

MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati – 517 102



SCHOOL OF COMPUTING

Bachelor of Computer Applications BCA

(3 Years Degree)

Bachelor of Computer Applications (Hons.) BCA(Hons.)

(4 Years Degree)

CURRICULUM AND SYLLABUS (From 2025-26 Admitted Students)

FULLY FLEXIBLE CHOICE BASED CREDIT SYSTEM (FFCBCS)



MBU
MOHAN BABU
UNIVERSITY

MOHAN BABU UNIVERSITY

Vision

To be a globally respected institution with an innovative and entrepreneurial culture that offers transformative education to advance sustainability and societal good.

Mission

- ❖ Develop industry-focused professionals with a global perspective.
- ❖ Offer academic programs that provide transformative learning experience founded on the spirit of curiosity, innovation, and integrity.
- ❖ Create confluence of research, innovation, and ideation to bring about sustainable and socially relevant enterprises.
- ❖ Uphold high standards of professional ethics leading to harmonious relationship with environment and society.

SCHOOL OF COMPUTING

Vision

To lead the advancement of computer science research and education that has real-world impact and to push the frontiers of innovation in the field.

Mission

- ❖ Instil within our students fundamental computing knowledge, a broad set of skills, and an inquisitive attitude to create innovative solutions to serve industry and community.
- ❖ Provide an experience par excellence with our state-of-the-art research, innovation, and incubation ecosystem to realize our learners' fullest potential.
- ❖ Impart continued education and research support to working professionals in the computing domain to enhance their expertise in the cutting-edge technologies.
- ❖ Inculcate among the computing engineers of tomorrow with a spirit to solve societal challenges.

DEPARTMENT OF COMPUTER APPLICATIONS

Vision

To become a center of excellence in the field of computer science and applications.

Mission

- Imparting knowledge and skills through contemporary curriculum to the diverse group of students.
- Creating a talent pool of faculty in diverse domains of computer applications through continuous training.
- Domain and transferable skill development for the holistic personality of students to inculcate values and ethics for effective professional practice and as an entrepreneur.

BACHELOR OF COMPUTER APPLICATIONS (BCA)

PROGRAM EDUCATIONAL OBJECTIVES

After few years of graduation, the graduates of BCA Program will be able to:

- PEO1.** Pursue higher education in the core and allied areas of computer science and applications.
- PEO2.** Become professionals in industry or academia with an ability to adapt to evolving technologies in cultivating skills for thriving career.
- PEO3.** Become successful entrepreneurs and be engaged in technology innovation and deployment.
- PEO4.** Exhibit professionalism, and uplifting health, safety, legal, environmental, ethical, and cultural diversity issues for serving the society and communicating with local and national peers, bound within regulations and leading to lifelong learning.

PROGRAM OUTCOMES

On successful completion of the Program, the graduates of BCA Program will be able to:

- PO1. Computational Knowledge:** Apply knowledge of mathematics and computing fundamental and domain concepts to find solutions for defined problems.
- PO2. Problem Analysis:** Adapt basic principles of Mathematics and Computing sciences to identify and formulate problems to provide solutions for complex problems.
- PO3. Design/Development of Solutions:** Design, analyze and develop solutions and evaluate system components or processes to meet a specific need for local, regional and global public health, societal, cultural, and environmental systems.
- PO4. Conduct Investigations of Complex Computing Problems:** Use research-based knowledge and methods with the ability to perform analysis and development to reach valid conclusions.
- PO5. Modern Tool Usage:** Design, analyze and develop solutions using modern tools by considering the limitations.

- PO6. Professional Ethics:** Commit to ethics and cyber regulations, responsibilities, and norms in computing practices.
- PO7. Life-long Learning:** Enhance understanding and building research, strategies in independent learning for continual development as computer applications professional.
- PO8. Project Management and Finance:** Establishing strategies for developing and implementing ideas in multi-disciplinary environments using computing and management skills as a member or leader in a team.
- PO9. Communication Efficacy:** Contribute to progressive community and society in comprehending computing activities by writing effective reports, designing documentation, making an effective presentation, and understanding instructions.
- PO10. Societal and Environmental Concern:** Apply mathematics and computing knowledge to access and solve issues relating to health, safety, societal, environmental, legal, and cultural issues within local, regional and global contexts.
- PO11. Individual and Teamwork:** Self and continuous learning to improve knowledge and competence as a member or leader of a team.
- PO12. Innovation and Entrepreneurship:** Identify a timely opportunity and use innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

PROGRAM SPECIFIC OUTCOMES

On successful completion of the Program, the graduates of BCA Program will be able to:

- PSO1.** Design, develop, test, and deploy computing applications of specified requirements using programming skills.
- PSO2.** Apply tools and techniques to create database and adapt managerial and domain Skills of Information Management for storage and decision making.
- PSO3.** Apply suitable techniques and algorithms to integrate Operating System Services, Network devices, Security mechanisms and Infrastructure to meet the requirements for the deployment of an application on networks.



MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati 517 102

Centre for Distance and Online Education

B.C.A. – Bachelor of Computer Applications

Academic Year 2025-26

Course Structure

I-Semester

Sl. No	Course Code	Course Title	Lecture	Tutorial	Practical	Project-based Learning	Credits
			L	T	P	S	
1.	CA25UG0501L	COMPUTER HARDWARE AND SYSTEM ESSENTIALS	-	1	2	-	2
2.	CA25UG0101L	MATRIX THEORY AND LINEAR ALGEBRA	3	-	-	-	3
3.	CA25UG0102L	COMPUTER FUNDAMENTALS AND ORGANIZATION	3	-	-	-	3
4.	CA25UG0201L	DATABASE MANAGEMENT SYSTEMS	3	-	2	-	4
5.	CA25UG0202L	PROGRAMMING FOR PROBLEM SOLVING	3	-	2	-	4
6.	CA25UG0103L	INNOVATION AND DESIGN THINKING	-	-	-	-	2
7.	CA25UG0104L	PROFESSIONAL ETHICS AND HUMAN VALUES	-	-	-	-	2
TOTAL			12	1	6	-	20



MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati 517 102

Centre for Distance and Online Education

B.C.A. – Bachelor of Computer Applications

Academic Year 2025-26

Course Structure

II-Semester

Sl. No	Course Code	Course Title	Lecture	Tutorial	Practical	Project-based Learning	Credits
			L	T	P	S	
1.	CA25UG0105L	PRINCIPLES OF ACCOUNTING	3	-	-	-	3
2.	CA25UG0106L	NUMERICAL ANALYSIS, PROBABILITY AND STATISTICS	3	-	-	-	3
3.	CA25UG0107L	BASICS OF VIRTUALIZATION AND CLOUD TECHNOLOGY	3	-	-	-	3
4.	CA25UG0108L	COMPUTER COMMUNICATION	3	-	-	-	3
5.	CA25UG0203L	DATA STRUCTURES	3	-	2	-	4
6.	CA25UG0401L	OBJECT ORIENTED PROGRAMMING THROUGH C++	3	-	2	4	5
7.	CA25UG0204L	GENERAL ENGLISH	2	-	2	-	3
8.	CA25UG0109L	HUMAN RESOURCE MANAGEMENT	3	-	-	-	3
TOTAL			23	-	6	4	27



MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati 517 102

Centre for Distance and Online Education

B.C.A. – Bachelor of Computer Applications

Academic Year 2025-26 Batch – II

Course Structure

III-Semester

Sl. No	Course Code	Course Title	Lecture	Tutorial	Practical	Project-based Learning	Credits
			L	T	P	S	
1.	25CA104006S	PROBLEM SOLVING THROUGH JAVA	3	-	2	4	5
2.	25CA101005	SOFTWARE ENGINEERING	3	-	-	-	3
3.	25CA102009	COMPUTER NETWORKS	3	-	2	-	4
4.	25CA102014	CLOUD COMPUTING	3	-	2	-	4
5.	25CA101017	ARTIFICIAL INTELLIGENCE	3	-	-	-	3
6.	25CA101011	INTERNET OF THINGS	3	-	-	-	3
TOTAL			18	-	6	4	22

Course Code	Course Title	L	T	P	S	C
CA25UG0501L	COMPUTER HARDWARE AND SYSTEM ESSENTIALS	-	1	2	-	2

Pre-Requisite -

Anti-Requisite -

Co-Requisite -

COURSE DESCRIPTION: This course provides a detailed discussion and hands-on experience on internal components of a computer, assemble a computer system, install an operating system, and troubleshoot using system tools and diagnostic software. Students will also be able to understand various network cables, connectors and TCP/IP networks, and work group.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

CO1. Identify different hardware components on personal computer and peripheral devices.

CO2. Create network architecture using TCP and UDP protocols for data transmission.

CO3. Devise the solutions for the problems occurred in personal computer in Operating Systems.

CO4. Implement the functionalities of different peripheral devices and networks by configuring multi-functional devices.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	2	2	-	2	1	1	1	2	1	3	-	2
CO2	3	3	3	3	2	-	2	2	2	2	2	2	3	-	3
CO3	3	3	3	3	2	-	2	2	2	2	2	2	3	-	3
CO4	3	3	3	3	3	-	2	2	2	2	2	2	3	-	3
Course Correlation Mapping	3	3	3	3	2	-	2	2	2	2	2	2	3	-	3

Correlation Levels: 3: High; 2: Medium; 1: Low

EXPERIENTIAL LEARNING

HARDWARE CONFIGURATION

1. Peripherals of the computer
2. Implement the following activities:
 - a) Configure settings using BIOS/UEFI tools on a PC.
 - b) Identify different components of mother board
 - c) Analyze various RAM types, PC extensions cards and storage devices
3. Implement the following activities:
 - a) Identify various CPU's, cooling methods and PC connection interfaces
 - b) Identification of various power supply, display devices and common PC Connectors
 - c) Install and configure common peripheral devices and SOHO multi-function device

PC Networking using Packet Tracer

Network Types and Components, Topologies, Wired and Wireless Transmission, Protocols.

4.
 - a) Identify various types of network cables and connectors and characteristics
 - b) Implement the following characteristics of TCP/IP
 - i) IPv4 and IPv6
 - ii) Client side DNS Settings
5.
 - a) Identify following TCP and UDP Ports
 - i) 21-FTP
 - ii) 443-HTTPS
 - iii) 80-HTTP
 - iv) Telnet
 - b) Analyze the following TCP and UDP protocols
 - i) SMD
 - ii) SNMP
 - iii) DHCP
6.
 - a) Configure the following network types
 - i) LAN
 - ii) WAN
 - iii) WLAN
 - b) Configure network architecture using the following
 - i) HUB
 - ii) Switch
 - iii) Router

OPERATING SYSTEMS

Introduction to Operating System, Characteristics of Operating System, Types of Operating System and its components

7. Installation of Windows Operating System
8. Installation of Application and Device Drivers management

TROUBLESHOOTING

Introduction to trouble shooting, Hardware and Software Trouble shooting

9.
 - a) Study the common problems related to the following
 - i) Mother Board
 - ii) RAM
 - iii) CPU
 - iv) Power
 - b) Troubleshoot the following failures
 - i) Read/Write Failure
 - ii) Slow Performance
 - iii) Failure to boot
10. Trouble Shoot the following symptoms of Video, projector and display issues
 - i) VGA Mode
 - ii) No Image of Screen
 - iii) Dead Pixels
 - iv) Color patterns
11. Trouble shoot the following issues of wired, wireless and mobile devices
 - i) No Connectivity
 - ii) IP conflict
 - iii) Ghost Cursor
 - iv) Sticking Keys

RESOURCES

REFERENCES:

1. David Anfinson, Allan Johnson and Kathleen Czurda, IT Essentials v7 Companion Guide, CISCO Press, 2020
2. Brian W. Kernighan, Understanding the Digital World: What You Need to Know about Computers, the Internet, Privacy, and Security, Second Edition, Princeton University Press, 2021
3. Kavin Wilson, Exploring Computer Systems: The Illustrated Guide to Understanding Computer Systems, Hardware & Networks, Elluminent Press, 2019

SOFTWARE/TOOLS:

1. Windows 8/10 operating systems
2. Cisco Packet Tracer

VIDEO LECTURES:

1. <https://www.edx.org/course/computer-hardware-and-operating-systems>
2. <https://www.coursera.org/learn/computer-hardware-software>

WEB RESOURCES:

1. <https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/interface/configuration/15-s/ir-15-s-book.pdf>
2. <https://www.certexams.com/comptia/a+/cert-notes-aplus-networking.htm>
3. <https://www.rcboe.org/cms/lib/GA01903614/Centricity/Domain/4399/Network%20n10-007.pdf>
4. <https://www.tutorialsworld.com/CertNotes/CompTIA-cert/A+/aplu-prac-10.htm>

Course Code	Course Title	L	T	P	S	C
CA25UG0101L	MATRIX THEORY AND LINEAR ALGEBRA	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course provides a discussion on matrix factorizations, linear systems, Eigenvalues and Eigenvectors, vector spaces, linear transformations, and orthogonality.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate the concepts of matrix factorizations and solutions of the linear system.
- CO2.** Determine the Eigenvalues and Eigenvectors of the matrix and implement the concept of Eigen values and decompositions of a matrix in intelligent systems.
- CO3.** Apply the concepts of vector spaces and linear transformation on problems of computational systems.
- CO4.** Use the inner product spaces for the study of orthogonality and to construct orthonormal basis.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO 4	PO 5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	-	2	1	1	-	1	2	3	2	2
CO2	3	3	3	3	3	-	2	2	1	-	1	2	3	2	3
CO3	3	3	3	3	3	-	2	2	1	-	2	2	3	2	3
CO4	3	3	2	3	2	-	2	1	1	-	2	2	3	2	3
Course Correlation Mapping	3	3	3	3	3	-	2	2	1	-	2	2	3	2	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: MATRICES AND LINEAR SYSTEM OF EQUATIONS (10 Periods)

Elementary Row-operations, Elementary Matrices, Echelon form, Rank of a matrix by row-reduction, Solutions of system of linear equations by row reduction, Matrix Factorization, LU factorization, LDU factorization.

Module 2: EIGEN VALUES, EIGEN VECTORS AND DIAGONALIZATION (09 Periods)

Characteristic equation of a matrix, Eigenvalues and Eigenvectors of matrices, Similarity of matrices, Diagonalization by using Eigenvalues, Cayley-Hamilton Theorem (Without Proof), Evaluation of matrix powers.

Module 3: VECTOR SPACES (10 Periods)

Vector spaces, Sub-spaces, Four fundamental subspaces of a matrix, Span, Linearly independent and dependent, Basis, construction of Basis, dimensions, Finite dimensional vector space.

Module 4: LINEAR TRANSFORMATION (08 Periods)

Linear transformation, Kernel and range of linear transformation, Basic properties, Invertible linear transformation, Matrix of linear transformation.

Module 5: INNER PRODUCT SPACES (08 Periods)

Inner product, Norm, Distance, Inner product space, Orthogonal and orthonormal basis, Gram-Schmidt orthogonalization, Singular Value Decomposition for square matrices.

Total Periods: 45

EXPERIENTIAL LEARNING

1. Explain the role of Eigen values and Eigen vectors in image processing.
2. Explain how the matrix theory is used in robotics and automachines.

RESOURCES

TEXT BOOKS:

1. Peter Selinger, "Matrix theory and linear algebra", 1st Edition, creative commons license, 2018.
2. Stephen H. Friedberg, Arnold J. Insel and Lawrence E. Spence, "Linear Algebra," Pearson Education, 5th Edition, 2022.

REFERENCE BOOKS:

1. Kuladeep Sing, "Linear Algebra step by step," 1st edition, Oxford University press, 2014.
2. David Poole, "Linear Algebra: A Modern Introduction," Brooks/Cole, 2nd edition, 2005.
3. Edgar G. Goodaire, "Linear Algebra," Cambridge University Press, 1st Edition, 2014.
4. M. Thamban Nair and Arindama Singh, "Linear Algebra," Springer, 1st Edition, 2019.
5. Hans Schneider and George Phillip Barker, "Matrices and Linear Algebra," Dover Publications, 2nd Edition, 1989.
6. Jim DeFranza, Daniel Gagliardi, "Introduction to Linear algebra with applications," Mc GrawHill, 2009.

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/111106051>
2. <https://nptel.ac.in/courses/111106135>

WEB RESOURCES:

1. <https://catalogimages.wiley.com/images/db/pdf/9781119570271.excerpt.pdf>
2. [https://web.northeastern.edu/dummit/docs/linalgprac_5_eigenvalues_and_diagonalization.p df](https://web.northeastern.edu/dummit/docs/linalgprac_5_eigenvalues_and_diagonalization.pdf)
3. <https://web.auburn.edu/holmerr/2660/Textbook/vectorspace-print.pdf>
4. <https://textbooks.math.gatech.edu/ila/linear-transformations.html>
5. <https://linear.axler.net/InnerProduct.pdf>

Course Code**Course Title****L T P S C****CA25UG0102L****COMPUTER FUNDAMENTALS AND
ORGANIZATION**

3 - - - 3

Pre-Requisite -**Anti-Requisite** -**Co-Requisite** -

COURSE DESCRIPTION: This course deals with fundamentals of computers, Data Representations, input and output units, primary and secondary storage devices, programming languages, Binary Arithmetic and Logic circuits. Understand Computer architecture and organization working scenario in addition to Reduced Instruction Set Computer (RISC) platform.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate fundamentals of computer and computer organization concepts to design of modern processors, memories and I/Os.
- CO2.** Analyze the performance of commercially available computers.
- CO3.** Develop to use the computer organization techniques in digital systems.
- CO4.** Identify, formulates, and solve hardware and software computer engineering problems using sound computer engineering principles.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	3	2	3	2	3	-	2	1	1	-	2	2	3	2	3
CO2	3	3	2	3	3	-	2	2	1	1	2	2	3	2	3
CO3	3	2	3	3	3	-	2	2	2	-	2	3	3	3	3
CO4	3	3	3	3	3	-	3	2	2	1	2	3	3	3	3
Course Correlation Mapping	3	3	3	3	3	-	2	2	2	1	2	3	3	3	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO COMPUTERS (09 Periods)

Computer basics: Algorithms, simple model of a computer, characteristics of computer, problem solving using computers: flow chart, Program, working of a computer, stored program concept.

Data Representation: Representation of characters in Computers, integers, fractions, Hexadecimal representation of numbers, decimal to binary conversion, error detecting codes.

Input units: Traditional computer Input/ Output units: Keyboard, Display unit, computer mouse, other input technologies: touch pad, touch screen, magnetic Ink Character recognition (MICR), Optical Mark Reading and Recognition (OMR), Flatbed Scanner and barcode.

Module 2: OUTPUT UNITS AND COMPUTER MEMORY (09 Periods)

Computer Output Devices: Flat panel display technology, E-ink display, printers, inkjet printers, dot matrix printers, line printers, plotters, choosing printer.

Computer Memory: Memory cell, Memory Organization, Read Only Memory, Serial Access Memory, Physical Devices used to Construct Memories, magnetic hard disk, Compact disk read only memory, Magnetic tape drive and memory hierarchy.

Processor: Structure of Instructions, Description of Processor, Machine Language Program, Algorithm to Simulate the Hypothetical Computer, Enhancing Hypcom.

Module 3: BINARY ARITHMETIC AND LOGIC CIRCUIT (09 Periods)

Binary Arithmetic: Binary addition, subtraction, Signed Numbers, Two's complement representations of Numbers, additions/subtractions of Numbers in 2's complement Notation, binary multiplications, Binary division, floating point representation of numbers.

Logic circuit: Switching Circuit, AND/OR Operations, NOT Operation, Boolean functions, Postulates, duality Principle, theorems, Precedence operators, canonical Forms for Boolean Functions, Logic circuits, Parallel and Serial adders and Physical devices used to construct gates.

Module 4: COMPUTER ORGANIZATION AND CENTRAL PROCESSING UNIT ORGANIZATION (09 Periods)

Computer Organization and Design: Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs. RISC.

Central Processing Unit organization: General Register Organization, Stack organization, Instruction formats, Addressing modes, Data Transfer and Manipulation, Program Control, CISC and RISC processors.

Module 5: MEMORY ORGANIZATION AND INPUT/OUTPUT ORGANIZATION (09 Periods)

Memory Organization: Semiconductor Memory Technologies, Memory hierarchy, Interleaving, Main Memory-RAM and ROM chips, Address map, Associative memory-Hardware organization. Match logic. Cache memory-size vs. block size, Mapping functions-Associate, Direct, Set Associative mapping. Replacement algorithms, write policies, Auxiliary memoryMagnetic.

Total Periods: 45

EXPERIENTIAL LEARNING

1. a) Understand the Role of Binary arithmetic in digital computers and perform the following operations.
 - i. Binary Addition
 - ii. Binary Subtraction
 - iii. Binary Multiplication
 - iv. Binary Division
2. b) How to represent the following expression into Logic gates and Truth table?

Boolean Expression

$$Q = \overline{(A.B)} . \overline{(A+B)} . C$$
3. c) Illustrate the organization of i5 Processors.

RESOURCES:

TEXT BOOKS:

1. V. Rajaraman, Neeharika Adabala, "Fundamentals of Computers", PHI Learning Private Ltd, 6th Edition, 2015.
2. M. Moris Mano, "Computer System Architecture", 3rd edition, Pearson/PHI, India, 2006.

REFERENCE BOOKS:

1. Priti Sinha, Pradeep K., Sinha "Computer Fundamentals: Concepts, Systems & Applications", 8th Edition, BPB Publications.
2. William Stallings, "Computer Organization and Architecture- designing for performance", 8th edition, Prentice Hall, New Jersey, 2010.

VIDEO LECTURES:

1. https://onlinecourses.swayam2.ac.in/cec19_cs06/preview
2. https://onlinecourses.nptel.ac.in/noc21_cs61/preview

WEB RESOURCES:

1. <https://www.electrical4u.com/binary-arithmetic/>
2. <https://www.youtube.com/watch?v=JQBRzsPhw2w>
3. <https://www.techtarget.com/whatis/definition/logic-gate-AND-OR-XOR-NOT-NAND-NOR-and-XNOR>

Course Code	Course Title	L	T	P	S	C
CA25UG0201L	DATABASE MANAGEMENT SYSTEMS	3	-	2	-	4
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This Course provides theoretical concepts and hands-on experience on Database systems, Database design, Relational model, Relational algebra, SQL queries, Constraints and triggers, PL/SQL, Schema refinement and normal forms, Transaction management, Concurrency control, Overview of storage and indexing.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Analyze and apply the concepts of ER-modelling and normalization to design viable data models for a given problem.
- CO2.** Formulate relational database schemas, apply suitable integrity constraints, for querying databases.
- CO3.** Use SQL to store, query, and manipulate data in relational databases.
- CO4.** Develop PL/SQL blocks to centralize database applications for maintainability and reusability.
- CO5.** Analyze transaction processing, concurrency control and storage methods for database management.
- CO6.** Work Independently and Communicate Effectively in Oral and Written forms.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO4	PO 5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	-	-	-	-	-	-	-	-	-	2	3	-
CO2	1	3	2	2	3	-	-	-	-	-	-	-	2	3	-
CO3	1	2	3	2	3	-	-	-	-	-	-	-	2	3	-
CO4	2	3	3	3	3	2	-	-	-	-	-	-	3	2	-
CO5	3	3	-	-	-	-	-	-	-	-	-	-	2	3	-
CO6	-	-	-	-	-	-	-	-	3	3	-	-	-	-	-
Course Correlation Mapping	2	3	3	2	3	1	-	-	3	3	-	-	2	3	-

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO DATABASE SYSTEMS

(08 Periods)

Introduction to Database : Advantage of Database System, File ,systems vs. DBMS View of Data, When not to use a DBMS, Database System Concepts and Architecture, When not to use a DBMS, The database system Environment, Classification of DBMS. Data Models, Schemas, Instances and schemas, Database languages and interfaces, The database system Environment, Different people behind DBMS.

Module 2: ENTITY-RELATIONSHIP MODEL

(08 Periods)

Entity-Relationship Model: Basic Concepts, Constraints, Keys: Primary Key, Super key, Candidate key, Entity Types, Entity Sets, Design issues, Entity-Relationship Diagram, Relations, Relationship types, Roles and Structural Constraints, Weak Entity sets, Extended ER Features, Design of E-R Database Schema, Reduction of an E-R Schema to tables.

Module 3: RELATIONAL MODEL AND CONSTRAINTS

(09 Periods)

Relational Model and Constraints: Relational model Concepts, Structure of Relational Databases, Constraints: Entity integrity, Referential Integrity, Domain Constraints, Assertions, Triggers, Security and Authorization, Authentication and Encryption.

Module 4: RELATIONAL DATABASE LANGUAGES

(10 Periods)

SQL AND PL/SQL: Data definition in SQL, Queries in SQL, Insert, Delete and Update Statements in SQL, Views in SQL, Specifying General Constraints as Assertions, specifying indexes, Embedded SQL. PL /SQL: Introduction.

Module 5: THE RELATIONAL ALGEBRA AND TRANSACTION MANAGEMENT (10 Periods) AND RECOVERY TECHNIQUES

The Relational Algebra: Tuple Relational Calculus, Data Normalization: Functional dependencies, Normal form concepts up to 3rd Normal form.

Transaction Management and Recovery Techniques: Introduction to Transaction Processing, Transaction Concepts and Properties, Schedules, Serializability of Schedules, Conflict and view serializable schedules, Recovery Concepts, Recovery from Transactions, Introduction to Concurrency Control Techniques.

Total Periods: 45

EXPERIENTIAL LEARNING

- 1) Design and analyze an ER-Model for the following use case.

Roadway Travels is in business since 1977 with several buses connecting different places in India. Its main office is in Hyderabad. The company wants to computerize its operations in the following areas:

- a) Reservations
- b) Ticketing
- c) Cancellations

Reservations:

Reservations are directly handled by booking office. Reservations can be made 60 days in advance in either cash or credit. In case the ticket is not available, a wait-listed ticket is issued to the customer. This ticket is confirmed against the cancellation.

Cancellation and Modification:

Cancellations are also directly handled at the booking office. Cancellation charges will be charged. Wait-listed tickets that do not get confirmed are fully refunded.

- 2) a) Implement Single Row functions-Character, Numeric and Date functions.
- b) Implement Data Definition Language commands>Create, Alter, Drop, Truncate, and Rename.
- c) Implement Data Manipulation Language commands-Insert, Select, Update, and Delete.
- 3) Implement various types of integrity constraints-NOT NULL constraint, DEFAULT constraint, UNIQUE constraint, PRIMARY key, FOREIGN key, CHECK constraint.
- 4) a) Implement group functions with different operators such as aggregate operators, group by, having and order by.
- b) Implement nested and correlated nested queries using set operators and set comparison operators.
- 5) a) Creation of views, synonyms, sequence, indexes and save point
- b) Implement various types of joins-outer join and inner join.

Basic PL/SQL:

- 6) Construct PL/SQ block for the following:
- a) To determine whether a number is palindrome
 - b) To determine whether a number is an Armstrong number
 - c) To find greatest of three numbers
 - d) To display Fibonacci series

Control Structures:

- 7) a) Write a programming PL/SQL to update the salary of a specific employee by 8% if the salary exceeds the mid-range of the salary against this job and update up to mid-range if the salary is less than the mid-range of the salary, and display a suitable message.
- b) Write a PL/SQL program to display the description against a student's grade using CASE statement.

Exception Handling:

- 8) a) Develop a PL/SQL program that displays the name and address of a student whose ID is given. If there is no student with the given student ID in the database, the program should raise a run-time exception NO_DATA_FOUND, which should be captured in the EXCEPTION block.
- b) Construct the user-defined exceptions to get the salary of an employee and check it with the job's salary range. If the salary is below the range, raise an exception BELOW_SALARY_RANGE. If the salary is above the range, raise the exception ABOVE_SALARY_RANGE.

Functions:

- 9) a) Write a function that accepts two numbers A and B and performs the following operations.
- Addition
 - Subtraction
 - Multiplication
 - Division
- b) Write a PL/SQL block that updates salary of an employee in Employee table by using in cr function which takes employee number as argument and calculates increment and returns increment based on the following criteria.
- If salary <= 3000, increment = 30% of salary
 - If salary > 3000 and <= 6000, increment = 20% of salary
 - Else increment = 10% of salary

Procedures:

- 10) a) Write a procedure that accepts two numbers and displays their sum
- b) Write procedures to demonstrate IN, IN OUT and OUT parameters

Cursors:

- 11) a) Write a block in PL/SQL to create a Cursor that displays the employee name and number of jobs he/she has done in the past.
- b) Write a program in PL/SQL to create a cursor to display the name and salary of each employee in the EMPLOYEES table whose salary is less than that' specified by a passed-in parameter value.

Triggers:

- 12) Develop a suitable student database application by considering appropriate attributes. Couple of attributes to be maintained is the attendance of a student in each subject for which he/she has enrolled and internal assessment Using TRIGGERS for the following:
- Whenever the attendance is updated, check if the attendance is less than 85%; if so, notify the concerned head of the department.
 - Whenever, the marks in an internal assessment test are entered, check if the marks are less than 40%; if so, notify the concerned head of the department.

RESOURCES**TEXT BOOKS:**

- Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems," McGrawHill, 3rd Edition, 2014.
- Abraham Silberschatz, Henry. F. Korth, S. Sudarshan, "Database System Concepts," McGrawHill, 7th edition, 2019.

REFERENCE BOOKS:

- Ivan Bayross, SQL, PL/SQL: "The Programming Language of Oracle," BPB publications, 4th Edition, 2017.
- Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson, 2015.
- Thomas Connolly, Carolyn Begg, "Database Systems," Pearson, 6th edition, 2019.
- S.K. Singh, "Database System Concepts, Design and Applications," 1st Edition, Pearson Education, 2006.
- Satish Ansani, "Oracle Database 11g: Hands-on SQL and PL/SQL," PHI, 2010.
- Dr. Rajiv Chopra, "Database Management Systems," Sultan Chand, 2016.
- Pranab Kumar Das Gupta, P. Radha Krishna, "Database Management System Oracle SQL and PL/SQL," PHI, 2nd Edition, 2009.

VIDEO LECTURES:

- https://swayam.gov.in/nd1_noc19_cs46/preview
- <https://nptel.ac.in/courses/106105175>
- https://onlinecourses.nptel.ac.in/noc21_cs04/preview
- <https://www.youtube.com/watch?v=MDQxqYVXiVU>
- <https://www.youtube.com/watch?v=c5HAWKX-suM>

WEB RESOURCES:

- <https://www.classcentral.com/course/swayam-introduction-to-database-systems-17660>
- <https://www.scaler.com/topics/dbms/>
- https://www.academia.edu/27988617/Database_Management_System_DBMS_Tutorial
- <https://nptel.ac.in/courses/106104135>
- <https://downloads.mysql.com/docs/mysql-tutorial-excerpt-5.7-en.pdf>
- https://docs.oracle.com/cd/E11882_01/server.112/e40540/intro.htm#CNCPT88786

Course Code	Course Title	L	T	P	S	C
CA25UG0202L	PROGRAMMING FOR PROBLEM SOLVING	3	-	2	-	4
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course provides a detailed discussion and hands-on experience on C Programming concepts, Operators and Expressions, Input and Output Functions, Control Structures, Problem Solving Aspects, Arrays and Strings, Functions, Pointers, Structures and Unions and File Handling.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Design algorithms using problem-solving techniques for given problems.
- CO2.** Apply functions and Arrays to enhance reusability and data manipulation.
- CO3.** Develop programs using pointers for efficient memory management.
- CO4.** Apply structures, unions and file handling concepts to develop societal applications.
- CO5.** Work independently or in team to solve problems with effective communications.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	3	3	3	3	2	-	2	1	1	-	2	2	3	2	3
CO2	3	3	3	2	2	-	2	1	1	-	2	2	3	2	3
CO3	3	3	3	3	3	-	2	2	1	-	2	2	3	3	3
CO4	3	3	3	3	2	-	2	2	2	1	2	3	3	3	3
CO5	2	2	2	2	2	-	3	2	3	1	3	2	2	2	2
Course Correlation Mapping	3	3	3	3	2	-	2	2	2	1	2	2	3	2	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO C PROGRAMMING (09 Periods)

Basics of C Programming: Introduction, Structure of a C program, Concept of a variable,

Data types in C, Program statement, Declaration, Storing the data in memory, Tokens, Operators and expressions, Lvalues and Rvalues, Type conversion in C.

Input and Output: Basic screen and keyboard I/O in C, Non-formatted input and output, formatted input and output functions.

Module 2: CONTROL STATEMENTS AND INTRODUCTION TO PROBLEM SOLVING (08 Periods)

Control Statements: Specifying test condition for selection and iteration, Writing test expression, Conditional execution and selection, Iteration and repetitive execution, goto statement, Special control statements, Nested loops.

Introduction to Problem Solving: Algorithms, Flowcharts, Problem solving aspect, Top-down design, Implementation of algorithms, program verification and efficiency of algorithms.

Module 3: ARRAYS & STRINGS AND FUNCTIONS (10 Periods)

Arrays and Strings: One-dimensional array – Declaration, Initialization, Accessing elements, operations; Multi-dimensional arrays – Declaration, Initialization, Working with 2D arrays; Strings – Declaration, Initialization, Printing strings, String input, Character manipulation, String manipulation; Arrays of strings – Initialization, manipulating string arrays.

Functions: Concept of function, Using functions, Call by value mechanism, working with functions, passing arrays to functions, Scope and extent, Storage classes, Recursion.

Module 4: POINTERS (08 Periods)

Introduction to Pointers: Understanding memory addresses, Address operator (&), Pointer – declaration, Initialization, Indirection operator and dereferencing, Void and Null pointers, Use of pointers, Arrays and pointers, Pointers and strings, Pointer arithmetic, Pointers to pointers, Array of pointers, Pointers to an array, Two-dimensional arrays and pointers, Pointers to functions, Dynamic memory allocation.

Module 5: USER-DEFINED DATA TYPES AND FILES (10 Periods)

User-Defined Data Types: Structures - Declaration, Accessing the members, Initialization, typedef and its use, Arrays of structures, Arrays within structure, Structures and pointers, Structures and functions; Unions, Enumeration types, Bitfields.

Files: Using files in C, Working with text and binary files, Direct File Input and Output, Files of records, Random access to files of records.

Total Periods: 45

EXPERIENTIAL LEARNING

1.
 - a) Write a C program to perform the arithmetic operations on two integer numbers.
 - b) Write a program to evaluate the following expressions by reading the necessary values from the keyboard.
 - i. $(ax + b)/(ax - b)$
 - ii. $2.5 \log x + \cos 32^\circ + |x^2 + y^2|$
 - iii. $ax^5 + bx^3 + c$
 - iv. ae^{kt}
2.
 - a) Write a C program to find the roots of a quadratic equation.
 - b) In a town, the percentage of men is 52. The percentage of total literacy is 48 and the total percentage of literate men is 35 of the total population. Write a C program to find the total number of illiterate men and women if the population of the town is 7000.
3.
 - a) Write a C Program to compute an electricity bill based on the following slab rates.

Consumption units	Rate (in Rupees/unit)
-------------------	-----------------------

0-100	4.0
101-150	4.6
151-200	5.2
201-300	6.3
Above 300	8.0

(**Hint:** Take current and old meter readings from the user to get consumption units)

- b) An insurance company computes the premium amount based on the following;
 - i. If a person's health is excellent and the person is between 25 and 35 years of age and lives in a city, and is a male then the premium is Rs.4 per thousand and the policy amount cannot exceed Rs.2 lakhs.
 - ii. If a person satisfies all the above conditions and is female then the premium is Rs.3 per thousand and the policy amount cannot exceed Rs.1 lakh.
 - iii. If a person's health is poor and the person is between 25 and 35 years of age and lives in a village and is a male then premium is Rs.6 per thousand and the policy cannot exceed Rs. 10000.
 - iv. In all other cases the person is not insured.
Write a C program to determine whether the person should be insured or not, his/her premium rate and maximum amount for which he/she can be insured.
- c) Write a C Program to find the grade for a student using a Switch case. The user needs to enter a subject score (varies from 0 to 100) and then display the grade as described below.

Score	Grade	Score	Grade
$> = 90$	O	$> = 50$ to < 60	D
$> = 80$ to < 90	A	$> = 40$ to < 50	E
$> = 70$ to < 80	B	< 40	Fail
$> = 60$ to < 70	C		

4. a) A Fibonacci sequence is defined as follows:

The first and second terms in the sequence are 0 and 1. Sub-sequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.

- b) Write a C program to find the sum of individual digits of a positive integer.

- c) Write a C program to read two numbers x and n, and then compute the sum of the geometric progression: $1+x+x^2+x^3+\dots+x^n$. Show appropriate error message for $n < 0$.

(Example: if n is 3 and x is 5, then the sum is: $1+5+25+125$)

- d) Write a C program to print the following pattern.

```

                                     1
                                1   2   1
                           1   2   3   2   1
                      1   2   3   4   3   2   1
                 1   2   3   4   5   4   3   2   1
```

5. a) Write a C program to find both the largest and smallest numbers in a list of integers.

- b) Write a C program that uses function to perform the following:

i) Addition of Two Matrices ii) Multiplication of Two Matrices

6. a) Write a C program to insert a sub-string in to a main string at a given position.

- b) Write a C program to count the lines, words and characters in a given text.

7. a) Write a C program to generate all the prime numbers between 1 and n, where n is

8. a) value entered by the user. Define a separate function to generate prime numbers.

- b) Write C program that uses recursive function to find
i) Factorial of a given integer ii) GCD of two given integers

9. a) Write a C program to print the elements of an array in reverse order using pointers.

- b) Write a C program to count the number of vowels and consonants in a string using pointers.

- c) Write a C program to store n elements in an array and print the elements in sorted order using pointers.

10. a) Write a C program that performs the following operations:
- i. Reading a complex number ii. Writing a complex number
 - iii. Addition of two complex numbers iv. Multiplication of two complex numbers (**Note:** Represent complex number using a structure.)
- b) Define a structure to store employee details include Employee-Number, Employee-Name, Basic-pay, Date-of-Joining. Write a C program for the following.
- i) A function to store 10 employee details.
 - ii) A function to implement the following rules while revising the basic pay. If Basic-pay $\leq$ Rs.5000 then increase it by 15%.
If Basic-pay $>$ Rs.5000 and $\leq$ Rs.25000 then it increase by 10%. If Basic-pay $>$ Rs.25000 then there is no change in Basic-pay.
 - iii) A function to print the details of employees who have completed 20 years of service from the Date-of-Joining.
11. a) Write a C program to reverse the first n characters of a given text file.
- b) Write a C program to merge two files into a new file.
12. a) Develop a phone book application to save users contact information include name, mobile number and email id as well as to edit and delete contact details.

RESOURCES

TEXT BOOKS:

1. PradipDey and Manas Ghosh, "Programming in C," Oxford University Press, New Delhi, 2ndEdition, 2013.
2. R. G. Dromey, "How to Solve it by Computer," Pearson Education, 1stEdition, 2013.

REFERENCE BOOKS:

1. Byron S Gottfried and Jitender Kumar Chhabra, "Programming with C," 4thEdition, McGraw Hill Education, 2019.
2. YashavantKanetkar, "Let Us C," 5thh Edition, BPB Publications, 2017.
3. E. Balagurusamy, "Programming in C," McGraw Hill Education Pvt, Ltd, New Delhi, 7thEdition, 2017.
4. Behrouz A. Forouzan and Richard F. Gilberg, "Computer Science: A Structured Programming Approach Using C," Cengage Learning, 3rd Edition, 2008.

SOFTWARE/TOOLS:

1. Software: Turbo C++/Dev C++

VIDEO LECTURES:

1. <https://www.digimat.in/nptel/courses/video/106105171/L03.html>
2. <https://nptel.ac.in/courses/106104128>

WEB RESOURCES:

1. Learn C Programming - <https://www.programiz.com/c-programming>
2. Learn C Programming - <https://www.tutorialspoint.com/cprogramming/index.htm>
3. C Programming Exercises, Practice, Solution - <https://www.w3resource.com/c-programming-exercises/>
4. Basic programming exercises and solutions in C - <https://codeforwin.org/2015/05/basic-programming-practice-problems.html>
5. C Programming Exercises, Practice, Solution - <https://www.w3resource.com/c-programming-exercises/>
6. Basic programming exercises and solutions in C - <https://codeforwin.org/2015/05/basic-programming-practice-problems.html>

Course Code	COURSE TITLE	L	T	P	S	C
CA25UG0103L	INNOVATION AND DESIGN THINKING	2	-	-	-	2
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course introduces design thinking in product innovation. This course is intended to familiarize product design process, introduce the basics of design thinking, to bring awareness on idea generation, to familiarize the role of design thinking in services design.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- C01.** Demonstrate fundamental knowledge of Design thinking.
- C02.** Analyze characteristics of successful product development
- C03.** Demonstrate knowledge on Idea generation techniques
- C04.** Analyze the process of design thinking for services.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PSO2	PSO2	PSO2	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C01	2	2	2	2	1	2	2	2	2	1	2	3	3	-	-
C02	2	3	3	3	2	2	2	3	2	1	2	3	3	3	3
C03	1	2	2	2	1	2	2	2	2	1	2	3	3	3	-
C04	2	3	3	3	2	2	2	3	2	1	2	3	3	3	-
Course Correlation Mapping	2	3	3	3	2	2	2	3	2	1	2	3	3	3	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO DESIGN

(06 Periods)

Introduction to design, characteristics of successful product development, product development process, identification of opportunities, product planning, Innovation in product development.

Module 2: DESIGNING THINKING

(06Periods)

Design Thinking: Introduction, Principles, the process, Innovation in Design Thinking, benefits of Design thinking, design thinking and innovation, case studies.

Module 3: IDEA GENERATION

(06 Periods)

Idea generation: Introduction, techniques, Conventional methods, Intuitive methods, Brainstorming, Gallery method, Delphi method, Synectics etc., case studies

Module 4: DESIGN THINKING IN INFORMATION TECHNOLOGY

(06 Periods)

Design Thinking in Business process model, Design Thinking for agile software development, virtual collaboration, multi user and multi account interaction, need for communication, TILES toolkit, Cloud implementation.

Module 5: DESIGN THINKING FOR SERVICE DESIGN

(06 Periods)

How to design a service, Principles of service design, Benefits of service design, Service blueprint, Design strategy, organization, principles for information design.

Total Periods: 30

EXPERIENTIAL LEARNING

1. What data-driven insights can help us evolve our product?
2. How can we gather ongoing feedback to drive future enhancements?

RESOURCES

TEXT BOOKS:

1. Lagerfeld Kilian., "Design Thinking for Beginners", 1st Edition, Personal Growth Hackers, 2020.
2. PavanSoni, "Design Your Thinking: The Mindsets, Toolsets and Skill Sets for Creative Problem- Solving", 2nd Edition, Penguin Random House India Private Limited, 2020.
3. Marc stickdorn and Jacob Schneider, This is Service Design Thinking, Wiely, 2011

REFERENCE BOOKS:

1. Walter Brenner and Falk Uebernickel, "Design Thinking for Innovation Research", 1stEdition, Springer, 2016.
2. GK Van Patter and Garry K. VanPatte, "Rethinking Design Thinking: Making Sense of the Future", 1stEdition, Humantific Publishing, 2020.

VIDEO LECTURES:

1. <https://www.youtube.com/watch?v=4nTh3AP6knM>
2. <https://www.youtube.com/watch?v=NRQLznGIKk>

WEB RESOURCES

1. <https://web.stanford.edu/~mshanks/MichaelShanks/files/509554.pdf>
2. https://www.tutorialspoint.com/hi/design_thinking/design_thinking_tutorial.pdf
3. <http://www.algarytm.com/it-executives-guide-to-design-thinking:e-book>

Course Code	Course Title	L	T	P	S	C
CA25UG0104L	PROFESSIONAL ETHICS AND HUMAN VALUES	2	-	-	-	2
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course deals with personal conviction, and ethics and describes the accepted principles and standards of conduct regarding moral duties and virtues as applied to an organization. Codes of professional ethics guide the stakeholders of an organization about the desirable and undesirable acts related to the profession.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate the principles of ethics, professional values, and social responsibility.
- CO2.** Analyze the problems in the implementation of moral autonomy and use ethical theories in resolving moral dilemmas.
- CO3.** Develop suitable strategies to resolve problems that arise in practicing professional ethics and Industrial standards.
- CO4.** Function as a member, consultant, manager, advisor and leader in multi-disciplinary teams.
- CO5.** Provide solutions to complex problems associated with professional ethics using analysis and interpretation.

CO-PO Mapping Table:

Course Outcomes	Program Outcomes											Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	-	-	-	-	-	3	2	3	-	3	-	1	-
CO2	2	-	-	-	-	-	3	2	3	-	3	1	-	-
CO3	2	-	-	-	-	-	3	2	3	-	3	-	-	-
CO4	2	-	-	-	-	-	3	2	3	-	3	-	1	
CO5	2	-	-	-	-	-	3	2	3	-	3	-	-	2
Course Correlation Mapping	2	-	-	-	-	-	3	2	3	-	3	2	3	1

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: PROFESSIONAL ETHICS

(06 Periods)

Scope and significance of ethics, varieties of moral issues, types of ethical inquiry, moral dilemmas, moral autonomy — Kohlberg's theory; Gilligan's theory of moral development.

Module2: PROFESSIONAL IDEALS AND VIRTUES

(06 Periods)

Theories of virtues and professional ideals; professional responsibility and accountability; self-respect and integrity; role of customs and religion in ethics; religion and divine command theory; resolving moral dilemmas; moral leadership.

Module 3: SOCIAL EXPERIMENTATION

(06 Periods)

Social experimentation and its similarities to standard scientific experiments; learning from the past and knowledge gained; responsibilities of engineers as experimenters; the Challenger's case study; role of industrial standards; issues and limitations in engineering law

Module 4: RESPONSIBILITIES AND RIGHTS

(06 Periods)

Collegiality and professional loyalty; respect for authority; collective bargaining and employee rights; confidentiality and trust; occupational crime and ethical misconduct; rights and responsibilities of engineers; whistle-blowing and its implications; the BART case study.

Module 5: HARMONY WITH PROFESSIONALETHICS

(06 Periods)

tance and practice of human values; ethical human conduct in personal and professional life; transition from the present state to a universal human order; role of engineers and technologists in promoting social and ecological responsibility; contribution to enriching institutions and organizations.

Total Periods: 30

EXPERIENTIAL LEARNING

1. nstrate orally using your experiences of what is naturally acceptable in a relationship – Feeling of respect or disrespect and what is naturally acceptable is to nurture or exploit others.
2. fy community partners and discuss with a community partner or organization. Prepare a report by identifying and analysing the issues or opportunities.
3. experiences may be directed to include a range of time-intensive endeavours that require varying levels of student interaction. Prepare a report on visiting a Juvenile home.
4. nts read a speech in the classroom by former United Nations Secretary-General Kofi Annan on human values.
5. nts are encouraged to bring a daily newspaper to class or to access any news related to the need for human values and note down the points.
6. out the relevance of engineering ethics theory and practice with relevance to current trends.
7. sional ideals and virtues are important to everyone. Prepare a case study on the professional ideals and virtue of any one of the famous sports personalities from India.
8. are the present to the past in engineering experimentations concerning the change in professionalism.

9. a study on occupational crime and the role of modern technology in finding solutions.
 10. re a case study on how to maintain harmony with different cultural people using professional ethics.
- indicative one. The course instructor may change the activities and the same shall be reflected in Course Handout)*

RESOURCES

TEXTBOOKS:

1. Alavandar, A., and S. Ramachandran. *Professional Ethics and Human Values*. CRC Press, 2022. Print.
2. Whitbeck, Caroline. *Ethics in Engineering Practice and Research*. 2nd ed., Cambridge University Press, 2021. Print.
3. Harris, Charles E., Michael Pritchard, and Ray James. *Engineering Ethics: Concepts and Cases*. 5th ed., Routledge, 2021. Print.

REFERENCE BOOKS:

1. O'Regan, Jeffrey K. *Introduction to Engineering Ethics*. Springer, 2020. Print.
2. Lobo, V. T. *Handbook of Ethics in Science and Engineering*. CRC Press, 2021. Print.
3. Badaracco Jr., R. G. *Contemporary Ethics in Engineering*. Wiley, 2020. Print.
4. R. Subramanaian, *Professional Ethics*, Oxford Higher Education, 2013.

VIDEO LECTURES:

1. <https://youtu.be/Nhadb9Nw-Bc?si=GUIeINlek1J5Uol>
2. <https://www.youtube.com/watch?v=QFH0tH54oUc>
3. <https://www.youtube.com/watch?v=JJshY11nX14>
4. <https://www.youtube.com/watch?v=TyP09S0UEzA>
5. https://www.youtube.com/watch?v=0QMwjV_ZVtc

WEB RESOURCES:

1. <https://siiet.ac.in/wp-content/uploads/2020/09/7.1.10-professional-ethics-manual.pdf>
2. <https://soaneemrana.org/onewebmedia/Professional%20Ethics%20and%20Human%20Values%20by%20R.S%20NAAGARAZAN.pdf>
3. <https://india.oup.com/productPage/5591038/7421214/9780199475070>

Course Code	Course Title	L	T	P	S	C
CA25UG0105L	PRINCIPLES OF ACCOUNTING	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION:

The course is designed so that the students can understand the scientific preparation of accounting books with relevant interpretation. The course also provides students with knowledge of preparation of final accounts.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate knowledge of basic concepts of financial accounting.
- CO2.** Analyze Cash book and reconciliation statements and acquaint oneself with preparation of statements.
- CO3.** Demonstrate knowledge on preparation of Bank Reconciliation statements.
- CO4.** Analyze and prepare financial statements for the sole proprietorship of business.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO2	PSO3
CO1	2	2	-	1	1	3	2	2	1	-	1	2	-	2	-
CO2	2	3	-	2	2	2	2	3	2	-	1	2	-	3	-
CO3	2	3	-	2	2	2	2	3	1	-	1	2	-	3	-
CO4	3	3	2	2	2	3	2	3	2	-	2	3	1	3	-
Course Correlation Mapping	2	3	2	2	2	3	2	3	2	-	2	2	1	3	-

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO ACCOUNTING (09 Periods)

Meaning – Definition – Objectives of Accounting - Scope of accounting – Accounting principles – Branches of accounting - Classification of accounts - Double Entry System - Book Keeping – Users of Accounting Information - Advantages and limitations of accounting

Module 2: PROCESS OF ACCOUNTING (09 Periods)

Meaning and Definition of Journal – Importance of Journal Entry – Proforma of Journal Entries – Journalising – Posting - Problems on Journal Entries – Ledger – Problems on Ledger accounts – Balancing of Ledger accounts

Module 3: SUBSIDIARY BOOKS (09 Periods)

Introduction to Subsidiary Books- Purchase Book, sales Book, Purchase Returns, Sales Returns, Bills Receivable, Bills Payable, Journal Paper-Preparation of Subsidiary Books- Cash Books, Three Column Cash Book– Trial Balance (Simple problems)

Module 4: BANK RECONCILIATION STATEMENT (09 Periods)

Meaning – Need - Reasons for differences between cash book and pass book balances – Favorable and over draft balances – Ascertainment of correct cash book balance (Amended Cash Book) - Preparation of Bank Reconciliation Statement. (Including problems)

Module 5: PREPARATION OF FINAL ACCOUNTS (09 Periods)

Introduction to Final Accounts - Trading account - Profit and Loss account - Balance Sheet with simple adjustments (Simple problems)

Total Periods: 45

EXPERIENTIAL LEARNING

1. Observation of Recording Business Transactions in an Audit firm.
2. Visiting a Business and knowing entire process of maintenance of Books of Accounts.

RESOURCES

TEXT BOOKS:

1. Maheswari S.N. & Maheswari S. K., "Introduction to Financial Accountancy," Vikas Publications, 2020.
2. Grewal. T.S., "Fundamentals of Accounting," Sultan Chand & Sons Pvt Ltd New Delhi, 2019.

REFERENCE BOOKS:

1. Jawahar Lal, Financial Accounting, Wheeler Publishing, New Delhi, 2021.
2. Chawla & Jain-Financial Accounting, Kalyani Publishers, New Delhi, 2008.

VIDEO LECTURES:

1. https://www.youtube.com/watch?v=tIDFXs6_Tjc
2. <https://www.youtube.com/watch?v=xWKfKcNqQAE>

WEB RESOURCES:

1. <https://icmai.in/upload/Students/Syllabus2016/Inter/Paper-5New.pdf>
2. https://ebooks.lpude.in/management/bba/term_2/DMGT104_FINANCIAL_ACCOUNTING.pdf

Course Code	Course Title	L	T	P	S	C
CA25UG0106L	NUMERICAL ANALYSIS, PROBABILITY AND STATISTICS	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course aims to provide an understanding the numerical solutions of linear and nonlinear equations, Numerical techniques of interpolation in various intervals, Numerical techniques of differentiation and Integration. The course also provides an introduction to probability and statistics, random variables, mathematical expectations, distributions, correlation and regression.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Solve equations, derivatives and integrals through numerical methods for the given data.
- CO2.** Identify the types of random variables and determine their mathematical expectation & probabilities.
- CO3.** Apply the concept of probability distributions to solve computing problems.
- CO4.** Apply the concepts of Correlation and Regression in result analysis and Business forecasting.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	3	3	-	2	2	1	-	2	2	3	2	3
CO2	3	3	2	2	2	-	2	1	1	-	1	1	3	2	2
CO3	3	3	2	3	2	-	2	2	1	-	2	2	3	3	3
CO4	3	3	2	2	2	-	2	3	2	1	2	3	3	3	2
Course Correlation Mapping	3	3	2	3	2	-	2	2	2	1	2	2	3	3	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: NUMERICAL SOLUTIONS OF EQUATIONS (09 Periods)

Solutions of Algebraic and Transcendental Equations: Bisection Method, Regula-falsi method, Newton-Raphson Method.

Systems of linear algebraic equations: Gauss elimination method, Gauss-Seidel methods, Gauss-Iteration methods.

Module 2: INTERPOLATION, NUMERICAL DIFFERENTIATION AND INTEGRATION (10 Periods)

Interpolation: Finite differences: Forward and backward differences, Newton's forward and backward interpolation formula, Divided difference: Newton's divided difference formula, Lagrange's interpolation formula.

Numerical differentiation, Integration: Numerical differentiation using forward and backward interpolation formula, Numerical integration using Trapezoidal and Simpson's rules.

Module 3: RANDOM VARIABLES (09 Periods)

Introduction to Probability, Random variables, Discrete and continuous random variables, Distribution function of random variables, Properties, Probability mass function, Probability density function, Mathematical expectation, Properties of mathematical expectation, Mean and Variance.

Module 4: PROBABILITY DISTRIBUTIONS (08 Periods)

Binomial Distribution, Poisson distribution, Normal distribution - Mean, Variance, Standard deviation.

Module 5: CORRELATION AND REGRESSION (09 Periods)

Introduction to simple correlation, Covariance, Correlation between two variables, Scatter diagram and its utility, Karl Pearson's coefficient of correlation Computation, Spearman's rank correlation coefficient, Estimation of regression lines.

Total Periods: 45

EXPERIENTIAL LEARNING

1. Solve the following the system of equations by Gauss- Jordan iterative method correct to three decimal places.
$$X + Y + 54Z = 110; 27X + 6Y - Z = ; 6X + 15Y + 2Z = 72.$$
2. If X is a normally distributed with mean 70 and standard deviation 16, find
 $P(38 \leq X \leq 46)$ (ii) $P(82 \leq X \leq 94)$ (iii) $P(62 \leq X \leq 86)$.

RESOURCES

TEXT BOOKS:

1. B. S. Grewal, Higher Engineering Mathematics, Khanna publishers, 44th edition, 2017.
2. J. Ravichandran, Probability and Statistics for Engineers, Wiley India, 1st Edition, 2019.

REFERENCE BOOKS:

1. Richard A Johnson, Miller and Freund's Probability and Statistics for Engineers, Prentice Hall of India, 8th Edition, 2011.
2. M. K. Jain, S.R.K Iyengar, R. K. Jain, Numerical Methods for Scientific & Engineering Computation, New Age International Publishers, 6th Edition, 2012.
3. G. Shankar Rao, Numerical Analysis, New Age International Publishers, 5th Edition, 2018.
4. S.S. Sastry, Introductory Methods of Numerical Analysis, Prentice Hall of India, 5th Edition 2012.

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/127106019>
2. <https://nptel.ac.in/courses/111106112>
3. <https://nptel.ac.in/courses/111105041>
4. <https://nptel.ac.in/courses/111106112>

WEB RESOURCES:

1. <https://www.pdfdrive.com/introductory-methods-of-numerical-analysis-by-ss-sastry-d148704487.html>
2. https://faculty.ksu.edu.sa/sites/default/files/probability_and_statistics_for_engineering_and_the_sciences.pdf
3. <http://brharnetc.edu.in/br/wp-content/uploads/2018/11/21.pdf>
4. <http://www.mi.sanu.ac.rs/~gvm/Teze/Numerical%20methods%20In%20Computational%20Engineering.pdf>

Course Code	Course Title	L	T	P	S	C
CA25UG0107L	BASICS OF VIRTUALIZATION AND CLOUD TECHNOLOGY	3	-	-	-	3

Pre-Requisite -

Anti-Requisite -

Co-Requisite -

COURSE DESCRIPTION: This course gives students an insight into the basics of cloud computing along with virtualization. It will provide the students basic understanding about cloud and virtualization.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate basic concepts of Virtualization and Cloud Technology of Cloud Computing.
- CO2.** Analyze desktop virtualization, providers with traditional IT service providers, Roots of cloud computing, hardware virtualization and Regularity issues for cloud technology.
- CO3.** Use Microsoft's Virtualization solutions for cloud virtualization
- CO4.** Adapt architectural influences using High-performance computing, Utility and Enterprise grid computing for the performance in Cloud environment.

CO-PO-PSO Mapping Table:

Course Outcome	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	-	2	-	1	1	2	2	3	2	2
CO2	3	3	3	3	2	-	2	2	2	2	2	2	3	2	3
CO3	3	3	3	2	3	-	2	2	2	1	2	2	3	3	3
CO4	3	3	3	3	2	-	2	2	2	2	2	3	3	3	3
Course Correlation Level	3	3	3	3	2	-	2	2	2	2	2	2	3	3	3

Correlation Levels: 3: High 2: Medium; 1: Low

COURSE CONTENT

Module 1: BASICS OF VIRTUALIZATION

(09 Periods)

Virtualization and cloud computing: Need of virtualization, cost, administration, fast deployment, reduce infrastructure cost, limitations.

Types of hardware virtualization: Full virtualization, partial virtualization, para virtualization

Desktop virtualization: Software virtualization, Memory virtualization, Storage virtualization, Data virtualization, Network virtualization

Module 2: HYPERVISORS AND VIRTUAL MACHINES

(10 Periods)

Server Virtualization: Understanding Server Virtualization, types of server virtualization, Virtual machine basics, types of virtual machines, hypervisor concepts and types.

Module 3: VIRTUALIZATION SOLUTIONS

(09 Periods)

Understanding Microsoft's Virtualization solutions: Microsoft's Infrastructure Optimization Model, Virtualization and the Infrastructure Optimization Model, Benefits of Virtualization, Achieving the Benefits of Datacenter Virtualization, Achieving the Benefits of Client Virtualization, Achieving the Benefits of Cloud Virtualization.

Module 4: BASICS OF CLOUD COMPUTING

(09 Periods)

Origins of Cloud computing, Cloud components, Essential characteristics, On-demand self-service, Broad network access, Location independent resource pooling, Rapid elasticity, Measured service, Comparing cloud providers with traditional IT service providers, Roots of cloud computing.

Module 5: CLOUD INSIGHTS

(08 Periods)

Architectural influences: High-performance computing, Utility and Enterprise grid computing; Cloud scenarios, Benefits: scalability, simplicity, vendors, security, Limitations, Sensitive information, Application development, security level of third party, security benefits, Regularity issues: Government policies.

Total Periods: 45

EXPERIENTIAL LEARNING

1. Creating Virtualized Environment using VM ware.
2. Deploying an WebApp/MobileApp in Public Cloud.
3. Invoke an operating System in an Virtualized environment.

RESOURCES

TEXT BOOKS:

1. David Marshall, Wade A. Reynolds, "Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center," Auerbach.
2. Anthony T. Velte, Toby J. Velte Robert Elsenpeter "Cloud computing a practical approach", TATA McGraw- Hill, 2010.

REFERENCE BOOKS:

1. Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate," 2008.
2. Rajkumar Buyya, James Broberg, Andrzej Goscinski, "Cloud Computing (Principles and Paradigms)", John Wiley & Sons, Inc. 2011
3. Judith Hurwitz , Robin Bloor , Marcia Kaufman , Fern Halper, "Cloud computing for Dummies", Wiley Publishing Inc., 2010

VIDEO LECTURES:

1. <https://www.youtube.com/watch?v=Jph3H1wZTKM>
2. <https://www.digimat.in/nptel/courses/video/106105167/L07.html>

WEB RESOURCES:

1. <https://www.knowledgehut.com/blog/cloud-computing/virtualization-in-cloud-computing>
2. <https://www.analyticssteps.com/blogs/what-virtualization-cloud-computing-characteristics-benefits>
3. <https://www.javatpoint.com/virtualization-in-cloud-computing>

Course Code	Course Title	L	T	P	S	C
CA25UG0108L	COMPUTER COMMUNICATION	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course covers basic concepts of data communications, data and signals in Computer communication. Layered Tasks, The OSI Model, TCP/IP Protocol Suite and addressing are used for network modelling. Analog and Digital Signals are used for transmitting of data without attenuation, Distortion and Noise. Multiplexing and Spread spectrum for reliable bandwidth utilization along with IEEE standards and protocols.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate concepts of data communications, data and signals, network models and protocols & standards in Computer communication.
- CO2.** Apply digital-to-digital conversion, analog-to-digital conversion, digital-to-analog conversion, analog-to-analog conversion for reliable data communication.
- CO3.** Analyze network models, analog and digital signals, periodic analog signals, digital signals, digital transmission, transmission modes and analog transmission for transmitting the data over Communication channel.
- CO4.** Use multiplexing for bandwidth utilization, protocols & standards and Wired LANs(Ethernet) IEEE standards in Computer communication in compliance with communication standards

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	-	-	-	-	-	-	-	-	-	-	-	3
CO2	3	2	3	-	3	-	-	-	-	-	-	-	-	-	3
CO3	3	3	3	-	3	-	-	-	-	-	-	-	-	-	3
CO4	3	2	2	-	3	3	-	-	-	-	-	-	-	-	3
Course Correlation Mapping	3	3	3	-	3	3	-	-	-	-	-	-	-	-	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: DATA COMMUNICATIONS AND PROTOCOLS & STANDARDS (09 Periods)

Concepts of Data Communications: Components, Data Representation, Dataflow

Networks: Distributed Processing, Network Criteria, Physical Structures, Network Models, Categories of Networks, Interconnection of Networks.

PROTOCOLS AND STANDARDS: Protocols, Standards, Standards Organizations, Internet Standards.

Module 2: NETWORK MODELS (09 Periods)

Layered Tasks: Sender, Receiver, Carrier, Hierarchy.

The OSI Model: Layered Architecture, Peer-to-Peer Processes, Encapsulation.

TCP/IP Protocol Suite: Physical and Data Link Layers, Network Layer, Transport Layer, Application Layer.

Addressing: Physical Addresses, Logical Addresses, Port Addresses, Specific Addresses.

Module 3: DATA AND SIGNALS (09 Periods)

Analog and Digital: Analog and Digital Data, Analog and Digital Signals, Periodic and Non-periodic Signals.

Periodic Analog Signals: Sine Wave, Phase, Wavelength, Time and Frequency Domains, Composite Signals, Bandwidth.

Digital Signals: Bit Rate, Bit Length, Digital Signal as a Composite Analog Signal, Transmission of Digital Signals.

Transmission Impairment: Attenuation, Distortion, Noise.

Module 4: DIGITAL TRANSMISSION (10 Periods)

Digital-To-Digital Conversion: Line Coding, Line Coding Schemes, Block Coding, Scrambling.

Analog-To-Digital Conversion: Pulse Code Modulation (PCM), Delta Modulation(DM).

Transmission Modes: Parallel Transmission, Serial Transmission, Analog Transmission.

Digital-To-Analog Conversion: Aspects of Digital-to-Analog Conversion, Amplitude Shift Keying, Frequency Shift Keying, Phase Shift Keying, Quadrature Amplitude Modulation.

Analog-To-Analog Conversion: Amplitude Modulation, Frequency Modulation, Phase Modulation.

Module 5: BANDWIDTH UTILIZATION: MULTIPLEXING AND SPREADING (08 Periods)

Multiplexing: Frequency-Division Multiplexing, Wavelength-Division Multiplexing, Time-Division Multiplexing

Wired LANs(Ethernet):IEEE STANDARDS :Data Link Layer,Physical Layer; STANDARD ETHERNET : MAC Sublayer , Physical Layer; Changes in the standard :Bridged Ethernet , Switched Ethernet , Full-Duplex Ethernet

Total Periods:45

EXPERIENTIAL LEARNING

1. Configure Peer - to - peer network with at least three hosts.
2. Create desired standard network cable including Cross cable and test it by using cable tester.
3. Establishing Connection of Computers Using Wireless Media

RESOURCES

TEXT BOOK:

1. Behrouz A. Forouzan, "Data Communications and Networking", 5th Edition, Tata McGraw-Hill, 2013.

REFERENCE BOOKS:

1. William Stallings, "Data and Computer Communications", 10th Edition, Prentice-Hall, 2013.
2. Alberto Leon-Garcia and Indra Widjaja, "Communication Networks - Fundamental Concepts and Key architectures," 2nd Edition, Tata McGraw-Hill, 2004.
3. William Stallings, "Data and Computer Communication," 8th Edition, Pearson Education, 2007.
4. Larry L. Peterson and Bruce S. Davie, "Computer Networks – A Systems Approach," 4th Edition, Elsevier, 2007.
5. Nader F. Mir, "Computer and Communication Networks," Pearson Education, 2007.

SOFTWARE/TOOLS:

1. UTP/STP cable.
2. Connector (Mainly RJ45 connector)
3. Networks toolkit (Mainly Crimping tool.)
4. Line tester or cable tester.

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/106105082> : Layered Architectures
2. <https://www.digimat.in/nptel/courses/video/106105082/L35.html>
3. <https://nptel.ac.in/courses/106105082> : Data and Signals
4. <https://nptel.ac.in/courses/106105082> : Digital Signal
5. <https://nptel.ac.in/courses/106105082> : Analog Signals
6. <https://www.digimat.in/nptel/courses/video/106105082/L28.html>

WEB RESOURCES:

1. <https://ncert.nic.in/textbook/pdf/lecs111.pdf>
2. <http://eti2506.elimu.net/Introduction/Books/Data%20Communications%20and%20Networking%20By%20Behrouz%20A.Forouzan.pdf>

Course Code	Course Title	L	T	P	S	C
CA25UG0203L	DATA STRUCTURES	3	-	2	-	4
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course provides a detailed discussion and hands-on experience on data structure concepts like arrays, stack, queues, trees, graphs and real-time applications of data structures. This course also examines algorithms for sorting, searching graphs, hashing.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Analyze and develop solutions using linear data structures such as arrays, linked lists for efficient data organization and manipulation.
- CO2.** Analyze, implement and conduct investigations on data structures such as trees, graphs, hash tables for efficient search and retrieval of data.
- CO3.** Select and apply appropriate techniques for searching and sorting problems.
- CO4.** Apply knowledge to select appropriate data structures for modelling information in data.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	3	3	-	-	-	-	-	-	-	2	-	3
CO2	3	3	3	3	3	-	-	-	-	-	-	-	2	-	3
CO3	3	2	3	3	3	-	-	-	-	-	-	-	2	-	3
CO4	3	3	3	3	3	-	-	-	-	-	-	-	2	-	3
Course Correlation Mapping	3	3	3	3	3	-	-	-	-	-	-	-	2	-	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: LINKED LISTS

(07 Periods)

Overview of data structures and algorithms, Linear and non-linear data structures, Big O notation, Linked lists Definition, Operations; Single linked lists, Circular linked lists, Doubly linked lists, Sorted lists, Linked lists efficiency, Applications of linked lists.

Module 2: STACKS AND QUEUES

(09 Periods)

Stacks: Definition, Operations, Implementation using arrays and linked lists, Applications Reversing a word, Delimiter matching, Parsing arithmetic expressions.

Queues: Definition, Operations, Applications, Implementation using arrays and linked lists, Circular queue, Double-ended queues, Priority queues.

Module 3: BINARY TREES AND SEARCH TREES

(11 Periods)

Tree terminology, Binary trees, Trees represented as arrays, Binary search trees - Concepts, Advantages, Operations, Finding maximum and minimum values, Efficiency; Balanced and unbalanced trees, AVL search trees Concepts, Operations; Red-Black trees Concepts, Rotations, Inserting a node, Efficiency.

Module 4: SEARCHING AND SORTING

(09 Periods)

Searching: Linear search, Binary search.

Sorting: Bubble sort, Selection sort, Insertion sort, Sorting objects, Shell sort, Partitioning, Quick sort, Merge sort, Heap sort, radix sort, count sort, enumeration sort.

Module 5: GRAPHS AND HASHING

(09 Periods)

Graphs: Concepts, Representation, Operations, Depth-first search, Breadth-first search, Minimum spanning trees.

Hashing: Introduction, Open addressing, Separate chaining, Characteristics of good hash functions - Quick computation, Random and Non-random keys, Folding; Hashing efficiency.

Total Periods: 45

EXPERIENTIAL LEARNING

1. A college has N number of students and the following details of all the students are maintained – register number, name, branch, phone number. Write a program to store the details of the students using a singly linked list. Develop functions to perform the following operations on the data.
 - a) Insert new student's details
 - b) Display the details of the students
 - c) Display the total number of students
 - d) Delete a given student's information

2. Department of CSE has readers club named 'Aalochana'. Students can be granted membership in readers club on their request. Similarly, one may cancel their membership of the club. Members of the club can rent books from the club. Write a program to create data structure to maintain readers club members information (Hall ticket number, name) using singly linked list. In singly linked list, the header node should store details of head of readers club and last node should store details of in-charge of readers club. Develop functions to perform the following operations on the data.
 - a) Store details of head and in-charge of the readers club
 - b) Grant and cancel memberships of students
 - c) Display total number of members
 - d) Display the details of the members
 - e) Display the sorted list of details of the members (sort based on their names in alphabetical order)
3. A company has N number of employees and it maintains the following details of each of its employees: ID, department, salary, phone number. Develop a menu driven program using doubly linked list to store the employees' data. Develop functions to perform the following operations on the data.
 - a) Add and delete employees
 - b) Display total number of employees
 - c) Display details of employees with salary more than Rs. 50,000
 - d) Display the phone number of the employee given the ID
4. a) Alexa has two stacks of non-negative integers, stack and stack where index denotes the top of the stack. Alexa challenges Nick to play the following game:
 - i. In each move, Nick can remove one integer from the top of either stack.
 - ii. Nick keeps a running sum of the integers he removes from the two stacks.
 - iii. Nick is disqualified from the game if, at any point, his running sum becomes greater than some integer max_sum given at the beginning of the game.
 - iv. Nick's final score is the total number of integers he has removed from the two stacks.b) Write a program to check whether a string is palindrome or not using stack data structure.

c) Mostly syntax errors in a computer program arise due to unbalanced braces (such as (), {}, []). Write a program using stack to check whether a given expression has balanced braces or not.
5. a) Develop a menu driven program to perform the following operations on a queue of characters (Array and linked list implementations of queue with maximum size MAX)
 - i) Insert an element
 - ii) Delete an element
 - iii) Display the status
 - iv) Demonstrate overflow and underflow situations (in array implementation)b) A restaurant based on its human resources can accept a maximum of N number of food orders. The food orders are served in first come first serve basis. The food orders

once placed cannot be cancelled. Write a program to simulate the food ordering and serving system in the restaurant using circular queue.

6. Write a program to perform the following operations on the binary search tree.
 - a) Construct binary search tree by inserting the values {6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2} in the given order.
 - b) Display the nodes of the tree using inorder, preorder and postorder traversal techniques.
 - c) Display the smallest number stored in the tree.
 - d) Search the tree for a given number.
7. There are train paths between cities. If there is a train between city A and city B then there is a route between the cities. The cost of the route is the distance between city A and city B. Represent the train travel route information as a graph. The node can be represented by the name of the city. Write a program to perform the following operations.
 - a) Store the details of train travel route information using adjacency list or adjacency matrix representation.
 - b) Traverse the graph and display the details of all trains between the cities along with the cost using breadth-first method.
 - c) Traverse the graph and display the details of all trains between the cities along with the cost using depth-first method.
8. Store register numbers of students who attended placement training program in a random order in an array. Write a function to search whether a student has attended placement training program or not using
 - a) Linear search
 - b) Binary search
9. A list of customer names could be sorted into alphabetical order by surname, or a list of people could be put into numerical order by age. Sorting a list of items can take a long time, especially if it is a large list. A computer program can be created to do this, making sorting a list of data much easier. Apply proper sorting mechanisms like quick and shell to sort the data.
10. a) Write a program to sort a given set of integers using merge sort.
 - b) Write a program to read the marks obtained by students in a mathematics examination and store the data using a heap data structure. Find out the maximum and minimum marks obtained by the students.
11. Apply suitable data structure concepts (like hashing) for mapping large chunks of data into small tables.
 - a) Separate Chaining Method
 - b) Open Addressing Method

12. Consider an online movie ticket booking system through which customers can book tickets to watch movies at theatres. The database stores the details of each transaction of ticket booking with the details - ID, customer name, customer phone number, movie name, theatre name, date of show, time of show, number of tickets booked, starting seat number, total amount. Write a menu driven program to perform create the database and given an ID, display a client's phone number. Use a hash table implementation to quickly Search through the database.

RESOURCES

TEXT BOOKS:

1. Robert Lafore, "Data Structures & Algorithms in Java," 2nd Edition, Pearson, 2007.
2. Goodrich, Tamassia, Goldwasser, "Data structures & Algorithms in Java," 6th Edition, Wiley, 2014.

REFERENCE BOOKS:

1. John R. Hubbard, "Programming with Java," McGraw Hill, 2nd Edition, 2009.
2. Debasis Samanta, "Classic Data Structures," Prentice Hall, 2nd Edition, 2009.

SOFTWARE/TOOLS:

1. Software: J2SDK 1.7
 - **Eclipse or Net beans**
2. Java compatible web browser

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/106105175>
2. <https://www.edx.org/course/introduction-to-data-structures>
3. <https://github.com/suhassrivats/Udacity-Data-Structures-and-Algorithms>
4. <https://www.linkedin.com/learning/programming-foundations-algorithms>

WEB RESOURCES:

1. <https://www.javatpoint.com/data-structure-tutorial>
2. <https://medium.com/javarevisited/10-data-structure-and-algorithms-articles-programmer-should-read-this-week-585404a9403b>
3. https://www.tutorialspoint.com/data_structures_algorithms/data_structures_algorithms_pdf_version.html
4. <https://www.w3schools.in/data-structures/intro>
5. https://www.tutorialspoint.com/data_structures_algorithms/data_structures_algorithms_tutorial.pdf
6. <https://lecturenotes.in/subject/81/data-structure-using-c-ds>
7. <https://code.tutsplus.com/series/data-structures-succinctly-part-1--cms-551>
8. <https://www.geeksforgeeks.org/data-structures/>
9. <https://visualgo.net/en>

Course Code	Course Title	L	T	P	S	C
CA25UG0401L	OBJECT ORIENTED PROGRAMMING THROUGH C++	3	-	2	4	5
Pre-Requisite	PROGRAMMING FOR PROBLEM SOLVING					
Anti-Requisite						
Co-Requisite						

COURSE DESCRIPTION: This course provides a detailed discussion on basic characteristics of Object-Oriented Programming through C++. It covers basic Object-Oriented Programming paradigms like Classes and Objects in custom application development. The OOP concepts types of overloading and inheritance are also covered. It provides hands-on experience in implementation of OOP features and other programming concepts like handling pointers, file and exceptions.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate the concepts of Object-oriented programming.
- CO2.** Apply function and operator overloading techniques for code optimization.
- CO3.** Apply inheritance and virtual functions to implement dynamic binding.
- CO4.** Develop robust applications using exception handling mechanism and file I/O.
- CO5.** Develop reliable applications to solve real world problems using Object oriented programming constructs.
- CO6.** Work independently or in team to solve problems with effective communication.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	-	-	-	-	-	-	-	-	-	-	-	3	-	3
CO2	3	2	2	3	2	-	-	-	-	-	-	-	3	-	3
CO3	3	2	2	3	2	-	-	-	-	-	-	-	3	-	3
CO4	3	2	3	3	2	-	1	-	-	-	-	-	3	-	3
CO5	3	2	3	3	2	-	1	-	-	-	-	-	3	-	3
CO6	-	-	-	-	-	-	-	-	3	3	-	-	3	-	3
Course Correlation Mapping	3	2	3	3	2	-	1	-	3	3	-	-	3	-	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: C++ FUNDAMENTALS

(10 Periods)

Need of Object-Oriented Programming - Comparison of procedural programming and Object-Oriented Programming - Characteristics of Object-Oriented Languages - C++ Programming Basics: Basic Program Construction - Data Types, Variables, Constants - Type Conversion, Operators, Library Functions - Loops and Decisions, Structures - Functions : Simple Functions, Passing arguments, Returning values, Reference Arguments. - Recursion, Inline Functions, Default Arguments - Storage Classes - Arrays, Strings.

Module 2: OOP FUNDAMENTALS

(09 Periods)

Classes and Objects: Classes, Friend functions, Friend classes, Inline functions, Parameterized constructors, Static Class Members, The Scope resolution operator, nested and local classes, Passing Objects to functions, returning objects, object assignment. Arrays, Pointers, References: Array of Objects, Pointers to objects, the This pointer.

Module 3: FUNCTION AND OPERATOR OVERLOADING

(09 Periods)

Function Overloading: Copy Constructors, and Default Arguments: Function Overloading, Overloading Constructors, Copy constructors, Default Function Arguments.

Operator Overloading: Creating a member operator function, operator overloading using a friend function, overloading new and delete, overloading some special operators, and comma operator.

Module 4: INHERITANCE AND VIRTUAL FUNCTIONS

(09 Periods)

Inheritance: Base Class Access Control, Inheritance and protected members, inheriting multiple base classes, constructors, destructors, and inheritance, granting access, virtual base classes.

Virtual Functions: Pointers to objects, Pointers to derived classes, Virtual Functions, Pure Virtual Functions.

Module 5: FILE I/O AND EXCEPTIONS

(08 Periods)

Files: File Pointers - Error handling in File I/O - File I/O with member Functions - Overloading the extraction and Insertion Operators - Multi File Programs

Exceptions: Need of Exceptions, keywords, Simple and Multiple Exceptions - Re-throwing Exception and Exception Specifications, Custom Exception

Total Periods: 45

EXPERIENTIAL LEARNING

1.
 - a. Write a C++ program to identify appropriate data types and variables to find the size of various datatypes. Display the variables along with their size.
 - b. Write a C++ program to take name, address as character array, age as int, salary as float and contains inline functions to set the values and display it.
2.
 - a. Write a C++ program to display names, roll no and grades of 3 students who have appeared in examination. Declare the class of name, roll no and grade. Create an array of class objects, read and display the contents of array.
 - b. Create a class TIME with members hours, minutes, seconds. Take input, add two time objects passing objects to function and display the resultant time in hours, minutes & seconds.
3. Given that an EMPLOYEE class contains following members: data members: Employee number, Employee name, Basic, DA, IT, Net Salary and print data members. Write a C++ program to read the data of N employee and compute Net salary of each employee (DA=52% of Basic and Income Tax (IT) =30% of the gross salary).

POLYMORPHISM

4.
 - a. Create a base class basic_info with data members name, roll no, sex and two member functions getdata and display. Derive a class physical_fit from basic_info which has data members height and weight and member functions getdata and display. Display all the information using object of derived class.
 - b. Consider two complex number in the form $a+bi$ and $c+dj$, Write a program overload binary + operator to perform addition of the complex numbers.

INHERITANCE

5.
 - a. Write a C++ Program to calculate the area and perimeter of rectangles using concept of Hierarchical inheritance. Area class is derived from base class Rectangle. Perimeter class is also derived from base class Rectangle.
 - b. Write a C++ program to create Employee and Student inheriting from Person using Hierarchical Inheritance. The Person class should contain the common attributes of Employee and Student class.
6. Design three classes STUDENT, EXAM and RESULT. The STUDENT class has data members such as rollno, name. Create a class EXAM by inheriting the STUDENT class. The EXAM class adds data members representing the marks scored in six subjects. Derive the RESULT from the EXAM class and has its own data members such as total marks. Write a C++ program to model this relationship.
7. Create a base class called SHAPE. Use this class to store two double type values. Derive two specific classes called TRIANGLE and RECTANGLE from the base class. Add to the base class, a member function getdata() to initialize base class data members and another member function display to compute and display the area of figures. Make

display a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes design a program that will accept driven of a TRIANGLE or RECTANGLE interactively and display the area.

FILE I/O

8.
 - a. Write a C++ program named store_temps.cpp that creates a file named raw_temps.txt with temperature data. Fill the file with at least 50 temperature readings.
 - b. Assuming that a text file named FIRST.TXT contains some text written into it, write a function named copyupper() in C++, that reads the file FIRST.TXT and creates a new file named SECOND.TXT contains all words from the file FIRST.TXT in uppercase.
9. Write a C++ program that accepts two file names and produces a new file that is the contents of the first file followed by the contents of the second; that is, the program concatenates the two files.

EXCEPTION HANDLING

10. Add an exception handler (try/catch/throw) to the class that throws an error message (e.g. "value out of range") in the getValue function that is caught and handled in the main program. Implement two version of the program. In version1, throw the error in getValue, catch the error in the main, display an error message in the main, and then allow the program to terminate. In the second version, perform the same basic actions (throw and catch) but keep re-invoking the getValue function from the main program until the user enters a valid value.

PROJECT BASED LEARNING:

11. Faculty shall provide Projects relevant to the contents of the course.

Sample Projects:

Bookshop inventory system

The project is to build the bookshop inventory system in C++ that helps to keep track of all the book records in a shop. Below are the features to be implemented:

- Add new book and the details of the book are:
 - Book ID
 - Name of book
 - Name of Author
 - Number of books
- Delete a book
- Update an existing book detail
- Display summary of all the books
- Search a book

12. **Online Examination Management System**

The project is to build the online examination management system in C++ that helps to conduct online proctored examinations. Below are the features to be implemented:

- Add New Students
- Student Login
 - Registration for examination

- Attempting the questions
- Faculty Login
 - Uploading the questions
- Proctor Login
 - Online monitoring of students
- Admin View
 - Consolidation of marks
 - Release of results

RESOURCES

TEXT BOOKS:

1. Herbert Schildt, "C++ - The Complete Reference", 4th edition, Tata McGraw Hill, 2018.
2. E.Balaguruswamy "Object Oriented Programming with C++", 6th edition, Tata McGraw Hill Education, 2015.

REFERENCE BOOKS:

1. Cohoon and Davidson, "C++ Program Design: An introduction to Programming and object – Oriented Design", 3rd Edition, Tata McGraw Hill, 2003.
2. Robert Lafore, "Object-Oriented Programming in C++", 4th edition, Pearson Education, 2008.
3. Walter Savitch, "Problem Solving with C++", 9th edition, Pearson Education, 2015.

SOFTWARE/TOOLS:

1. Software: Dev C++

VIDEO LECTURES:

1. <https://www.coursera.org/learn/c-plus-plus-a#syllabus>
2. <https://www.udemy.com/free-learn-c-tutorial-beginners/>
3. https://onlinecourses.nptel.ac.in/noc21_cs02/preview

WEB RESOURCES:

1. <http://www.cplusplus.com/files/tutorial.pdf>
2. <http://mazonka.com/shared/Strastrup4th.pdf>

Course Code	Course Title	L	T	P	S	C
CA25UG0204L	GENERAL ENGLISH	2	-	2	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course deals with selected literary works of eminent writers, conversation practice, listening skills, reading comprehensions, writing techniques, vocabulary building and functional grammar.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Analyze the texts using effective reading techniques.
- CO2.** Apply grammatically correct English in writing
- CO3.** Develop ability to understand various accents by developing listening skills.
- CO4.** Apply general and technical vocabulary in effective communication.
- CO5.** Apply different communication styles in various situations.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	-	-	1	-	-	2	-	3	-	2	-	-	-	-
CO2	-	-	-	-	-	-	2	-	3	-	2	1	-	-	-
CO3	-	-	-	-	-	-	2	-	3	-	1	-	-	-	-
CO4	-	-	-	-	-	-	2	-	3	-	2	1	1	1	-
CO5	-	-	-	-	-	1	2	2	3	-	3	2	-	-	-
Course Correlation Mapping	-	-	-	1	-	1	2	2	3	-	2	2	1	1	-

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: A SNAKE IN THE GRASS

(10 Periods)

Reading 'A Snake in the Grass' short story by R.K. Narayan (from the prescribed text book)

Speaking Common Everyday Conversations

Listening Listening for specific information (from the selected audios and videos)

Writing Paragraph Writing: Sentence Structures- use of phrases and clauses - proper punctuation- introducing, logical order, coherence, unity and summarizing.

Grammar Parts of Speech: Nouns, Pronouns, Verbs, Adjectives and Adverbs, Prepositions, Conjunctions and Interjections

Module 2: ON SAYING PLEASE

(09 Periods)

Reading On saying Please 'short essay by A. G. Gardiner (from the prescribed text book)

Listening Listening for understanding concepts (from the selected audios and videos)

Speaking Tongue twisters practice

Writing Letter Writing: Parts of a Letter - Formats of Letters- Types of Letters(enquiry, Complaints, seeking permission, seeking internship etc.)

Grammar Basic sentence structures, word order in sentences, verb agreement, Pronoun Agreement

Module 3: BECAUSE I COULDN'T STOP FOR DEATH

(08 Periods)

Reading Because I couldn't Stop for Death by Emily Dickinson

Listening Listening for understanding concepts (from the selected audios and videos)

Speaking Communication at Work Place

Writing Descriptions: Writing introduction- defining – classifying -describing technical/specific features of an Automobile / gadget/ product or the process- installation manuals

Grammar Usage of Articles and omission of Article, Prepositions and phrasal prepositions

Module 4: AFTER THE SUNSET

(09 Periods)

Reading 'After the Sunset' short story by Bhoopal (from the prescribed text book)

Listening Listening to short audio texts and answering a series of questions

Speaking Reviewing a book or movie/ summarizing or reporting any incident

Writing

Grammar Tenses- Active Voice & Passive Voice; Conditional Sentences

Module 5: MAN’S PERIL

(06 Periods)

Reading ‘Man’s Peril’ a speech made by Bertrand Russell (from the prescribed text book)

Speaking Prepared or extempore speeches

Writing Essay Writing: Writing structured essays on specific topics- Introducing, analyzing and concluding an issue-creating coherence-Usage of proper punctuation.

Vocabulary Antonyms, Synonyms, Idioms and Phrases, One-word substitutes

Grammar Common Errors: Identifying and correcting common errors (articles, prepositions, tenses, subject verb agreement, pronoun agreement etc.)

Total Periods: 45

EXPERIENTIAL LEARNING

1. In rainy seasons a lot of snakes are found crawling around. Prepare a write-up on the reactions of people when they found snakes.
2. India is now for entrepreneurs and the government announced a lot of startup programmes for that. Prepare a presentation on recent entrepreneurs.
3. Small courtesies play a major role in creating an impression on other people. List out a few examples.
4. Prepare a PowerPoint presentation on the present scenario in higher education and jobs in India.
5. Being a shopkeeper and persuading a customer to buy a product which is introduced newly in the market. Prepare a conversation.
6. The English language has a rich vocabulary. List out the homophones and homonyms and write down the pronunciation and meaning of those words.
7. Describe a situation in your college where teamwork is needed and explain the strategies to manage the team effectively.
8. India is a country of unity in diversity. List out the existence of different racial and religious people and bring out reasons for the harmonious relationship among the people.
9. Forget and forgive are the most important quality of any human being. Prepare a write-up on any two experiences which come across in your life where you forgive or forget to maintain good relationships with friends or relatives.
10. Make a case study on the problems of second language learners of English and suggest solutions to overcome them.
11. How do you feel that the role of science and technology in nation-building?

RESOURCES

TEXT BOOKS:

1. G. Damodar "English Language for Undergraduate Students", Cambridge University-2019.

REFERENCE BOOKS:

1. Word Power Made Easy by Norman Lewis, Goyal Publishers, 2020.
2. A Communicative Grammar of English, Geoffrey Leech and Prof Jan Svartvik, Pearson Publication, 2013
3. Grammar in Use Intermediate with Answers, Raymond Murphy, Cambridge University Press, 2019.
4. How to Speak Effectively: A Guide to Engaging Conversations, Presentations, Patrik King, Penguin, 2024

VIDEO LECTURES:

1. <https://www.youtube.com/watch?v=WnOOKO0CdaM>
2. <https://www.youtube.com/watch?v=H6Nlz8qmcFc>
3. <https://www.youtube.com/watch?v=-ITliZO85YM>
4. <https://www.youtube.com/watch?v=048YjXwgHWE>
5. <https://www.youtube.com/watch?v=XLLQm7Grmcc>

WEB RESOURCES:

1. <https://englicist.com/topics/be-the-best>
2. <https://inspiration624.wordpress.com/wp-content/uploads/2018/01/on-saying-please.pdf>
3. <https://englishlanguage-lit.blogspot.com/2021/05/after-sunset-short-story-by-bhoopal.html>
4. <https://www.taylorfrancis.com/chapters/mono/10.4324/9781003090359-31/man-peril-bertrand-russell?context=ubx&refId=1d767e2d-ceb1-4537-9de5-6417eab47d1e>
5. <https://learnenglish.britishcouncil.org/grammar/english-grammar-reference>
6. <https://www.usingenglish.com/reference/grammar/>
7. https://www.englishclub.com/esl-activities/#google_vignette
8. <https://eslvault.com/word-association-games/>
9. <https://testbook.com/english-grammar/tongue-twisters>

Course Code	Course Title	L	T	P	S	C
CA25UG0109L	HUMAN RESOURCE MANAGEMENT	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION:

Concepts of HRM; Environmental Scanning; Human Resource Planning; Job analysis; Job design; Job evaluation; Recruitment; Selection; Placement; Orientation; Training and Development; Performance appraisal; Merit rating; Compensation; Industrial relations; Trade unions; Industrial disputes; Ethical issues; Employee safety.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate the knowledge on the principles, processes and practices of human resource management.
- CO2.** Analyze the key issues related to administering the human elements such as motivation, recruitment, training and development, compensation, appraisal, and career development.
- CO3.** Provide solutions to plan and manage human resource functions effectively within organization.
- CO4.** Apply HRM concepts and techniques in strategic planning to improve organizational effectiveness.
- CO5.** Evaluate HRM related social, cultural and safe responsibilities and issues in a global context.

CO-PO Mapping Table

Course Outcome s	Program Outcomes											Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	-	1	-	-	-	-	-	-	2	-
CO2	3	3	1	1	-	1	-	-	-	-	-	1	-	1
CO3	3	2	3	1	-	-	-	-	-	-	-	-	1	-
CO4	2	1	1	1	3	1	-	-	-	-	-	-	-	-
CO5	3	1	1	1	1	1	2	3	-	-	-	1	-	2
Course Correlation Mapping	3	2	1	1	2	1	2	3	-	-	-	1	-	2

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO HRM & HRP (09 Periods)

Introduction to Human Resource Management (HRM): Objectives, Scope and significance of HRM, Functions of HRM, Prospects in HRM, Environmental scanning.

Human Resource Planning (HRP): Introduction, Nature and importance of HRP, Factors affecting HRP, The planning process, Human resource planning and the Government, Requisites for successful HRP, Barriers to HRP.

Module 2: RECRUITMENT AND PLACEMENT (09 Periods)

Job Analysis – Nature and process of job analysis, Methods of collecting job data, Potential problems with job analysis, Requisites for job analysis; Job Design - Factors, Job design approaches, Contemporary issues; Job evaluation - Process, Methods; Recruitment - Nature, Purposes and importance, Factors governing recruitment, Recruitment process, Evaluation and control; Selection – Nature, Process, Barriers to effective selection, Evaluation of selection process, Placement; Separation.

Module 3: HUMAN RESOURCE DEVELOPMENT AND COMPENSATION (09 Periods)

Orientation - Orientation programme, Requisites of an effective programme, Evaluation of orientation programme, Problems of orientation; Training and development – Nature, Inputs, Training process, Methods, Impediments to effective training, Management development, Career development, Talent management; Performance Appraisal - Nature, Appraisal process, Challenges of performance appraisal; Merit rating; Compensation - Philosophy, Components, Theories, Factors influencing employee compensation, Challenges, Wage and salary administration.

Module 4: INDUSTRIAL RELATIONS AND TRADE UNIONS (09 Periods)

Industrial Relations (IR): Nature of IR, Importance of Peaceful IR; Approaches to IR - Unitary Approach, Pluralistic approach, Marxist approach; Parties to IR; IR strategy; Industrial Disputes - Nature, Causes, and Settlement.

Trade unions: Nature of trade unions, Strategic choices before unions, Union tactics, Trade union movement in India, Trends in trade union movement, Managing unions; Indian Factories Act; Employee's compensation Act; Industrial disputes Act.

Module 5: ETHICAL ISSUES AND SAFETY ADMINISTRATION (09 Periods)

Managing Ethical Issues in HRM: Nature of ethics, Sources of business ethics, Myths about ethics, Ethical dilemmas, HR ethical issues, Managing ethics, Improving ethical decision making.

Employee Safety: Safety, Need for safety, Types of accidents, Safety programme, ISO safety standards.

Total Periods: 45

EXPERIENTIAL LEARNING

1. Prepare a human resource plan for a hypothetical organization based on projected manpower requirements and business objectives.
2. Conduct a job analysis for a selected position and prepare a Job Description (JD) and Job Specification (JS).
3. Visit an organization or industry and Evaluate HRM related social, cultural, ethical and environmental responsibilities and issues in a global context.

4. Design and publish a recruitment advertisement for a vacant position and develop a suitable recruitment strategy.

(Note: It's an indicative one. Course instructor may change the activities and the same shall be reflected in course handout)

RESOURCES

TEXT BOOKS:

1. K. Aswathappa & Sadhna Dash, Human Resource Management: Text and Cases, McGraw Hill Education, 9th Edition, 2021.
2. Gary Dessler & Biju Varkkey, Human Resource Management, Pearson India, 17th Edition, 2024.

REFERENCE BOOKS:

1. Raymond A. Noe, John R. Hollenbeck, Barry Gerhart & Patrick M. Wright, Human Resource Management: Gaining a Competitive Advantage, McGraw-Hill, 13th Edition, 2026.
2. George W. Bohlander, Scott A. Snell & Shad Morris, Principles of Human Resource Management, Cengage Learning, 19th Edition, 2022.
3. Edwin B. Flippo, Personnel Management, McGraw-Hill, 6th Edition, 2007 (latest edition available; no newer editions have been published after the 6th edition).

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/122105020>
2. https://onlinecourses.nptel.ac.in/noc20_mg15/preview

WEB RESOURCES:

1. <https://www.scribd.com/document/892911640/HRM-Unit-3-2023-24>
2. <https://www.scribd.com/document/629313274/Complete-Notes-HRM>

Course Code	Course Title	L	T	P	S	C
CA25UG0402L	PROBLEM SOLVING THROUGH JAVA	3	-	2	4	5
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course introduces Java programming and object-oriented concepts, covering platform independence, data types, operators, and control structures, along with arrays, strings, methods, recursion, and debugging techniques to enhance problem-solving skills. It emphasizes core object-oriented principles such as classes, objects, inheritance, polymorphism, abstraction, and interfaces. Advanced topics, including exception handling and multithreading are also covered. The course further explores the Java Collection Framework to enable the development of efficient, modular, and reusable code.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate knowledge of Java fundamentals, including OOP Concepts, JVM, data types, I/O methods and use classes, objects, constructors, and methods to create modular and reusable Java programs.
- CO2.** Apply control structures, arrays, strings methods and recursion to develop modular and efficient Java programs and effectively debug and manage code.
- CO3. Develop** object-oriented applications using polymorphism, inheritance, interfaces, abstract classes, and packages to promote code extensibility and reusability.
- CO4.** Utilize string handling methods, exception handling, multithreading, and collection framework classes to solve real-world problems.
- CO5.** Demonstrate effective individual teamwork skills and effectively communicate experimental results and project outcomes.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes											Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	3	-	-	-	-	-	-	-	-	-
CO2	3	2	-	-	3	-	-	-	-	-	-	-	-	-
CO3	3	3	3	3	3	-	-	-	-	-	-	-	1	-
CO4	3	2	2	2	3	-	-	-	-	-	-	-	1	2
CO5	-	-	-	-	-	-	-	-	3	-	3	-	-	-
Course Correlation Mapping	3	3	3	3	3	-	-	-	3	-	3	-	2	1

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO JAVA AND OOP FUNDAMENTALS (10 Periods)

Features of Java, Platform Independence, **Java Environment**, Java Program Structure, Source and Class Files, Data Types, Variables, Type Conversion, Type Casting, Basic Input and Output Methods.

Operators: Arithmetic, Relational, Logical, Assignment.

String Operations: String Length, String Literals, Character Extraction: `charAt()`; String Comparison: `equals()`.

Classes and Objects: Introduction to classes, objects, Constructors, Garbage Collection, `this` keyword, Access Control, Features of Object Oriented Programming.

Module 2: CONTROL STRUCTURES AND ARRAYS (9 Periods)

Control Structures: Decision-Making Statements: `if`, `if-else`, `switch`; Looping Constructs: `for`, `while`, `do-while`, `break`, `continue`.

Arrays: One-dimensional Arrays, Multi-Dimensional Arrays, Storage Layouts, Basic Operations.

Module 3: METHODS AND POLYMORPHISM (8 Periods)

Methods: Method Structure, Parameter Passing, Return Mechanisms, Recursion.

Polymorphism: Method Overloading, Constructor Overloading.

Module 4: INHERITANCE, PACKAGES AND INTERFACES (8 Periods)

Inheritance: Inheritance Basics, Using `Super`, Multi-Level Hierarchy, Method Overriding, Abstract Classes, `Final` Keyword.

Packages: Basics, Creating and Accessing a package, `CLASSPATH`, Importing packages.

Interfaces: Definition, Implementing Interfaces, Nested Interfaces, Applying Interfaces, Variables in Interfaces, Extending Interfaces.

Module 5: EXCEPTION HANDLING, MULTITHREADING, STRING HANDLING AND COLLECTION FRAMEWORK (10 Periods)

Exception Handling: Exception, Types of Exception, Keywords: `try`, `catch`, `throw`, `throws` and `finally`, Built-in Exceptions, User Defined Exceptions.

Multithreading: Process, Thread, Thread Model, Creating a thread, Priorities, Thread Synchronization, Inter-thread Communication.

String Handling: String constructors, String Conversion and `toString()`, Search Strings, String Buffer.

Collection Framework: Collection Interface: `List`, `Set`; Collection Classes: `ArrayList`, `LinkedList`, `HashSet`, `HashMap`.

Total Periods: 45

EXPERIENTIAL LEARNING

(This is an indicative. The Course Instructor may modify the activities, and the same shall Be reflected in the Course Handout.)

(All source code should be maintained and version-controlled on GitHub)

1) INTRODUCTION TO JAVA AND OOP FUNDAMENTALS

Leetcode: #2235. Add Two Integers

Given two integers `num1` and `num2`, return the sum of the two integers.

Example 1: Input: num1 = 12, num2 = 5 Output: 17

Example 2: Input: num1 = -10, num2 = 4 Output: -6

Constraints: $-100 \leq \text{num1}$, $\text{num2} \leq 100$

✓ This problem reinforces the fundamentals of Java arithmetic operators and integer data types. It helps beginners understand how basic expressions are evaluated in Java and builds confidence in writing and testing simple methods.

2) **INTRODUCTION TO JAVA AND OOP FUNDAMENTALS**

A retail store wants to manage its product inventory using object-oriented principles. Design a Java program using classes, objects, and constructors to model this scenario.

Program Specifications:

Create a class named **Product** with attributes: String productName, int productId, double price, int quantity.

Include a parameterized constructor to initialise all attributes.

Add a method displayProduct() that prints all product details in a formatted manner.

Add a method updateStock(int units) that increases quantity by the given units and prints the updated stock.

Create an **Inventory** class with a main method that creates at least two Product objects, displays their details, and demonstrates updateStock().

Sample Output:

Product ID: 101

Product Name: Laptop

Price: Rs.55000.0

Quantity in Stock: 10

Updating stock for Laptop by 5 units.

New Stock: 15

✓ This problem reinforces the creation of classes and objects, the use of parameterized constructors, and the concept of encapsulation. It gives students hands-on practice in modelling real-world entities using Java OOP principles.

3) **CONTROL STRUCTURES AND ARRAYS**

Leetcode: #1480. Running Sum of 1d Array

Given an array nums, return the running sum of nums. $\text{runningSum}[i] = \text{sum}(\text{nums}[0] + \text{nums}[1] + \dots + \text{nums}[i])$.

Example 1: Input: nums = [1,2,3,4] Output: [1,3,6,10]

Example 2: Input: nums = [1,1,1,1,1] Output: [1,2,3,4,5]

Example 3: Input: nums = [3,1,2,10,1] Output: [3,4,6,16,17]

Constraints: $1 \leq \text{nums.length} \leq 1000$ $-10^6 \leq \text{nums}[i] \leq 10^6$

✓ This problem strengthens one-dimensional array traversal and in-place accumulation techniques. It helps learners practice iterating through arrays using loops and applying cumulative logic, which is foundational to many data-processing algorithms.

4) **CONTROL STRUCTURES, ARRAYS AND METHODS**

Leetcode: #2798. Number of Employees Who Met the Target

There are n employees, numbered 0 to n-1. Employee i worked hours[i] hours. The company requires at least target hours. Return the count of employees for whom $\text{hours}[i] \geq \text{target}$.

Example 1: Input: hours = [0,1,2,3,4], target = 2 Output: 3

Example 2: Input: hours = [5,1,4,2,2], target = 6 Output: 0

Constraints: $1 \leq n = \text{hours.length} \leq 50$ $0 \leq \text{hours}[i]$, $\text{target} \leq 10^5$

✓ This problem develops array traversal skills combined with conditional logic. It gives students practice in counting elements that satisfy a criterion, which is a fundamental pattern used in searching and filtering operations.

5) **CONTROL STRUCTURES, ARRAYS AND METHODS**

Leetcode: #2011. Final Value of Variable After Performing Operations

A language has four operations on variable X: ++X and X++ increment X by 1; --X and X-- decrement X by 1. Initially X = 0. Given an array of strings operations, return the final value of X.

Example 1: Input: operations = ["--X","X++","X++"] Output: 1

Example 2: Input: operations = ["++X","++X","X++"] Output: 3

Constraints: $1 \leq \text{operations.length} \leq 100$ Each operation is ++X, X++, --X, or X--.

✓ This problem reinforces looping constructs and string comparison in Java. It trains students to iterate over an array, parse string tokens, and apply conditional updates, which are core skills for building control-flow logic in real applications.

6) **CONTROL STRUCTURES, ARRAYS AND METHODS**

Leetcode: #2160. Minimum Sum of Four Digit Number After Splitting Digits

You are given a four-digit positive integer num. Split num into two new integers new1 and new2 using its digits. Leading zeros are allowed; all digits must be used. Return the minimum possible sum of new1 and new2.

Example 1: Input: num = 2932 Output: 52 (new1=29, new2=23)

Example 2: Input: num = 4009 Output: 13 (new1=04, new2=09)

Constraints: $1000 \leq \text{num} \leq 9999$

✓ This problem deepens understanding of integer digit extraction using the modulo and division operators. It challenges students to apply sorting on a small array and construct numbers from individual digits, reinforcing data-type manipulation skills covered in Module 2.

7) **METHODS AND STRING**

Leetcode: #2283. Check if Number Has Equal Digit Count and Digit Value

You are given a 0-indexed string num of length n. Return true if for every index i in range 0 to n-1, digit i occurs exactly num[i] times in num; otherwise return false.

Example 1: Input: num = "1210" Output: true

Example 2: Input: num = "030" Output: false

Constraints: $n == \text{num.length}$ $1 \leq n \leq 10$ num consists of digits.

✓ This problem combines loop traversal with character-to-integer conversion and frequency counting. It helps students practice iterating over character arrays and strengthens their ability to map problem conditions to array indices, a key skill in algorithmic thinking.

8) **METHODS AND STRING**

Leetcode: #412. Fizz Buzz

Given an integer n, return a string array (1-indexed) where: answer[i] == "FizzBuzz" if i is divisible by 3 and 5; answer[i] == "Fizz" if divisible by 3; answer[i] == "Buzz" if divisible by 5; otherwise answer[i] is i as a string.

Example 1: Input: n = 3 Output: ["1","2","Fizz"]

Example 2: Input: n = 5 Output: ["1","2","Fizz","4","Buzz"]

Constraints: $1 \leq n \leq 10^4$

✓ FizzBuzz solidifies understanding of loop constructs and nested conditional statements. It encourages students to think about operator precedence and condition ordering, which are critical debugging skills for any Java programmer.

9) **POLYMORPHISM AND INHERITANCE**

A geometry application needs to calculate areas of different shapes. Design a Java program demonstrating method overloading to compute the area of a circle, rectangle, and triangle using a single class.

Program Specifications:

Create a class **AreaCalculator** with the following overloaded methods:

double calculateArea(double radius): area of a circle using $\text{Math.PI} * \text{radius} * \text{radius}$.

double calculateArea(double length, double breadth): area of a rectangle.

double calculateArea(double base, double height, String shape): when shape is "triangle", returns $0.5 * \text{base} * \text{height}$.

Create a **Main** class that reads inputs from the user for each shape and displays the calculated area to two decimal places.

Sample Input / Output:

Enter radius: 7.0

Area of Circle: 153.94

Enter length and breadth: 4.0 5.0

Area of Rectangle: 20.00

Enter base and height: 6.0 8.0

Area of Triangle: 24.00

✓ This problem demonstrates compile-time polymorphism through method overloading. Students learn how the Java compiler resolves method calls based on the number and type of arguments, and how overloading makes APIs cleaner and more intuitive.

10) **POLYMORPHISM AND INHERITANCE**

A transportation company wants to model different types of vehicles. Design a Java program using an abstract class and an interface to demonstrate abstraction and runtime polymorphism.

Program Specifications:

Define an interface **Serviceable** with a method void scheduleService().

Create an abstract class **Vehicle** implementing Serviceable with: String vehicleName, int vehicleId; abstract method void displayInfo(); concrete method void startEngine() printing "[vehicleName] engine started".

Create two concrete classes **Car** and **Truck** that extend Vehicle and implement displayInfo() and scheduleService().

In a **Main** class, create Car and Truck objects using Vehicle references and demonstrate polymorphic method calls.

Sample Output:

Car [ID: 201, Name: Sedan] - Fuel Type: Petrol Sedan engine started. Service scheduled for Sedan at nearest service center.

✓ This problem illustrates runtime polymorphism using abstract classes and interfaces. Students gain practical experience in designing extensible class hierarchies and understand the role of abstraction in building flexible, maintainable software systems.

11) **EXCEPTION HANDLING**

A bank requires a robust account management module that validates deposit and withdrawal transactions. Design a Java program using user-defined exception classes to enforce business rules.

Program Specifications:

Create a custom exception class **InsufficientFundsException** extending Exception. Accept a double shortfall in the constructor and return a meaningful message via getMessage().

Create a custom exception class **InvalidAmountException** extending Exception, thrown when a non-positive amount is provided.

Create a class **BankAccount** with: String accountHolder, double balance; deposit(double amount) throwing InvalidAmountException if amount is 0 or negative; withdraw(double amount) throwing InvalidAmountException for non-positive amounts and InsufficientFundsException for insufficient balance; displayBalance().

In a **Main** class, create a BankAccount, perform transactions based on user input, and handle all exceptions using try-catch-finally.

Sample Output:

Deposited Rs.5000.0 successfully.

Current Balance: Rs.5000.0

Error: Insufficient funds.

Shortfall: Rs.2000.0

✓ This problem gives students practical experience with user-defined exceptions, making them understand how to enforce domain-specific rules through the exception hierarchy. It reinforces the use of try, catch, and finally blocks in building reliable Java applications.

12) **MULTITHREADING**

Leetcode: [#1114. Print in Order](#)

Consider a class with three methods: first(), second(), and third(). The same instance will be passed to three different threads. Thread A calls first(), Thread B calls second(), Thread C calls third(). Design a mechanism to ensure second() executes after first(), and third() executes after second().

Example 1: Input: nums = [1,2,3] Output: firstsecondthird

Example 2: Input: nums = [1,3,2] Output: firstsecondthird

Constraints: nums is a permutation of [1, 2, 3].

✓ This problem introduces students to thread synchronisation and execution order in Java. It bridges classes/objects with concurrent programming and encourages learners to explore Java synchronisation primitives such as Object.wait() and Object.notify().

1) **SAMPLE CONSOLE-BASED MINI PROJECT (PROJECT BASED LEARNING)**

Library Book Management System

Description:

Develop a console-based Library Book Management System in Java that allows a librarian to manage book records. The system should support adding new books, displaying all available books, searching for a book by title or author, issuing a book to a member, and returning an issued book. The system must use object-oriented principles throughout.

Functional Requirements:

1. Add Book: Accept book details (Book ID, Title, Author, Genre, Total Copies) from the console and store them.
2. Display All Books: Show all books with their current available copies.
3. Search Book: Allow searching by title (case-insensitive partial match) or by author name.
4. Issue Book: Reduce available copies by 1 when a member borrows a book; display an error if no copies are available.
5. Return Book: Increase available copies by 1 when a member returns a book.
6. Exit: Terminate the application.

OOP Design Guidelines:

Create a class **Book** with private attributes and public getters/setters to encapsulate book data.

Create a class **Library** that maintains an ArrayList of Book objects and implements all operations as methods.

Create a **Main** class with a menu-driven loop that calls appropriate Library methods based on the user choice.

Use a Scanner for all console input. Apply exception handling for invalid menu choices and input mismatches.

Sample Menu:

1. Add Book
2. Display All Books
3. Search Book
4. Issue Book

Enter your choice:

✓ This project integrates classes, objects, encapsulation, ArrayList-based collections, and menu-driven console I/O into a cohesive application. Students learn to decompose a real-world problem into well-structured classes and gain experience in building an end-to-end Java program with in-session data persistence.

2) **Student Report Card Generator**

Description:

Design and implement a console-based Student Report Card Generator in Java. The system allows a teacher to enter details and marks for multiple students, automatically calculates each student's total, average, percentage, and grade, and prints a formatted report card. The application must be built using object-oriented principles.

Functional Requirements:

1. Input: Accept the student's name, roll number, and marks in five subjects (out of 100) via console.
2. Computation: Calculate total marks, percentage (total / 500 * 100), and assign a grade: A+ (90% and above), A (80% and above), B (70% and above), C (60% and above), D (50% and above), F (below 50%).
3. Report Card: Display a formatted report card showing name, roll number, subject marks, total, percentage, grade, and pass/fail status (pass if percentage is 40% or

above).

4. Class Summary: After processing all students, display the class topper (highest percentage) and the class average percentage.

5. The system should support multiple students in a single session using a loop.

OOP Design Guidelines:

Create a class **Student** with private fields for name, rollNumber, and an int[] marks array of size 5; include methods computeTotal(), computePercentage(), and getGrade(). Create a class **ReportCard** with a method generateReport(Student s) that prints the formatted report card.

Create a **Main** class that manages a list of Student objects, collects input, triggers report generation, and computes class-level statistics.

Apply encapsulation (private fields with public methods) and separation of concerns between data handling and report generation.

Sample Output (partial):

===== REPORT CARD =====

Name: Priya Sharma

Roll No: 21CS001

Maths: 88

Science: 79

English: 92

History: 74

CS: 95

Total: 428/500

Percentage: 85.60%

Grade: A

Result: PASS

✓ This project provides a holistic exercise in OOP design: students practice encapsulation using private fields and accessor methods, separation of concerns by splitting data and presentation into different classes, and collection management using ArrayList. The class summary feature introduces basic aggregate computations, preparing students for real-world data-processing tasks.

RESOURCES

TEXT BOOKS:

1. Herbert Schildt, *Java: The Complete Reference*, McGraw-Hill Education, 13th Edition, 2024.
2. Paul Deitel, Harvey Deitel, *Java: How to Program*, Pearson, 11th Edition, 2017.

REFERENCE BOOKS:

1. Cay S. Horstmann, *Core Java Volume I – Fundamentals*, 13th Edition, Pearson, 2025.
2. Kathy Sierra, Bert Bates, *Head First Java*, 2nd Edition, O'Reilly Media, 2005.

SOFTWARE/TOOLS:

1. Java JDK (Java SE 17 or above)
2. IDEs: IntelliJ IDEA / Eclipse / NetBeans

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/106105191>
2. https://www.youtube.com/playlist?list=PLsyeobzWxl7pe_IiTfNyr55kwJPWbgxB5
3. https://www.youtube.com/watch?v=xk4_1vDrzzo

4. https://www.youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q
5. <https://www.youtube.com/watch?v=eIrMbaQSU34>

WEB RESOURCES:

1. <https://www.geeksforgeeks.org/java/java/>
2. <https://www.programiz.com/java-programming>
3. https://www.w3schools.com/java/java_oop.asp
4. <https://www.hackerrank.com/domains/java>
5. <https://leetcode.com/problemset/?language=Java>
6. <https://www.codechef.com/practice/java>

Course Code	Course Title	L	T	P	S	C
CA25UG0110L	SOFTWARE ENGINEERING	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: This course provides a detailed discussion on concepts of Software Engineering, Software Process Models, Conventional and Agile Process Models, Software Requirements Engineering Process, System Analysis, Architectural Design, User Interface Design and Re-engineering, Software Testing, Risk and Quality Management

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Demonstrate concepts of software engineering and analyze process models required to develop a software system.
- CO2.** Analyze software requirements and model requirements for the given scenario.
- CO3.** Apply design concepts and metrics for software development.
- CO4.** Apply testing strategies and techniques for quality software.
- CO5.** Analyze risks in software development life cycle and apply risk strategies to mitigate risks.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	-	-	-	-	-	-	-	3	-	-
CO2	2	3	-	-	-	3	3	2	-	-	2	-	3	-	-
CO3	2	3	3	2		2	2	-	-	-	3	-	3	-	-
CO4	2	-	-	-	2	3	-	-	-	-	-	-	3	-	-
CO5	2	3	-	-	2	3	-	-	-	-	-	-	3	-	-
Course Correlation Mapping	2	3	2	2	1	3	2	2	-	-	2	-	3	-	-

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: SOFTWARE PRODUCT AND SDLC

(11 Periods)

Software Engineering Fundamentals, Definition of Software Products, Phases of Software Development Life Cycle, Software Development Paradigm, Software Life Cycles Models: Build and Fix Model, Waterfall Model, Prototype Model, Iterative Model, Evolutionary Model, Spiral Model.

Module 2: SOFTWARE REQUIREMENT SPECIFICATION (SRS)

(07 Periods)

Need for SRS-Requirement process, Problem Analysis using UML (Unified Modelling Language) and Data dictionary, Characteristics of SRS, Components of an SRS. IEEE standard for SRS.

Module 3: SOFTWARE DESIGN PRINCIPLES

(09 Periods)

Software Design, Design Process, Design Principles: Abstraction, Refinement, Modularity, Information Hiding, Modular Design: Effective Modular Design and Functional Independence, Cohesion, Coupling, Top down and Bottomup Strategies, Coding: Coding Standard and Guidelines, Testing: Black Box Testing and White Box Testing.

Module 4: SOFTWARE RELIABILITY AND REUSABILITY

(08 Periods)

Software reliability metrics , Software reliability Specification , Statistical testing ,Reliability Growth modeling, Fault avoidance & tolerance, Exception handling & defensive programming , Software development with reuse, Software' development for reuse , Generator based reuse, Application System Portability.

CASE TOOLS AND PROJECT MANAGEMENT

Module 5: FUNDAMENTALS:

(10 Periods)

Relevance of CASE Tool, Building block for CASE Tools, Integrated Case Tool Environment, Generation of CASE Tool, High End and Low End CASE Tools.

Definition of Project, Project Specification and Parameters, COCOMO model, Principles, of Project Management, Project Management Life Cycle, Program Management Plan: Concept, Elements, Planning Issues, Benefits of Program Management..

Total Periods: 45

EXPERIENTIAL LEARNING

1. Identify Functional and Non-Functional Requirements for:
 - a. Online Ticket Reservation for Railways
 - b. Recruitment Procedure for Software Industry

2. **Online Ticket Reservation for Railways**

Problem Statement: Computer plays an integral part of the day in today's life. It makes the entire job easier and faster, every job is computerized so as the ticket reservation we can book over the online ticket reservation system. During the booking of the ticket reservation passenger has to select origin, date of journey, destination, class of train etc. The reservation counter keeps track of passenger's information. Thus the system will have all the details about the trains and facilities provided by them. There are various trains with the different level of convenience for the passengers. The whole database will be maintained by database administrator. There are varieties of trains where the passengers can select the train according to the convenience for their destination journey. The journey could be within the state or across the India. Each train has the three types of classes i.e. Sleeper class, First class and the AC compartment. Design the application for the above problem description. Model using Visual modelling tools in different views i.e. component view, Deployment view.

3. **Recruitment Procedure for Software Industry**

Problem Statement: In the software industry the recruitment procedure is the basic thing that goes in the hand with the requirement as specified by the technical management team. HR first gives an advertisement in leading Newspapers, Journals, Weeklies and Websites. The job seekers can apply for it through by Post or by e-mail to the company. The technical skill and the experience of the candidates are reviewed and the short listed candidates are called for the interview. There may be different rounds for interview like the written test, technical interview, and HR interview. After the successful completion of all rounds of interview, the selected candidates' names are displayed. Meanwhile HR gives all the details about the salary, working hours, terms and conditions and the retirement benefit to the candidate. Model using Visual modelling tools in different views i.e. Use case view, logical view

RESOURCES

TEXT BOOKS:

1. Roger S. Pressman, "Software Engineering - A Practitioner's Approach," McGraw-Hill International Edition, 8th Edition, 2015.
2. Ian Sommerville, "Software Engineering," Pearson Education, 9th Edition, 2011.

REFERENCE BOOKS:

1. Grady Booch, James Rumbaugh and Ivar Jacobson, "The Unified Modeling Language User Guide," 2nd Edition, Pearson Education, 200
2. K. K. Aggarwal and Yogesh Singh, "Software Engineering," New Age International Publishers, 3rd Edition, 2007.
3. Shelly Cashman Rosenblatt, "Systems Analysis and Design," Thomson Publications, 6th Edition, 2006.

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/106105087/>
2. <https://nptel.ac.in/courses/106105182/>

WEB RESOURCES:

1. Agile Modeling : <https://www.techtarget.com/searchsoftwarequality/definition/agile-software-development>
2. Functional and Non-Functional Requirements:
<https://www.linkedin.com/learning/software-design-developing-effective-requirements/functional-vs-non-functional>

Course code	Course Title	L	T	P	S	C
CA25UG0205L	COMPUTER NETWORKS	3	-	2	-	4
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: Concepts of Computer Networks, The Physical Layer, The Data Link Layer, The Network Layer, The Transport Layer, The Application Layer.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Understand the concepts of Networking, reference models, transmission media.
- CO2.** Apply error and flow control techniques and implement congestion control mechanisms and transport protocols for reliable data communication.
- CO3.** Analyze design issues of layers, techniques for quality of service, elements of transport and application Protocols ensuring the communication procedures.
- CO4.** Examine the layered and e-mail architectures, networking protocols and e-mail message formats in compliance with communication standards.

CO-PO-PSO Mapping Table:

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	-	-	-	-	-	-	-	-	-	-	-	3
CO2	2	2	3	1	3	1	-	-	-	-	-	-	-	-	3
CO3	2	3	3	-	2	2	-	-	-	-	-	-	-	-	3
CO4	-	1	1	-	-	2	-	-	-	-	-	-	-	-	2
Course Correlation Mapping	2	2	3	1	2	2	-	-	-	-	-	-	-	-	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: NETWORK CONCEPTS AND PHYSICAL LAYER (09 Periods)

Network Concepts: Uses of Computer Networks, Network Hardware, Network Software, Reference Models-OSI, TCP/IP. **The Physical Layer:** Guided Transmission media-Magnetic Media, Twisted Pairs, Coaxial Cable, Fiber Optics. Wireless Transmission-The Electromagnetic Spectrum, Radio Transmission, Infrared Transmission and Light Transmission

Module 2: THE DATA LINK LAYER (09 Periods)

The Data Link Layer: Data Link layer design issues, Error Detection and Correction, Elementary Data Link Protocols - Unrestricted simplex protocol, Simplex stop-and-wait protocol, Simplex protocol for a noisy channel. Sliding Window protocols - One-bit sliding window protocol, Protocol using Go back N, Protocol using Selective Repeat.

Module 3: THE NETWORK LAYER (10 Periods)

Network layer design issues, Routing Algorithms-Optimality principle, Shortest Path Routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing, Broadcast Routing, Multicast Routing, Congestion Control Algorithms, Internetworking, The Network layer in the Internet-The IP Version 4 protocol, IP Addresses.

Module 4: THE TRANSPORT LAYER (10 Periods)

The Transport service, Elements of Transport protocols -Addressing, Connection Establishment, Error Control and Flow Control, Multiplexing and Crash recovery, Remote Procedure Call, Connectionless Transport: UDP, Principles of Reliable Data transfer, Connection-Oriented Transport: TCP, Principles of Congestion Control, TCP Congestion Control.

Module 5: THE APPLICATION LAYER (07 Periods)

DNS - The Domain name space, Resource records and Name servers; Electronic Mail-Architecture and services, the user agent, message formats, message transfer and Final Delivery.

Total Periods: 45

EXPERIENTIAL LEARNING

1. Write a C program to implement the algorithm for parity method for error control.
2. Write a C program to implement the algorithm on hamming method for error correction (both single and block errors).
3. Write a C program to implement the algorithm for check sum computation
4. Explain the functions of different network layers using real-world examples.
5. Design and simulate small computer networks using Cisco Packet Tracer.

6. Configure IP addressing and verify network connectivity.
7. Analyze TCP/IP communication using packet analysis tools.
8. Demonstrate error detection and flow control mechanisms.
9. Understand DNS, email services, and application-layer protocols through practical exercises.

RESOURCES

TEXT BOOK:

1. Andrew S. Tanenbaum and David J. Wetherall, "Computer Networks," Pearson Education, 5th Edition, 2015.

REFERENCE BOOKS:

1. Behrouz A. Forouzan, "Data Communications and Networking," The McGraw-Hill, 4th Edition, 2011.
2. James F. Kurose and Keith W. Ross, "Computer Networking: A Top-Down Approach," Pearson Education, 6th Edition, 2017.

VIDEO LECTURES:

1. <https://nptel.ac.in/courses/106/105/106105081/>
2. https://www.youtube.com/watch?v=6_PINy02_g0

WEB RESOURCES:

1. <https://www.cisco.com/c/en/us/solutions/small-business/resource-center/networking/networking-basics.html>
2. <https://memberfiles.freewebs.com/00/88/103568800/documents/Data.And.Computer.Communications.8e.WilliamStallings.pdf>
3. [https://www01.ibm.com/servers/resourcelink/svc0302a.nsf/pages/zVMV7R1sc246333/\\$file/kijl0_v7r1.pdf](https://www01.ibm.com/servers/resourcelink/svc0302a.nsf/pages/zVMV7R1sc246333/$file/kijl0_v7r1.pdf)
4. <http://ns2simulator.com/ns2-tcp-congestion-control/>

Course Code	Course Title	L	T	P	S	C
CA25UG0206L	CLOUD COMPUTING	3	-	2	-	4

Pre-Requisite -

Anti-Requisite -

Co-Requisite -

COURSE DESCRIPTION: Virtualization, Cloud Computing Fundamentals, Deployment Models; Cloud Computing Architecture; Cloud Computing Mechanisms; Cloud Security; Working with Clouds; and Case Studies.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1.** Understand the importance of virtualization and how this has enabled the development of Cloud Computing.
- CO2.** Analyze the cloud computing models, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.
- CO3.** Implement different types of Virtualization technologies and Service Oriented Architecture systems
- CO4.** Apply modern technologies & tools of Cloud Computing in solving resource sharing problems in industry.
- CO5.** Identify security and privacy issues in cloud computing.

CO-PO-PSO Mapping Table:

Course Outcome	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	-	-	3	-
CO2	2	3	-	-	-	-	-	-	-	-	-	-	-	3	-
CO3	2	3	2	1	-	-	-	-	-	-	-	-	-	3	-
CO4	3	3	3	1	-	-	-	-	-	-	-	-	-	3	-
CO5	2	3	2	-	-	2	-	-	-	-	-	-	-	3	-
Course Correlation Mapping	2	3	2	1	-	2	-	-	-	-	-	-	-	3	-

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: INTRODUCTION TO VIRTUALIZATION**(07 Periods)**

Virtualization: Introduction to Virtualization, objectives of virtualization, benefits of virtualized technology, Adding guest Operating system.

Virtualization Technologies: Ubuntu, VMware, Microsoft Hyper-V.

Module 2: DEFINING CLOUD COMPUTING**(10 Periods)**

Defining Cloud Computing: Defining Cloud Computing, Cloud Types - The NIST model, The Cloud Cube Model, Deployment models, Service models, Benefits of cloud computing, Disadvantages of cloud computing, Assessing the Role of Open Standards.

Understanding Cloud Architecture: Exploring the Cloud Computing Stack, Compos ability, Infrastructure, Platforms, Connecting to the Cloud.

Module 3: UNDERSTANDING SERVICES AND APPLICATIONS**(10 Periods)**

Understanding Services And Applications By Type: Defining Infrastructure as a Service (IaaS), IaaS workloads, Pods, aggregation, and silos, Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS), SaaS characteristics.

Understanding Abstraction and Virtualization: Using Virtualization Technologies, Load Balancing and Virtualization, Understanding Hypervisors, Virtual machine types, VMware vSphere.

Module 4: EXPLORING PLATFORM AS A SERVICE**(10 Periods)**

Using Amazon Web Services: Understanding Amazon Web Services, Amazon Web Service Components and Services, Working with the Elastic Compute Cloud (EC2), Amazon Machine Images, Pricing models, System images and software, Creating an account and instance on EC2.

Managing the Cloud: Administrating the Clouds, Management responsibilities, Lifecycle management, Emerging Cloud Management Standards, DMTF cloud management standards, Cloud Commons and SMI.

Module 5: Understanding Cloud Security**(08 Periods)**

Understanding Cloud Security: Securing the Cloud, The security boundary, Security service boundary, Security mapping, Securing Data, Brokered cloud storage access, Storage location and tenancy, Encryption, Auditing and compliance, Establishing Identity and Presence, Identity protocol standards, Windows Azure identity standards, Presence.

Total Periods: 45**EXPERIENTIAL LEARNING**

1. Create a word document of your class time table and store locally and on the cloud with doc, and pdf format.
2. Create a spread sheet which contains employee salary information and calculate gross and total sal using the formula DA=10% OF BASIC HRA=30% OF BASIC PF=10% OF BASIC IF BASIC≤3000 12% OF BASIC IF BASIC>3000 TAX=10% OF BASIC IF BASIC≤1500 =11% OF BASIC IF BASIC>1500 AND BASIC≤2500 =12% OF BASIC IF BASIC>2500 NET_SALARY=BASIC_SALARY+DA+HRA-PF-TAX.
3. Prepare a ppt on cloud computing – Introduction, models, services, and Architectures. PPTs should contain explanations, images and at least 20 slides
4. Create your resume in a neat format using google cloud Programs on PaaS.

5. Create an EC2 instance and invoke Ubuntu operating system with a given set of configuration on amazon web services under IaaS.
6. Create S3 bucket and store a file in it using AWS.
7. Install **Oracle VirtualBox** and create a virtual machine.
8. Install **Ubuntu Linux** as a guest operating system and perform basic Linux commands.
9. Deploy a simple web application using a Docker container.
10. Configure secure remote access to a Linux virtual machine using **OpenSSH**.

RESOURCES

TEXT BOOKS:

1. Barrie Sosinsky, "Cloud Computing Bible," Wiley India Pvt Ltd, 1st Edition, 2011.
2. Ivanka Menken Ivanka Menken, "Cloud Computing Virtualization Specialist Complete Certification Kit - Study Guide Book", Emereo Publishing, 2nd Edition, 2012.

REFERENCE BOOKS:

1. Anthony T. Velte, Toby J. Velte Robert Elsenpeter, "Cloud Computing: A practical Approach", Tata Mc Graw Hill, ISBN: 9780071626941, 1st Edition, 2010.
2. John W. Rittinghouse, James F. Ransome, "Cloud Computing implementation, Management and Security", CRC Press, ISBN: 9788120341609, Taylor & Francis group, 1st Edition 2010.
3. George Reese, "Cloud Application Architectures", Oreilly publishers, 1st Edition, 2010.
4. David S. Linthicum, "Cloud Computing and SOA Convergence in your Enterprise", Addison- Wesley, 1st Edition, 2010.

VIDEO LECTURES:

1. https://onlinecourses.nptel.ac.in/noc21_cs14/preview
2. <https://www.coursera.org/specializations/cloud-computing>
3. <https://www.udemy.com/course/introduction-to-cloud-computing-on-amazon-aws-for-beginners/>

WEB RESOURCES:

1. https://onlinecourses.nptel.ac.in/noc21_cs14/preview
2. [https:// trailhead.salesforce.com/en/home](https://trailhead.salesforce.com/en/home)
3. <https://mkyong.com/tutorials/google-App- engine-tutorial/>
4. <https://www.awsacademy.com>

Course Code	Course Title	L	T	P	S	C
CA25UG0111L	ARTIFICIAL INTELLIGENCE	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: Introduction to artificial intelligence, Designing intelligent agents, Solving general purpose problems, Search in complex environments, Probabilistic reasoning, Represent knowledge and reason under uncertainty, Robotics, Ethics and safety in AI.

COURSE OUTCOMES: After successful completion of this course, the students will be able to:

- CO1.** Architect intelligent agents using artificial intelligence techniques and principles.
- CO2.** Analyze and interpret the problem, identify suitable solutions using heuristic functions, optimization algorithms and search algorithms.
- CO3.** Select and apply appropriate knowledge representation to build Bayesian network models to reason under uncertainty.
- CO4.** Investigate robot hardware and frameworks for intelligent robotic perception.
- CO5.** Demonstrate knowledge on ethical implications of intelligent machines for providing privacy, trust, security and safety.

CO-PO-PSO Mapping Table:

Course Outcome	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	-	-	-	-	-	-	-	-	-	-	-	-	3
CO2	3	3	2	-	-	-	-	-	-	-	-	-	-	-	3
CO3	3	3	2	-	-	-	-	-	-	-	-	-	-	-	3
CO4	3	-	-	-	-	1	-	-	-	-	-	-	-	-	3
CO5	-	-	-	-	-	1	-	2	-	-	-	-	-	-	3
Course Correlation Mapping	3	2	1	-	-	1	-	2	-	-	-	-	-	-	3

Correlation Level: 3-High; 2-Medium; 1-Low

COURSE CONTENT

Module 1: INTRODUCTION TO ARTIFICIAL INTELLIGENCE (10 Periods)

Foundations of artificial intelligence, History of artificial intelligence, State of the art, Risks and benefits of AI, Intelligent agents – Agents and environments, the concept of rationality, Structure of agents.

Module 2: PROBLEM SOLVING BY SEARCHING (09 Periods)

Problem solving agents, Search algorithms, Uninformed search strategies, Informed search strategies – Greedy best-first search, A* search; Heuristic functions.

Module 3: SEARCHING COMPLEX ENVIRONMENTS (09 Periods)

Local search algorithms and optimization problems – Hill-climbing search, simulated annealing, Local beam search, Evolutionary algorithms; The minimax search algorithm, Alpha-Beta pruning, Move ordering; Monte Carlo tree search.

Module 4: PROBABILISTIC REASONING (09 Periods)

Representing Knowledge in an uncertain domain, Semantics of Bayesian networks, Probabilistic reasoning over time – Time and uncertainty, Inference in temporal models, Hidden Markov models, Kalman Filter.

Module 5: ROBOTICS, ETHICS AND APPLICATIONS (08 Periods)

Robotics: Robots, Robot hardware, Robotic perception, Alternative robotic frameworks, AI applications – Language Models

Ethics and Safety in AI: Limits of AI, Ethics of AI – Surveillance, security and privacy, Trust and transparency, AI safety.

Total Periods:45

EXPERIENTIAL LEARNING

Real-World Case Study Analysis on

1. AI in healthcare
2. AI in finance or e-commerce (credit scoring, dynamic pricing)

RESOURCES

TEXT BOOKS:

1. Zhongzhi SHI, Advanced Artificial intelligence, World Scientific Publishing Co. Pte. Ltd, 3rd edition, ISBN 9789811293986 (hardcover) ,3rd edition.

REFERENCE BOOKS:

1. Stuart Russell, Peter Norvig, Artificial Intelligence: A Modern Approach, 4th (2020) edition, Pearson publications.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, **Deep Learning, MIT Press**

VIDEO LECTURES:

1. <https://www.youtube.com/watch?v=1HpYwa52LeY>
2. <https://www.youtube.com/watch?v=kOkehUZrjBM>
3. https://www.youtube.com/playlist?list=PLxf3-FrL8GzRALeq_9BtdQclN6SF4bTCG

WEB RESOURCES:

1. <https://people.engr.tamu.edu/guni/csce625/slides/AI.pdf>
2. <https://dokumen.pub/advanced-artificial-intelligence-3nbsped-9789811293986-9789811293993-9789811294006.html>
3. <https://nou.edu.ng/coursewarecontent/CIT%20903%20Advanced%20Artificial%20Intelligence0.pdf>
4. <https://home.schoolnutritionandfitness.com/index.jsp/uploaded-files/M2E709/AdvancedArtificialIntelligenceBook.pdf>

Course Code	Course Title	L	T	P	S	C
CA25UG0112L	INTERNET OF THINGS	3	-	-	-	3
Pre-Requisite	-					
Anti-Requisite	-					
Co-Requisite	-					

COURSE DESCRIPTION: Basic concepts of IoT, evolution, characteristics, advantages and disadvantages of IoT, ADC and DAC Principle, IoT sensors and actuators, working with arduino board, applications of IoT in different domains.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

CO1. Understand the basic concepts of IoT to work with IoT environment.

CO2. Analyze the principles of ADCs and DACs to implement in the development of devices.

CO3. Design IoT devices with sensors and actuator technologies for effective results.

CO4. Make use of the Arduino board to develop IoT devices for different domains.

CO5. Develop various IoT related used cases and applications to solve real world problems.

CO-PO-PSO Mapping Table

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO4	PO 5	PO6	PO7	PO 8	PO 9	PO1 0	PO11	PO1 2	PSO 1	PSO 2	PSO 3
CO1	3	3	3	3	-	-	-	-	-	3	-	-	3	3	-
CO2	3	3	3	3	-	-	-	-	-	3	3	3	3	3	-
CO3	3	3	2	3	-	-	-	-	-	3	-	-	3	3	-
CO4	3	3	3	3	-	-	-	-	-	3	3	3	3	3	-
CO5	3	3	3	3	-	-	-	-	-	3	3	3	3	3	3
Course Correlation Mapping	3	3	3	3	-	-	-	-	-	3	3	3	3	3	3

Correlation Levels: 3: High; 2: Medium; 1: Low

COURSE CONTENT

Module 1: BASICS OF IOT

(08 Periods)

Definition of "Internet of Things", Technological trends which have led to IoT, impact of IoT on society, History of IOT, How IOT works, IOT Applications, Characteristics of IoT, Challenges of IoT, Advantages of IoT, Disadvantages IOT, evolution of IOT.

Module 2: ADCS AND DACS

(11 Periods)

DAC principle, weighted register DAC, R L1,L2,L3 -2R ladder DAC, performance characteristics of DAC, ADC principle, flash ADC, successive approximation ADC, counter type ADC, dual slope ADC, performance characteristics of ADC.

Module 3: IOT SENSORS AND ACTUATORS

(09 Periods)

Various IOT Sensors and actuators and technologies, Temperature sensors Moisture sensors, Light sensors, Acoustic and noise sensors, Water level sensors, Proximity sensors, motion sensors, Gyroscope, Chemical sensors, Image sensors.

Module 4: ARDUINO BOARD

(07 Periods)

Introduction to Arduino board, interfacing sensors and actuators to Arduino board.

Module 5: IOT USE CASES AND APPLICATIONS

(10 Periods)

Smart homes, wearables, smart city, smart grid, industrial internet, connected cars, connected health, smart retail, smart supply chain, smart farming.

Total Periods: 45

EXPERIENTIAL LEARNING

1. Compare and contrast different communication protocols used in IoT.
2. Discuss how sensor data is collected, processed, and transmitted to a central hub.
3. Identify potential security risks in an IoT ecosystem and propose strategies to mitigate these risks.

RESOURCES

TEXT BOOKS:

1. Internet of Things: A Hands-on Approach Arshdeep Bahga and Vijay Madisetti Universities Press 2nd Edition.
2. Internet of Things: Architecture and Design Principles Rajkamal McGraw Hill Education 1st edition.

REFERENCE BOOKS:

1. IoT Fundamentals Networking Technologies, Protocols and Use Cases for Internet of Things David Hanes , Gonzalo salgueiro Cisco Press Kindle Edition 2017
2. Designing the Internet of Things Adrian McEwen, Hakim Cassimally Paperback 1st edition
3. Analytics for the Internet of Things(IoT) Andrew Minter Kindle Edition 1st edition

VIDEO LECTURES :

1. [http://www.nptel.ac.in/\(https://youtu.be/WUYAjxnwjU4?list=PLaxu2gn9WXMf_In5pMvxjf043jzof4-i&t=13\)](http://www.nptel.ac.in/(https://youtu.be/WUYAjxnwjU4?list=PLaxu2gn9WXMf_In5pMvxjf043jzof4-i&t=13))
2. [http://www.iitk.ac.in/ \(https://youtu.be/p7kYStiASLo?list=PLbRMhDVUMngdcL\)](http://www.iitk.ac.in/(https://youtu.be/p7kYStiASLo?list=PLbRMhDVUMngdcL))

WEB RESOURCES:

1. [http://www.nptel.ac.in/ \(https://nptel.ac.in/courses/108108098/\)](http://www.nptel.ac.in/(https://nptel.ac.in/courses/108108098/))
2. [http://www.edureka.com/ \(https://youtu.be/LIhmzVL5bm8?list=PL9ooVrP1hQOGccf\)](http://www.edureka.com/(https://youtu.be/LIhmzVL5bm8?list=PL9ooVrP1hQOGccf))