageProcessing, RafaelC.Gonzales, RichardE.Woods, ThirdEdition, Pearson Education, 2010.

CourseCode: PBCA703	CourseTitle: AdvanceJavaProgramming
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Strengthen and apply advanced object-oriented programming concepts using Java.
- Design and develop graphical user interface (GUI) and multithreaded applications.
- Implement database connectivity and data manipulation using JDBC.
- Develop dynamic web applications using Java Servlets and related technologies.
- Explore modern Java frameworks, including Spring and Spring Boot, for enterprise application development.

CourseOutcomes(COs)

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

- Develop robust Java applications using advanced programming concepts and design principles.
- Implement multithreading, exception handling, and event-driven programming effectively.
- Create database-driven applications by integrating Java with relational databases using JDBC.
- Design and deploy web applications using Servlets and Java-based web technologies.
- Build RESTful web services and enterprise applications using Spring Boot.
- Analyze and apply modern Java application architecture and development practices.

PBCA703:AdvancedJava Programming

UNIT-I

Core Advanced Java Concepts: Review of OOP Concepts in Java, Packages and Access Modifiers, Exception Handling (try-catch, throw, throws, custom exceptions), Multithreading: Thread lifecycle, Synchronization, Java Collections Framework: List, Set, Map, File Handling in Java

UNIT-II

JDBC and Database Programming: Introduction to JDBC, JDBC Architecture, Types of Drivers, Establishing Database Connection, Statement, Prepared Statement, Callable Statement, ResultSet Handling, CRUD Operations using JDBC, Transaction Management

UNIT-III

Web Programming using Servlets: Introduction to Web Applications, HTTP Protocol Basics, Servlet Architecture and Lifecycle, Generic Servlet and HTTP Servlet, Handling Client Requests (doGet, doPost), Session Management: Cookies, Http Session, Introduction to JSP (Overview only)

UNIT-IV

ModernJavaDevelopment:IntroductiontoRESTArchitecture,RESTfulWebServicesusing Spring Boot, Spring Boot Basics: Controllers, Dependency Injection, Creating REST APIs (GET, POST, PUT, DELETE), JSON Data Handling, Connecting Spring Boot with Database (JPA Overview), Introduction to Microservices Architecture

RecommendedBooks

- TheCompleteReference Java,HerbertSchildt
- Head FirstServlets&JSP,KathySierra
- SpringBootinAction,Craig Walls

CourseCode: PBCA704	CourseTitle: Data Warehousing&Data Mining
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts, principles, and applications of Data Warehousing and Data Mining.
- Analyze data warehouse architectures, schemas, and design methodologies.
- Apply data preprocessing, transformation, and visualization techniques for knowledge discovery.
- Implement data mining algorithms for classification, clustering, association, and prediction.
- Explore real-world applications, tools, and case studies in data warehousing and data mining.

CourseOutcomes(COs)

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

- Explain the concepts, components, and significance of data warehousing and data mining.
- Design and implement data warehouse models using appropriate schemas and methodologies.
- Perform data preprocessing, transformation, and visualization on large datasets.
- Apply data mining techniques and algorithms to extract meaningful patterns and insights.
- Analyze real-world business problems and develop data-driven solutions using warehousing and mining tools.

PBCA704:Data Warehousing&Data Mining

Unit-I

Introduction to Data Warehousing: Introduction, Data Warehouse importance and functions, Multidimensional Data Model, Data Mining and its usage, Cost of data Mining, Metadata, Data warehouse Architecture, Building a Data warehouse, Implementation, Further Development, Planning and Project Management of Data Warehouse.

Unit-II

Data Mining: Data Warehousing to Data Mining, Evolution Analysis, Classification of Data Mining Systems, Architecture of data mining system, Major Issues in Data Mining. Data preprocessing: Needs preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation; Analysis of Attributes Relevance. Discriminating between Different Classes. Data Warehouse and OLAP Technology for Data Mining.

Unit-III

Association Rules: Association Rule Mining, Single-Dimensional Boolean Association Rules from Transactional Databases. Apriori algorithm, Use of sampling for frequent item-set, FP tree algorithm, Multi-Level Association Rules from Transaction Databases. Issues regarding classification & prediction. Different Classification Methods- Decision Tree, Bayes Classification, Rule based, Classification by Back-Propagation, Prediction.

Unit-IV

Clustering and Applications of Data Mining: Cluster Analysis, Types of Data Categorization of Major Clustering Methods, Kmeans, Partitioning Methods, Hierarchical Methods, Density Based Methods, Grid Based Methods, Model-Based Clustering Methods, Clustering High Dimensional Data, Constraint Based Cluster Analysis, Outlier Analysis, Data Mining Applications.

Future Trends: Multidimensional Analysis and Descriptive Mining of Complex Data Objects, Active learning, Reinforcement learning, Web Data Mining. Spatial Databases, Multimedia Databases.

Recommended Books:

- Data Warehousing in the Real World – SAMANAHORY & DENNIS MURRAY. Pearson Edn Asia.
- Data Mining – Concepts and Techniques – JIAWEI HAN & MICHELINE KAMBER Hareourt India.
- Data Warehousing; Reema Thareja; Oxford
- Data Mining Introductory and advanced topics MARGARETH DUNHAM PEARSON EDUCATION.
- Data Warehousing in Real World Anahory, Pearson Education.
- Data Mining Techniques – ARUNKPUJARI, University Press.
- Building the Data Warehouse – W.H. Inmon, 3rd Edition, Wiley, 2003.
- Data Warehousing Fundamentals – PAUL RAJ PONNAIAH WILEY STUDENT EDITION.

CourseCode: PBCA705	CourseTitle: DeepLearning
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Gain a strong understanding of the fundamental principles, concepts, and architectures of deep learning.
- Develop the ability to design, build, and implement deep neural networks for solving complex machine learning problems.
- Acquire practical skills in using deep learning frameworks and tools for model development and training.
- Apply deep learning techniques to address real-world challenges across different domains.
- Understand current trends and advancements in deep learning research and applications.

CourseOutcomes(COs)

After completing this course satisfactorily, students will be able to:

- Explain the fundamental concepts and architectures of deep neural networks.
- Build and train deep learning models using appropriate libraries and frameworks.
- Apply deep learning techniques to solve practical, real-world problems.
- Evaluate, tune, and optimize deep learning models for improved performance and accuracy.
- Demonstrate awareness of recent developments and research trends in deep learning.

PBCA705:Deep Learning

Unit-I

Introduction to Deep Learning: Overview of deep learning concepts and its significance, Basics of neural networks and gradient based optimization, Activation functions and loss functions in deep learning, Gradient Descent Intuition, Overfitting and Regularization, Overview of Machine Learning, Introduction to Artificial Neural Network (ANN).

Unit-II

Deep Neural Network: Architectures Feedforward neural networks and backpropagation algorithm, Convolutional neural networks (CNNs) for image processing, Recurrent neural networks (RNNs) for sequential data analysis

Unit-III

Deep Learning Frameworks and Libraries: Introduction to popular deep learning frameworks (e.g., TensorFlow, PyTorch), Building and training deep neural networks using frameworks, Transfer learning and pre-trained models.

Unit-IV

Advanced Deep Learning Techniques: Generative adversarial networks (GANs) for data generation, Autoencoders and variational autoencoders (VAEs), Reinforcement learning with deep neural networks, Transfer Learning, Real-world Case Studies, Applications of Deep Learning Deep learning for image classification and object detection, Natural language processing and sentiment analysis with deep learning, Deep learning in recommendation systems and autonomous vehicles.

ReferenceBooks:

- DeepLearning, IanGoodfellow,YoshuaBengio,andAaronCourville.
- DeepLearningwithPython,FrancoisChollet,Manning

CourseCode: PBCA751	CourseTitle: Data Visualization and Interpretation Lab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Understand the fundamentals of data visualization, including types of data, data sources, and the principles of effective visual representation. Create a simple introductory visualization.
2. Get started with Python, Power BI, and Tableau by importing data from various file formats and creating basic visualizations such as line charts, bar charts, and treemaps.
3. Perform calculations in Tableau using aggregate functions such as SUM, AVG, COUNT, and create custom calculated fields.
4. Apply calculated fields to visualizations and format charts, labels, legends, and other visual elements effectively.
5. Edit and format axes, manipulate datasets, and perform data pivoting operations in Tableau.
6. Organize data through sorting, filtering, grouping, and data reshaping techniques.
7. Create advanced visualizations using filters, detail cards, size controls, tooltips, and customized color schemes.
8. Design interactive dashboards and stories by incorporating filters, actions, and navigation components.
9. Publish, share, print, and export dashboards and reports using Tableau Online, Power BI Service, or equivalent platforms.
10. Develop advanced visualizations, including dual-axis charts, cyclical charts, circular area charts, and other custom visual representations.

CourseCode: PBCA752	CourseTitle: DigitalImageProcessingLab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Load and display grayscale and color images using an image processing tool such as MATLAB or Python.
2. Generate the negative of a binary and grayscale image.
3. Perform basic geometric transformations such as resizing, rotation, and flipping.
4. Enhance image contrast using contrast stretching and histogram equalization.
5. Separate and display the individual bit planes of a grayscale image.
6. Compute and visualize the 1-D and 2-D Fast Fourier Transform (FFT) of an image.
7. Calculate basic statistical measures such as mean, standard deviation, and correlation coefficient for an image.
8. Apply smoothing filters such as Mean and Median filter to reduce image noise.
9. Perform image sharpening using gradient-based filters and Laplacian operators.
10. Detect edges using Sobel, Prewitt, and Canny edge detection techniques.
11. Compress an image using basic compression techniques such as DCT and Huffman coding.
12. Restore degraded images using filtering and noise removal techniques.
13. Apply intensity slicing for image enhancement and region highlighting.

CourseCode: PBCA753	CourseTitle: AdvanceJavaLab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Develop a Java program to demonstrate the creation and execution of multiple threads.
2. Implement exception handling using try-catch-finally blocks and user-defined exceptions.
3. Write a Java program to perform file handling operations such as reading from and writing to files.
4. Create a JDBC Application to perform Insert, Update, and Delete operations on a database.
5. Develop a JDBC program using PreparedStatement for secure and efficient database interactions.
6. Implement a JDBC application to navigate and retrieve records using the ResultSet interface.
7. Create a Servlet-based web application to process and validate HTML form data.
8. Implement session management in a web application using cookies and HTTP sessions.
9. Develop a simple RESTful API using Spring Boot.
10. Create a CRUD (Create, Read, Update, Delete) application using Spring Boot and a relational database.

Semester-VIII

CourseCode: PBCA801	CourseTitle: Big Data Analytics
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts and importance of Big Data Analytics.
- Learn the Big Data framework, its characteristics, and real-world use cases.
- Gain knowledge of the Hadoop ecosystem and its role in handling large-scale data.
- Develop the ability to analyze industry-based Big Data problems using appropriate tools and techniques.

CourseOutcomes(COs)

After completing this course satisfactorily, students will be able to:

- Explain the key characteristics and challenges of Big Data.
- Describe the Hadoop framework and its core components.
- Apply HDFS and MapReduce concepts to solve Big Data analytics problems.
- Understand the architecture and functioning of YARN in the Hadoop ecosystem.
- Demonstrate knowledge of sorting, shuffling, and data processing mechanisms in distributed computing.

PBCA801:BigDataAnalytics

Unit-I

Introduction: Big data, Facts about Big Data, importance of Big Data Evaluation of Big Data a, –

Market Trends, Sources of Data Explosion, Types of Data, Need of Big Data, Big Data and its sources, Characteristics of Big Data – Difference between Traditional IT Approach and Big Data Technology, Capabilities of Big Data, Handling Limitations of Big Data, Technologies Supporting Big Data, Big Data Case Study for Netflix.

Unit-II

Introduction to Hadoop: Introduction, History and Milestone of Hadoop, Core Components of Hadoop, Difference between Regular File System and HDFS, Common Hadoop Shell Commands, Apache Spark (RDD, Data Frames, Spark SQL).

Unit-III

Concepts and Architecture: Data Flow (File Read, File Write), Fault Tolerance, Java Base API, Different Daemons in Hadoop cluster, Data Pipelines, Streaming Basics (Kafka Introduction), Loading a dataset into the HDFS.

Unit-IV

Introduction to Data Processing and Data Storage tools: Introduction of MapReduce, YARN, HDFS, HBase. Analogy of MapReduce, MapReduce Architecture, Example of MapReduce, Sorting, Shuffling, Reducing, Combiner, Partitioner. Cloud Platforms (AWS/GCP Basics).

ReferenceBooks:

- BigDataandAnalytics,SeemaAcharya,SubhashiniChellappan,(2015).WileyPublication.
- DataScienceandBigDataAnalytics:Discovering,Analyzing,VisualizingandPresenting Data (2015), EMC Education Services.
- BigData,BlackBook:CoversHadoop.
- MapReduce,Hive,YARN,Pig,Rand DataVisualization(2016),DTEditorial Services

CourseCode: PBCA802	CourseTitle: IOT
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Understand the fundamental concepts and principles of the Internet of Things (IoT).
- Study the IoT reference architecture and its components.
- Learn the basics of IoT communication protocols and standards.
- Gain knowledge of technologies, frameworks, and standards related to IoT systems.
- Develop skills in IoT system planning, design, and implementation strategies.

CourseOutcomes(COs)

After completing this course satisfactorily, students will be able to:

- Explain the key technologies and standards used in Internet of Things (IoT).
- Describe the IoT ecosystem required for large-scale deployment and adoption.
- Analyze the components and architecture of IoT systems.
- Develop technical strategies for IoT solutions at organizational and enterprise levels.
- Apply IoT concepts to design and implement real-world applications.

PBCA802:IOT

Unit-I

IoT: The Internet of Things Today, Time for Convergence, Towards the IoT Universe, Internet of Things Vision, IoT Strategic Research and Innovation Directions, Future Internet Technologies, Infrastructure, Networks and Communication, Processes, Data Management, Security, Privacy & Trust, Device Level Energy Issues, IoT Related Standardization, IoT Applications.

Unit-II

M2M to IoT: A Basic Perspective— Introduction, Some Definitions, Difference between IoT and M2M, M2M Value Chains, IoT Value Chains, an emerging industrial structure for IoT, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.

Unit-III

IoT Design and Platforms: Introduction to IoT design methodology with weather monitoring case study, logical design using Python (basic concepts and libraries), MQTT protocol and cloud integration.

IoT Hardware and Applications: Sensors, actuators, IoT devices, Arduino and Raspberry Pi, basic Linux and GPIO programming, applications in home automation, smart cities, and agriculture.

Unit-IV

Internet of Things Privacy: Security and Governance Introduction, Overview of Governance, Privacy and Security Issues, Security, Privacy and Trust in IoT-Data-Platforms for Smart Cities, Smart City Case Studies, Agriculture IoT Applications, First Steps Towards a Secure Platform, Smart Approach. Data Aggregation for the IoT in Smart Cities, Security.

Reference Books:

- Nitesh Dhanjani, Abusing the Internet of Things, Shroff Publisher/O'Reilly Publisher.
- Internet of Things, RMD Sundaram Shriram K Vasudevan, Abhishek S Nagarajan, John Wiley and Sons.
- Internet of Things, Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, John Wiley & Sons.
- Cuno Pfister, "Getting Started with the Internet of Things", Shroff Publisher/Maker Media.
- Francis da Costa, "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything", 1st Edition, Apress Publications.
- Massimo Banzi, Michael Shiloh Make: Getting Started with the Arduino, Shroff Publisher/Maker Media Publishers.

CourseCode: PBCA803	CourseTitle: Network Security and Cryptography
CourseCredit:04	Hours/Week: 04

Course Objectives (COs)

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

- Understand fundamental concepts of cryptography and network security.
- Learn various encryption techniques along with hash functions, Message Authentication Codes (MAC), and digital signatures.
- Analyze the role of cryptographic techniques in securing networks and computer systems.
- Apply security principles to design secure communication protocols and systems.

Course Outcomes (COs)

After completing this course satisfactorily, students will be able to:

- Analyze and design classical encryption techniques and block ciphers.
- Understand and evaluate the Data Encryption Standard (DES) and its applications.
- Apply public-key cryptography techniques including RSA, Diffie-Hellman Key Exchange, and ElGamal cryptosystem.
- Understand key management, distribution mechanisms, and user authentication protocols.
- Design and analyze hash functions, MAC algorithms, and digital signature schemes.
- Implement security protocols such as PGP, S/MIME, IPsec, SSL, TLS, HTTPS, and SSH.
- Understand intrusion detection techniques, malicious software, and cyber threats.
- Explain firewall architectures, types, configurations, and deployment strategies.

PBCA803: Network Security and Cryptography

Unit-I

Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security. Classical Encryption Techniques, Block Cipher Principles, The Data Encryption Standard (DES), The Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles.

Unit-II

Pseudorandom Number Generation: Principles of Pseudorandom Number Generation, Number Theory-Divisibility and the Division Algorithm, The Euclidean Algorithm, Modular Arithmetic, Prime Numbers, Fermat's and Euler's Theorems, Testing for Primality, The Chinese Remainder Theorem, Discrete Logarithms. Public-Key Cryptography, RSA and other Public-Key Crypto Systems: Principles of Public-Key Cryptosystems, The RSA Algorithm.

Unit-III

Cryptographic Hash Functions: Applications of Cryptographic Hash Function, Two Simple

Hash Functions Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA). Message Authentication Codes: Message Authentication

Requirements, Message Authentication Functions, Message Authentication Codes, Security of MACs, MACs Based on Hash Functions (HMAC). Digital Signatures- Digital Signature Standard (DSS),

Unit-IV

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption and Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates. User Authentication Protocols: Remote User Authentication Principles.

Web Security Issues: Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTP S, Secure Shell (SSH). IPSECURITY: IP Security Overview, IP Security Policy. Intruders- Intruders, Intrusion Detection. Malicious Software: Types of Malicious Software, Viruses, Worms. Firewalls: The Need for Firewalls, Firewall Characteristics, Types of Firewalls, Firewall Configurations. Wireshark Tool Basics, Ethical Hacking Introduction, Case Studies of Cyber Attacks.

Reference books:

- Cryptography And Network Security- Principles and Practice, William Stallings, 5th Edition, Pearson/PHI, 2011.
- Network Security Essentials (Applications and Standards), William Stallings, 4th Edition, Pearson Education. ,2012
- Network Security – Private Communication in a Public World, Charlie Kaufman, Radia Perlman and Mike Speciner, 2nd Edition, Pearson/PHI, 2002.
- Fundamentals of Network Security, Eric Maiwald, 1st Edition, Dreamtech Press, 2003.
- Principles of Information Security, Whitman, 3rd Edition, Thomson, 2009.
- Network Security: The complete reference, Robert Bragg, Mark Rhodes, 1st Edition, TMH, 200

CourseCode: PBCA804	CourseTitle: Computer Graphics
CourseCredit:04	Hours/Week: 04

CourseObjectives(COs)

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

- Provide a foundational understanding of computer graphics hardware, including input and output devices and display technologies.
- Deliver an in-depth exploration of 2D raster graphics, 3D modeling, and the mathematical principles of geometric transformations.
- Introduce the core concepts of 3D viewing, shading, and rendering pipelines.
- Equip students with a comprehensive overview of the fundamental principles governing modern digital imagery.

CourseOutcomes(COs)

After completing this course satisfactorily, students will be able to:

- Describe the operational mechanics of various computer graphics hardware and display technologies.
- Implement and compute mathematical 2D and 3D object transformations (scaling, rotation, translation).
- Demonstrate proficiency in 2D and 3D viewing techniques, including clipping, projections, and window-to-viewport mapping.
- Utilize fundamental modeling and rendering principles to create and manipulate spatial objects in a digital environment.

PBCA804:ComputerGraphics

Unit-I

Basics of Computer Graphics: Introduction, Area of Computer Graphics, Design and Drawing, Animation Multimedia applications, Simulation, Difficulties for displaying pictures. GRAPHIC DEVICES Cathode Ray Tube, Quality of Phosphors, CRTs for Color Display, Direct View Storage Tube, Tablets, The light Pen, Three-Dimensional Devices, Graphics software and standards, Applications of computer graphics.

Unit-II

Basics Graphics programming : initializing the graphics, C Graphical functions, simple programs SIMPLE LINE DRAWING METHODS Point Plotting Techniques, Qualities of good line drawing algorithms, The Digital Differential Analyzer (DDA), Bresenham's Algorithm, Generation of Circles, Basic transformation, Need for Clipping and Windowing, Line Clipping Algorithms, The midpoint subdivision Method, Other Clipping Methods, Sutherland - Hodgeman Algorithm, Viewing Transformations, OpenGL / WebGL Basics.

Unit-III

2D transformation and viewing: Animation Basics, Rendering Techniques, Transformations (translation, rotation, scaling), matrix representation, homogeneous coordinates, composite transformations, reflection and shearing, viewing pipeline and coordinates system, window-to-viewport transformation, clipping including point clipping, line clipping (Cohen Sutherland, liang- bersky, NLN), polygon clipping.

3D concepts and object representation: 3D display methods, polygon surfaces, tables, equations, meshes, curved lies and surfaces, quadric surfaces, spline representation, cubic spline interpolation methods, Bazier curves and surfaces, B-spline curves and surfaces.

Unit-IV

Advance topics: Visible surface detection concepts, back-face detection, depth buffer method, illumination, light sources, illumination methods (ambient, diffuse reflection, specular reflection), Color Models: properties of light, XYZ, RGB, YIQ and CMY color models.

Referencebooks:

1. ComputerGraphics,D.Hearn andP.Baker-PearsonEducation-CVersion
2. ComputerGraphics,Sinha&Udai,-TMH4.ComputerGraphics,Foleyand vanDam-Person Education
3. Theory&ProblemofComputerGaphics,Plastock,SchaumSeries.
4. FundamentalsofInteractiveComputerGraphics,Foley&VanDam,Addison-Wesley.
5. PrinciplesofInteractiveComputerGraphics,Newman,McGrawHill

CourseCode: PBCA851	CourseTitle: Big Data Lab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Install and configure Hadoop, set up the working environment, and explore basic Hadoop processes.
2. Practice basic Hadoop commands for managing and manipulating big data.
3. Configure and set up a Hadoop cluster environment.
4. Implement and explore HDFS operations and its key characteristics.
5. Study and analyze the features and properties of YARN architecture.
6. Import data from MySQL database into HDFS.
7. Import data from MS Excel files into HDFS.
8. Develop and implement Reducer programs in Hadoop MapReduce.
9. Implement custom Partitioner in Hadoop MapReduce framework.
10. Develop and execute nested MapReduce programs for complex data processing tasks.

CourseCode: PBCA852	CourseTitle: IOTLab
CourseCredit:02	Hours/Week: 04

Recommended Practical Exercises

1. Develop a program to blink 5 LEDs sequentially in forward and reversed direction using a microcontroller (Arduino).
2. Interface and control a relay module using Arduino.
3. Design and implement an intrusion detection system using ultrasonic and sound sensors.
4. Develop a program to control the speed and direction of a DC motor using Arduino.
5. Implement a smart street light system using an LDR sensor for automatic light control.
6. Develop a system to classify dry and wet waste using a moisture sensor (DHT22 or equivalent).
7. Design a program to measure and display pH values of different substances such as milk, lime water, and pure water.
8. Implement a gas leakage detection system using gas sensors.
9. Develop a weather monitoring system using Arduino to display environmental parameters.
10. Setup UART communication protocol and transmit/receive data strings between devices.
11. Design a water level detection system using an ultrasonic sensor.
12. Simulate and implement keypad interfacing to record and process keystrokes.

CourseCode: PBCA853	CourseTitle: NetworkSecurityandCryptogr aphy Lab
CourseCredit:02	Hours/Week: 04

RecommendedPracticalExercises(UsingCor Java)

1. Introductiontocryptography-basedsecuritytools.
2. Writeaprogramto implementsymmetric encryption.
3. Writeaprogram toimplementasymmetricencryption.
4. Writeaprogramthatcontainsastring“HelloWorld”andappliesXORwith0,then display the result.
5. WriteaprogramtoperformANDandXORoperationsonthestring“HelloWorld” with127 and display the output.
6. ImplementCaesarCipher,SubstitutionCipher,andHillCipher algorithms.
7. ImplementtheDESencryptionalgorithmlogic.
8. ImplementtheBlowfishalgorithmlogic.
9. ImplementtheRijndael (AES)algorithmlogic.
10. Encryptthetext“HelloWorld”usingBlowfishandgenerateakeyusingJavaKeyTool.
11. ImplementtheRSAalgorithmforencryptionanddecryption.
12. ImplementtheDiffie-HellmanKeyExchangemechanismusingHTMLandJavaScript.
13. GeneratemessagedigestusingSHA-1algorithm.
14. GeneratemessagedigestusingMD5 algorithm.
15. Prepareacasestudyoncybersecurityconceptsand applications