LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY
(An Autonomous Institution)
DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF INSTRUCTION & EXAMINATION [LR-24]
(With effect from the Academic Year 2025-26)
B.E. III-Semester

S. No.	Course Code	Course Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/ D	Contact Hours/Week	CIE	SEE	Duration in Hours	
Theory Course											
1	U24EN301	HSMC	English For Technical Communication	2	-	-	2	40	60	3	2
2	U24EC304	ESC	Digital Electronics and Computer Organization	3	-	-	3	40	60	3	3
3	U24MA301	BSC	Mathematics-III (Probability and Statistics)	3	-	-	3	40	60	3	3
4	U24CS301	PCC	Discrete Mathematics	3	-	-	3	40	60	3	3
5	U24CS302	PCC	Data Structures#	3	-	-	3	40	60	3	3
Practical/ Laboratory Course											
6	U24EN3L1	HSMC	Soft Skills and Employability Skills Lab	-	-	3	3	25	50	3	1.5
7	U24EC3L4	ESC	Digital Electronics Lab	-	-	3	3	25	50	3	1.5
8	U24CS3L1	PCC	Data Structures Lab	-	-	3	3	25	50	3	1.5
Bridge Courses*											
9	U24CS3L2	ESC	C Programming Lab	-	-	2	2	50	-	-	-
10	U24EN1L1	HSMC	Effective Communication Skills Lab	-	-	2	2	50	-	-	-
Mandatory Course*											
11	U24EN102	MC	Indian Constitution	2	-	-	2	40	60	3	-
Total				**14 (16*)**	**-**	**9 (13*)**	**23 (29*)**	**275 (415*)**	**450 (510*)**	**-**	**18.5**

**** Only for Lateral Entry Admitted Students.***

 # Term work (includes assignments, seminars, micro projects, industrial visits, any other student activities etc.)

L: Lecture *(Hrs/Wk/Sem)* **T**: Tutorial *(Hrs/Wk/Sem)* **P**: Practical / **D**: Drawing *(Hrs/Wk/Sem)*
CIE: Continuous Internal Evaluation **SEE**: Semester End Examination (Univ.Exam)
BSC: Basic Science Course **ES**: Engineering Science **MA**: Mathematics
HSMC: Humanities and Social Sciences **PC:** Programme Core Course **EN:** English
IT: Information Technology **EC:** Electronics and Communication Engineering

Note: Each contact hour is a Clock Hour. The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

Course Code	Course Title						Core/Elective
U24EN301	**ENGLISH FOR TECHNICAL COMMUNICATION**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Basic Physics	**2**	**0**	**0**	**0**	**40**	**60**	**2**

Course Objectives

To expose the students to:

1. Understand the significance of Technical Writing.
2. Different types of official correspondence.
3. Various styles of technical report writing.
4. Designing, creating and developing technical manual.
5. Familiarize with the technical features of information transfer.

Course Outcomes

On successful completion of the course, the students would be able to:

1. Apply technical communication skills effectively.
2. Adapt different types of official correspondence successfully.
3. Construct report writing productively using various techniques.
4. Develop the skills of manual writing adequately.
5. Interpret the information transfer from verbal to non-verbal data and vice-versa completely.

UNIT-I

Definition and Features of Technical communication: Definition, Types and Process of Communication, Definition and features of technical communication (precision, relevance, format, style, use of visual aids), Difference between general writing and technical writing, Types of technical communication.

UNIT-II

Technical Writing-I (Official correspondence): Emails, IOM, Business letters (Requisition, Complaint, Inquiry, Order), Business Proposals, Preparation of Minute of Meeting.

UNIT-III

Technical writing-II (Reports): Definition, Importance, Features of Report, Types of Reports- Manuscript, Feasibility report, Project report, Progress report, Evaluation report.

UNIT-IV

Technical writing-III (Manuals): Types of manuals, User manual, Product manual, Operation manual.

UNIT-V

Information Transfer and Presentations: Non-verbal (graphs, tables, flow chart, pie chart, tree diagram, pictogram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oral and visual presentations.

Reference Books:

1. Kumar, Kulbhushan, (2019), *Effective Communication Skills,* Khanna Publishing House.
2. Raman, Meenakshi & Sharma, Sangeeta. (2017). *Technical Communication: Principles and Practice*, OUP (3rd Ed.), New Delhi.
3. Rizvi, Ashraf, M. (2018). *Effective Technical Communication* (2nd Ed.). Tata McGraw Hill Education. New Delhi.
4. Sharma, R. C., & Mohan, Krishna. (2017). *Business Correspondence and Report Writing: A Practical Approach to Business &Technical Communication* (4th Ed.). Tata McGraw Hill Education. New Delhi.
5. Tyagi, Kavita & Misra, Padma, (2011), *Advanced Technical Communication*. New Delhi, PHI Learning.

Course Code	Course Title						Core/Elective
U24EC304	DIGITAL ELECTRONICS & COMPUTER ORGANIZATION						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Basic Physics	3	0	0	0	40	60	3

Course Objectives:

The objectives of this course are

1. To understand the basic building blocks of digital hardware and various minimization techniques.
2. To analyze and design the Combinational and Sequential circuits.
3. Describe the basic structure and operation of digital computer and understand various memory types

Course Outcomes:

On Successful completion of this course, student will be able to

1. Demonstrate the number system conversions and simplify Boolean functions.
2. Analyze and simplify Boolean expressions using Karnaugh-maps, tabulation method and design combinational circuits.
3. Analyze and design various Sequential circuits.
4. To illustrate the operation of digital computer and to understand its organization.
5. Understand the various memory types.

UNIT- I:

Basic Structure of Computers: Basic Structure of Computers, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers.

Data Representation: Binary Numbers Fixed Point Representation. Floating – Point Representation. Number base conversions, Octal and Hexadecimal Numbers, complements, Signed binary numbers, Arithmetic addition and subtraction, Binary codes.

UNIT- II:

Switching Theory& Digital Logic Circuits :Boolean Algebra and switching functions: Basic Logic Functions, Logic gates, universal logic gates, reducing Boolean Expressions, Minimization of switching functions – SOP, POS forms, K-Maps – Two, Three and Four variable.

Combinational Circuits – Half adder, Full adder. Sequential Circuits - Flip-flops Registers, Shift Registers, Binary counters, Decoders.

UNIT- III:

Computer Arithmetic & Basic processing unit: Algorithms for multiplication and division operations. Instruction Set & Instruction Formats, Addressing Modes, Basic Machine Instructions.

Register Transfer Language and Micro operations: Register Transfer Language, Register Transfer, Arithmetic, Logic, Shift Micro operations.

Microprogrammed control: Control memory, address sequencing, micro program example

UNIT- IV:

Input/ Output & Memory Organization :Input Output Organization: Peripheral devices, Input-output Interface, Asynchronous Data Transfer, Modes of Transfer - Priority interrupt, Direct Memory Access.Concept of Memory, RAM, ROM memories, memory hierarchy, cache memory and Mapping, secondary storage.

UNIT-V:

Pipelining:Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics. Pipelining: Basic Concepts, Data Hazards, Instruction Hazards, Influence on Instruction sets, Data path and control considerations, Super Scalar Operation

Suggested Readings :

1. M.Moris Mano ,*Computer Systems Architecture* , IIIrd Edition, Pearson
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky , *Computer Organization* , fifth edition, McGraw Hill.
3. A Anand Kumar ,*Switching Theory and Logic Design* –PHI
4. Miles Murdocca, Vincent Heuring, *Computer Architecture and Organization- An Integrated Approach,* Second Edition, Wiley India.
5. William Stallings, *Computer Organization and Architecture,* Sixth Edition, Pearson .

Course Code	Course Title						Core / Elective
U24MA301	MATHEMATICS – III (PROBABILITY AND STATISTICS) Branch -CSE(AIML), IT, AIML, ECE & CSE						Core
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Basic Maths	3	-	-	-	40	60	3

Course Objectives

The objective of the course is to:

1. Introduce the basic concepts of probability and statistics in engineering
2. Provide an overview of concepts of probability and statistics to engineers
3. Provide the knowledge of probability distributions, tests of significance
4. Acquire the concepts of curve fitting, correlation and regression.
5. Familiar with the concept of tests of hypothesis for decision making

Course Outcomes

After completing this course, the student will be able to:

1. Determine Probability, Random variables, distributions and its application
2. Apply the knowledge of some standard discrete probability distributions and moments
3. Calculate parameters of standard continuous probability distributions.
4. Find the parameters and concepts of correlation, regression and obtain the knowledge of sampling Theory with context to test of hypothesis.
5. Analyze and check the validity of statement using testing of hypothesis for various parameters and goodness of fit.

Unit-I

Introduction of Probability, Conditional probability, Baye's Theorem and its applications, Random variables, Types of random variables, Probability mass function and Probability density function, Mathematical expectations.

Unit-II

Discrete probability distributions: Binomial and Poisson distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions.

Unit-III

Continuous probability distributions, Uniform, Exponential and Normal distributions, Mean, varianceandevaluationofstatisticalparametersforthesedistributions.

Unit-IV

Curve fitting by the method of least squares: fitting of straight lines, second degree parabolas and more general curves, Correlation, regression and Rank correlation. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means.

Unit-V

Small Sample test for single mean, difference of means, test for ratio of variances, Chi-square test for goodness off it and independence of attributes, Low Rank Matrix, Singular Valued Decomposition (SVD).

Suggested Reading:

1. Advanced Engineering Mathematics, R.K.Jain & Iyengar, Narosa Publications.
2. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers,2000.
3. Engineering Mathematics,P.Sivaramakrishna Das &C. Vijaya Kumar, Pearson India
a. Education Services Pvt.Ltd.
4. Engineering Mathematics, SS Sastry, PHI Learning, Private Limited
5. Fundamentals of Mathematical Statistics, S.C.Gupta & V.K.Kapoor, S.ChandPub.
6. P.G.Hoel, S.C.Portand C.J.Stone, "Introduction to Probability Theory", Universal BookStall, 2003.
7. W.Feller,"An Introduction to Probability Theory anditsApplications",Vol.1, Wiley, 1968.
8. N.P. Bali and M.Goyal, "A textbook of Engineering Mathematics", Laxmi Publications

Course Code	Course Title						Core/Elective
U24CS301	DISCRETE MATHEMATICS						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Basic Maths	3	-	-	-	40	60	3

Course Objectives:

Develop ability to

1. Understand concepts of Mathematical Logic, mechanisms of inference rules for propositional and predicate logic and their applications
2. Understand the concepts of Sets, Relations, Functions and their applications.
3. Learn the concepts of Algebraic Structures, basics of counting, Principles of inclusion/exclusion and the pigeonhole methodology.
4. Understand Generating Functions, Recurrence Relations and various ways of solving them.
5. Understand basic definitions and properties of graphs and their applications in computer science and engineering.

Course Outcomes:

At the end of the course, student would be able to

1. Distinguish between Propositional Logic and Predicate Logic, deriving valid proofs of inference and checking the validity of inferences.
2. Illustrate by examples the basic terminology of sets, relations, functions and algebraic structures along with their associated operations.
3. Demonstrate basics of counting, principles of permutations, combinations, applying inclusion/exclusion principle and the pigeonhole methodology in solving counting problems.
4. Demonstrate Generating functions, write recurrence relations and apply various techniques solving recurrence relations.
5. Transform a problem in computer science and engineering as a graph to solve it efficiently using concepts of graph theory.

UNIT-I

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence implication, Normal forms, Quantifiers, universal quantifiers.Predicates: Predicative logic, Free & Bound variables, Rules of inference, Consistency, proof of contradiction, Automatic Theorem Proving.

UNIT-II

Relations: Properties of Binary Relations, equivalence, transitive closure, compatibility and partial ordering relations, Lattices, Hasse diagram. Functions: Inverse Function Composition of functions, recursive Functions, Lattice and its Properties.

UNIT-III

Elementary Combinatorics: Basis of counting, Combinations & Permutations, with repetitions,Constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, the principles of Inclusion – Exclusion. Pigeon hole principles and its application.

UNIT-IV

Recurrence Relation: Generating Functions, Function of Sequences Calculating Coefficient of generating function, Recurrence relations, Solving recurrence relation by substitution and Generating funds. Characteristics solution of inhomogeneous Recurrence Relation.

UNIT-V

Graph Theory: Basic Concepts, Representation of Graph, Isomorphism, Sub graphs, Spanning Trees, Planar Graphs, Multi graphs, Euler circuits, Hamiltonian graphs, Chromatic Numbers. Graph Theory and Applications.

Suggested Readings:

1. Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay, R. Manohar, McGraw Hill education (India) Private Limited. (UNITS - I , II)
2. Discrete Mathematics for Computer Scientists & Mathematicians, Joe L. Mott, Abraham Kandel, Theodore P. Baker, Pearson, 2nd ed. (Units - III, IV, V)
3. Elements of Discrete Mathematics- A Computer Oriented Approach- C L Liu, D P Mohapatra. Third Edition, Tata Mc Graw Hill.
4. Discrete Mathematics and its Applications, Kenneth H. Rosen, Fifth Edition.TMH.
5. Discrete Mathematical Structures Theory and Application-Malik & Sen, Cengage.

Course Code	Course Title						Core/Elective
U24CS302	DATA STRUCTURES						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS	3	-	-	-	40	60	3

Course Objectives:

Develop ability to

1. Develop skills to design and analyze simple linear and nonlinear data structures, such as stacks, queues and lists and their applications.
2. Gain programming skills to implement sorting and searching algorithms
3. Strengthen the ability to identify and apply the suitable data structures for the given real world problem.
4. Gain knowledge in practical applications of data structures
5. Understand essential for future programming and software engineering courses.

Course Outcomes:

At the end of the course, student would be able to

1. Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem
2. Implement various data structures using arrays, linked lists
3. Develop ADT necessary for solving problems based on Stacks and Queues
4. Implement binary trees, general tree structures, advanced search trees, heaps, graphs.
5. Implement hash functions and handle collisions.

UNIT – I

Introduction Data Structures and Algorithms:

Introduction to Data Structures, classification of data structures, operations on data structures; Algorithm Specification, Recursive algorithms, Data Abstraction, Performance analysis- Time Complexity and Space Complexity, Asymptotic Notation-Big O, Omega, and Theta notations.

UNIT – II

Searching Techniques: Linear search, Binary Search algorithms.

Sorting Techniques: Bubble Sort, Insertion sort, Selection Sort, Merge Sort, and Quick Sort. Comparison among sorting techniques.

Stacks: Stack ADT, definition and operations, Implementations of stacks using array, applications of stacks, Arithmetic expression conversion and evaluation.

UNIT - III

Queues: Queue ADT, definition and operations, Implementation of queues using Arrays, applications of linear queue, circular queue and double ended queue (deque).

Linked Lists: Introduction, Single linked list, representation of a linked list in memory, operations on a Singly Linked List, Implementation of Singly Linked List.

UNIT – IV

Doubly Linked Lists: Operations on Doubly Linked List, Implementation of Doubly Linked List, Circular linked list, Implementation of Stack and Queue using linked list.

Hashing: Static Hashing, Hash Tables, Hash Functions, Overflow Handling, Theoretical Evaluation of Overflow Techniques

UNIT – V

Trees: Introduction, Binary Trees, Tree Traversals, Threaded Binary Trees, Binary Search Tree, Heap Tree, AVL Tree.

Graphs: Graph Abstract Data Type, Representation of Graph, Graph Traversals -DFS and BFS, Spanning Tree, Prim‘s and Kruskal‘s Algorithms.

Suggested Readings:

1. “Fundamentals of Data Structures in C”, Ellis Horowitz, Sartaj Sahani, Susan Anderson Freed, Computer Science Press, 2007
2. S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2014.
3. D. Samanta, “Classic Data Structures”, PHI Learning, 2nd Edition, 2004.
4. Mark A Weiss, “Data Structures and Algorithm Analysis In C”, Second Edition (2002), Pearson
5. “Data Structures and Algorithms in C++”, Second Edition, 2011 by Michael T. Goodrich and Roberto Tamassia.

Course Code	Course Title						Core/Elective
U24EN3L1/ U24EN4L1	SOFT SKILLS & EMPLOYABILITY SKILLS LAB [Common to all branches]						Core
Pre-requisites	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	3	25	50	1.5

Course Objectives:
To expose the students to:
1.Apply soft skills at professional level.
2.Foster leadership skill with a mature outlook for effective functioning at work front.
3.Develop confidence through interpersonal skills.
4. Exhibit their ability and skills to write Resume/CV and cover letter
5. Boost the skills of group discussion and interview.

Course Outcomes:
On successful completion of the course the students would be able to:
1.Utilise soft skills at professional level effectively.
2. Function efficiently in multidisciplinary settings by using leadership skills.
3. Build confidence through interpersonal skills utterly.
4. Write Resume/CV and cover letter comprehensively.
5. Enhance the skills of group discussion and interview perfectly.

List of Activities

1. Soft Skills
Introduction to Soft Skills and Types; Time Management, Team work

2. Leadership Skill
Decision Making, Critical Thinking, Conflict Resolution, Adaptability Skills

3. Interpersonal Skills
Stress Management, Emotional Intelligence, Motivation, Presentation Skills

4. Job Skills
Resume/CV writing, Cover letter writing

5. Interview Skills
Dynamics of Group Discussion, Types; Interview, Types, Interview Etiquettes, Mock Interviews,

Suggested Readings:

1. Bhardwaj, Kumkum, (2019), *Fundamentals of Business Communication*, Wiley, India
2. Kapoor Shikha, (2020), *Personality development and Soft Skills-Preparing for Tomorrow*, Wiley India
3. Koneru, Arun, (2017), *Professional Communication*, Tata McGraw-Hill Publishing Company. Ltd, New Delhi
4. Mitra K. Barun. (2016). *Personality Development and Soft Skills*. Oxford University Press.
5. Raman Meenakshi & Sharma Sangeeta, (2017), *Technical Communication: Principles and Practice,* OUP (3rd Ed.). New Delhi.
6. Sharma, Prashant, (2019). *Soft Skills-Personality development for Life Success,* BPB Publications
7. Tyagi, Kavita & Misra, Padma. (2011). *Advanced Technical Communication*, PHI Learning, New Delhi.

Course Code	Course Title					Core/Elective
U24EC3L4	DIGITAL ELECTRONICS LAB					Core
Pre-requisites	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	3	25	50	1.5

Course Objectives:
This course aims to familiarize:
1. Basic operation of Logic gates
2. Operation of Combinational Circuits
3. Flip-Flops, Shift registers and Counters.

Course Outcomes:
Upon completion of this course, students will be able to:
1. Demonstrate the truth table of various logic gates.
2. Design, test and evaluate various combinational circuits such as adders, subtractors, comparators, multiplexers and demultiplexers.
3. Construct flips-flops, counters and shift registers.
4. Simulate full adder and up/down counters.

List of Experiments:
1. To verify truth tables of Logic gates.
2. To design and implement Full Adder using basic logic gates.
3. To design and implement Full subtractor using basic logic gates.
4. To design and implement 4-bit Parallel Adder/ subtractor.
5. Design and Implementation of 4-bit Comparator
6. To realize 4:1 Multiplexer and 1:8 Demux using gates.
7. Realize and 3:8 Decoder.
8. To realize the following flip-flops- (a) SR Flip-Flop (b) JK Flip-Flop.
9. To realize the following flip-flops- (a) T Flip -Flop (b) D Flip-Flop
10. To realize the Ring Counter.

Suggested Reading:
1. R. P. Jain, "Modern Digital Electronics", McGraw Hill Education, 2009.
2. M. M. Mano, "Digital logic and Computer design", Pearson Education India, 2016.
3. A. Kumar, "Fundamentals of Digital Circuits", Prentice Hall India, 2016.

Course Code	Course Title						Core
U24CS3L1	DATA STRUCTURES LAB						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS Lab	-	-	-	3	25	50	1.5

Course Objectives:

Develop ability to

1. Understand essential concepts of simple linear and nonlinear data structures.
2. Apply the suitable data structures for the given real-world problems.
3. Acquire knowledge in practical applications of data structures.
4. Provide solutions for various graphical concepts.
5. Analyze and implement programming skills to implement sorting and searching algorithms.

Course Outcomes:

At the end of the course, student would be able to

1. Perform Sorting and Searching and be able to justify which sorting and searching techniques is suitable
2. Apply stacks and queues in solving problems
3. Evaluate binary trees, general tree structures, advanced search trees, heaps, graphs.
4. Apply hash functions and handle collisions.
5. Implement various kinds of sorting techniques and apply appropriate techniques for solving a given Problem.

List of Experiments:

1. Implementation of Binary Search and Linear Search.
2. Implementation of Selection, Merge, Quick, and InsertionSort.
3. Implementation of Stacks and Queues using Arrays.
4. Implementation of Infix to Postfix Conversion, Postfix ExpressionEvaluation.
5. Implementation of Circular Queue using Arrays.
6. Implementation of Singly Linked List
7. Implementation of Doubly Linked List.
8. Implementation of Circular Linked List.
9. Implementation of Stacks, Queues using Linked Lists.
10. Implementation of Binary Search Tree. (Insertion, Deletion, and Search operations)
11. Implementation of Tree Traversal on Binary Trees.
12. Implementation of AVL Trees.
13. Implementation of Traversal on Graphs.
14. Implementation of Prim's and Kruskal's Algorithm.

Suggested Readings:

1. S. Lipschutz, "Data Structures", Tata McGraw Hill Education, Revised Edition, 2014.
2. D. Samanta, "Classic Data Structures", PHI Learning, 2nd Edition, 2004.
3. Mark A Weiss, Data Structures and Algorithm Analysis In C, SecondEdition (2002), Pearson

LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY
(An Autonomous Institution)
DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF INSTRUCTION & EXAMINATION [LR-24]
(With effect from the Academic Year 2025-26)
B.E. IV-Semester

S. No.	Course Code	Course Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/ D	Contact Hours/Week	CIE	SEE	Duration in Hours	
Theory Course											
1	U24CS401	PCC	Design and Analysis of Algorithms	3	-	-	3	40	60	3	3
2	U24ME404	ESC	Operations Research	3	-	-	3	40	60	3	3
3	U24IT401	PCC	Database Management Systems	3	-	-	3	40	60	3	3
4	U24CD401	PCC	Operating Systems	3	-	-	3	40	60	3	3
5	U24IT402	PCC	Java Programming#	3	-	-	3	40	60	3	3
Practical / Laboratory Course											
6	U24IT4L1	PCC	Database Management Systems Lab	-	-	3	3	25	50	3	1.5
7	U24CD4L1	PCC	Operating Systems Lab	-	-	3	3	25	50	3	1.5
8	U24IT4L2	PCC	Java Programming Lab	-	-	3	3	25	50	3	1.5
9	U24EP4L1	HSMC	Design Thinking Lab	-	-	2	2	25	50	3	1
Total				**15**	**-**	**11**	**26**	**300**	**500**	**-**	**20.5**

 # Term work (includes assignments, seminars, micro projects, industrial visits, any other student activities etc.)

L: Lecture *(Hrs/Wk/Sem)* **T**: Tutorial *(Hrs/Wk/Sem)* **P**: Practical / **D**: Drawing *(Hrs/Wk/Sem)*
CIE: Continuous Internal Evaluation **SEE**: Semester End Examination (Univ.Exam)
IT: Information Technology **PCC:** Programme Core Courses
HSMC: Humanities and Social Sciences

Note: Each contact hour is a Clock Hour. The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

Course Code	Course Title				Core/Elective		
U24CS401	DESIGN AND ANALYSIS OF ALGORITHMS				Core		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Data Structures	3	-	-	-	40	60	3

Course Objectives:

The objectives of this course are:

1. Acquire the knowledge of Algorithm and problem-solving technique, limitations of algorithms.
2. Understand different techniques like divide and conquer, transfer and conquer etc., to solve problems.
3. Understand different techniques like greedy method and dynamic programming
4. Understand the concepts of Np-Hard and Np-Complete.
5. Illustrating the methods of backtracking and branch bound techniques to solve the problems like N-queens problem, graph coloring

Course Outcomes

At the end of the course, the students will be able to:

1. Identify asymptotic notations and basic efficiency classes.
2. Solve problems using various techniques like divide-and-conquer and transfer-and-conquer.
3. Use different algorithms like TSP, Floyd's etc. to solve real world problems.
4. Introduce the P and NP classes.
5. Develop solutions for n - Queens problem, Subset – Sum Problem, Assignment problem, Knapsack problem etc.

UNIT - I

Characteristics of Algorithm, Analysis of algorithm: Time and space Complexity, All six Asymptotic notations, Iterative and recursive algorithms analysis (Code snippets for common time complexities), back-substitution method, recursive tree method, master's theorem. Review of Analysis of Sorting algorithms (Bubble, Quick, Heap, Merge, Insertion, Selection sort) and searching algorithms (Linear, binary).

UNIT - II

Divide and Conquer: General method, finding Max-Min, Strassen's matrix multiplication, Karatsuba method, Closest pair, Convex Hull problem.
Transform and Conquer: Introduction, Balanced search trees.
Decrease and Conquer: Introduction, Computing the Median

UNIT – III

Greedy method: General method, applications-Job sequencing with deadlines, Knapsack problem, Single source shortest path problem, Huffman Coding for compression. Max flow problem, Connected components, Biconnected components. Introduction to Randomized algorithms. Examples of Monte Carlo and Las Vegas algorithms

Unit IV

Dynamic Programming: General method, applications- Matrix chain multiplication, Optimal binary search tree, 0/1 knapsack problem, All pairs shortest path problem, Traveling salesperson problem, multi-stage graph, Warshall's , Transitive Closure, Floyd's Algorithm All pairs shortest algorithm, Reliability design.

UNIT – V

Branch and Bound: General method, applications - Traveling salesperson problem, 0/1 knapsack problem - LC Branch and Bound solution, FIFO Branch and Bound solution.
Backtracking: The general Method, 8-Queens Problem, Graph Coloring, and Hamiltonian Cycle
NP-Hard and NP-Complete Problems: Basic concepts, non-deterministic algorithms, NP-Hard and NP-Complete classes, Cook's theorem. Introduction to Approximate algorithms.

Suggested Readings:

1.Introduction to Algorithms 4 edition by Thomas H Cormen,Charles E Leiserson, Ronald L Rivest, Clifford Stein , MIT Press, 2022.
2.Data Structures and Algorithms Made Easy" by Narasimha Karumanchi Publisher: career monk, 2023
3. Data Structure & Algorithms by Dr. Zeeshan Ul Hassan Usmani, Sana Rasheed Publisher: ilmStore in Urdu Language
4. Introduction to the Design and Analysis of Algorithms, 3/e Anany Levitin Pearson

Course Code	Course Title						Core/Elective
U24ME404	OPERATIONS RESEARCH (Common to CSD, IT & Mech)						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	40	60	3

Course Objectives:
Develop ability to
1. Explain with examples, the basic terminology of functions, relations, and sets.
2. Perform the operations associated with sets, functions, and relations.
3. Relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
4. Describe the importance and limitations of predicate logic.
5. Use Graph Theory for solving problems.

Course Outcomes:
At the end of the course, student would be able to
1. Understand the ideas of mathematical induction to recursion and recursively defined structures.
2. Prepare the students to have the knowledge of Linear Programming Problem in Operations
3. Research at the end students would be able to understand the concept and develop the models for different applications.
4. Make students understand the concept Replacement models at the end students would able to explain various features and applications of replacement models in real time scenario.
5. Prepare the students to understand theory of Game in operations research at the end students would able to explain application of Game theory in decision making for a conflict

UNIT – I

Introduction: Definition and Scope of Operations Research.

Linear Programming: Introduction, Formulation of linear programming problems, graphical method of solving LP problem, simplex method, maximization methods and minimization, Degeneracy in LPP, Unbounded and, Infeasible solutions.

UNIT – II

Transportation Models: Finding an initial feasible solution - North West corner method, least cost method, Vogel‘s Approximation method, Finding the optimal solution, optimal solution by stepping stone and MODI methods, Special cases in Transportation problems - Unbalanced Transportation problem.

Assignment Problems: Hungarian method of Assignment problem, Maximization in Assignment Problem, unbalanced problem, problems with restrictions, travelling salesman problems.

UNIT – III

Network Fundamentals- scheduling the activities -Fulkerson’s Rule –CPM- earliest and latest times - determination of ES and EF in the Forward Pass - LS and LF in backward pass determination of Critical Path, Crashing, time cost trade off. PERT-Beta Distribution, probabilistic models, Calculation of CP, resource analysis and allocation.

UNIT – IV

Replacement Models: Introduction, replacement of items that deteriorate ignoring change in money value, replacement of items that deteriorate considering change in money value with time, replacement of items that fail suddenly - Individual replacement policy, Group replacement policy.

Game Theory: Introduction, 2 person zero sum games, Maximin - Minimax principle, Principle of Dominance, Solution for mixed strategy problems, Graphical method for 2 x n and m x 2 games.

UNIT – V

Sequencing Models: Introduction, General assumptions, processing n jobs through 2 machines, processing n jobs through m machines, Processing 2 jobs through m machines

Queuing Theory: Introduction, single channel - Poisson arrivals - exponential service times with infinite population & finite population, Multi-channel - poison arrivals - Exponential service times with infinite population.

Introduction to Optimization Techniques: Single objective & Multi objective optimization Techniques like G.A, NSGA, P.Q.O & MPSO Techniques.

Suggested Readings:

1. Hamdy, A. Taha, ―Operations Research-An Introduction‖, Sixth Edition, Prentice Hall of India Pvt. Ltd.,1997
2. S.D. Sharma, Operations Research, Kedarnath, Ramnath & Co., Meerut,2009
3. Dr. Mohammed Asif Kattimani, Mr. Shaik Mohammed Ali, Dr. P R Venkatesh, Introduction to Operations Research & Application of Python, Infotech Standards India Pvt. Ltd, 2025.
4. J.B. Gupta, ―Utilization of Electric Power and Electric Traction‖ S.K. Kataria & Sons Publications, 2010 Hrvey M. Wagner, Principles of Operations Research, Second Edition,Prentice Hall of India Ltd., 1980.
5. V.K. Kapoor, Operations Research, S. Chand Publishers, New Delhi,2004
6. R. Paneer Selvam, Operations Research, Second Edition, PHI Learning Pvt. Ltd., New Delhi,2008.

Course Code	Course Title						Core/Elective
U24IT301/ U24IT401	DATABASE MANAGEMENT SYSTEMS						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Data Structures	3	-	-	-	40	60	3

Course Objectives:

Develop ability to

1. Understand the role of database management system in an organization and learn the database concepts.
2. Design databases using data modeling and Logical database design techniques.
3. Construct database queries using relational algebra and calculus and SQL.
4. Understand the concept of a database transaction and related concurrent, recovery facilities.
5. Understand the concepts of Triggers and Stored Procedures.

Course Outcomes:

At the end of the course, student would be able to

1. Design ER-models to represent simple database application scenarios and Construct database queries using SQL.
2. Construct database queries using relational algebra and calculus.
3. Recognize and identify the use of normalization and functional dependency in database design.
4. Apply the concept of a database transaction and related concurrent, recovery facilities
5. Apply and relate how to evaluate a set of queries in query processing.

UNIT - I

Conceptual Modeling Introduction: Introduction to Data bases: Purpose of Database systems, view of data, data models, Database languages, Database users, various components of overall DBS architecture, various concepts of ER model, basics of Relational Model.

SQL Query – BASICS:

SQL – Data Definition commands, Queries with various options, Data manipulation commands, Views, Joins, views, integrity and security.

UNIT – II

Relational Approach

Relational algebra and calculus: Relational algebra, selection and projection, set operations, renaming, joins, division, examples of algebra queries, relational calculus: Tuple relational calculus, Domain relational calculus, expressive power of algebra and calculus.

UNIT - III

Introduction to NoSQL: Introduction, Overview and History of NoSQL Databases – The Definition of the Four Types of NoSQL Databases, differences between SQL and NoSQL .

Normalization: Pitfalls of RDBD, Lossless join decomposition, functional dependencies, Armstrong axioms, normalization for relational databases 1st, 2nd and 3rd normal forms, BCNF, Basic definitions of MVDs and JDs, 4th and 5th normal forms.

UNIT - IV

Transaction Management: Transaction processing: Transaction concept, transaction State, implementation of atomicity and durability, concurrent executions, serializability, recoverability. Concurrency Control: Lock-based protocols, timestamp-based protocols, validation-based protocols, multiple granularities, multi-version schemes, deadlock handling.

UNIT – V

Data Storage: Overview of physical storage media, magnetic disks, storage access, file organization, organization of records in files.
Indexing and Hashing: Basic concepts, types of indexing, difference between B and B+ Indexing, static hashing, Dynamic Hashing.

Suggested Readings:

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw- Hill, 7th Edition, 2019.
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamental Database Systems", Pearson Education, 6th Edition, 2014.
3. Raghu Ramakrishnan, "Database Management System", Tata McGraw-Hill Publishing ompany, 3rd Edition, 2007.
4. Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, "Database System Implementation", Pearson Education, United States, 1st Edition, 2000.
5. Peter Rob, Corlos Coronel, "Database System, Design, Implementation and Management", Thompson Learning Course Technology, 5th Edition, 2003.

Course Code	Course Title						Core/Elective
U24CD401, U23CD503	OPERATING SYSTEMS						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS	3	-	-	-	40	60	3

Course Objectives:

1. To understand the services provided by and the design of an operating system.
2. To understand the structure and organization of the file system.
3. To understand what a process is and how processes are synchronized and scheduled.
4. To understand different approaches to memory management.
5. To Understand Virtual machine concepts, calls for managing processes, memory and the file system

Course Outcomes:

At the end of the course, the students will be able to:

1. Understand the fundamental concepts and Functions of operating system.
2. Analyze various scheduling algorithms.
3. Understand deadlock, prevention and avoidance algorithms.
4. Compare and contrast various memory management schemes.
5. Understand the functionality of file systems and perform administrative tasks on Linux Servers

UNIT-I

Introduction: What Operating system do, Defining operating system, Computer System Operation, Storage Structure, Operating system Structure, Operating System Operations, Computer system architecture: Single Processor, Multi-Processor, Multiprogramming, Multitasking, Process, Memory and Storage Managements, Protection and Security.
System Structures: System calls, Types of System Calls, System Programs, System boot

UNIT-II

Processes: Process concepts, Process Scheduling, Operations on Processes, Inter process communication, Communication in Client/Server Systems
CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, Priority, Round Robin) **Synchronization**: Background, The Critical-Section Problem, Peterson‟s Solution, Synchronization, Dining Philosophers problem Monitors. **Deadlocks**: System Model, Deadlock characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT-III

Memory-Management Strategies: Background, Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.
Virtual Memory Management: Background, Demand paging, Copy-on-write, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory.
Storage Management: File System, File Concept, Access Methods, Directory Structure, File-System Mounting, File sharing, Protection.

UNIT-IV

Implementing File Systems: File System-Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery, Log- Structured File Systems, NFS.
Secondary Storage Structure: Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, RAID Structure, Stable-Storage Implementation, Tertiary-Storage Structure.

UNIT-V

Protection and Security: Goals of Protection, Principles of Protection, Domain of protection, Access Matrix, Implementation of Access Matrix, Access control, Revocation of access rights,
System Security: The security problem, program Threats, System and System Network Threats, Cryptography as a Security tool, User Authentication, Implementing Security Defences, firewalling to protect Systems and Networks, Case Studies- Linux System

Suggested Readings:

1. Modern Operating Systems by Andrew S. Tanenbaum& Herbert Bos , 5th Edition 2022
2. Operating System Concepts, Siliberchatz, Galvin & Gagne, 10th Edition 2023.

Course Code	Course Title						Core/Elective
U24IT402	JAVA PROGRAMMING						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Python Programming	3	-	-	-	40	60	3

Course Objectives:

Develop ability to

1. Learn fundamentals of object-oriented programming in Java which includes defining classes, invoking methods, difference between applet and application programs,using class libraries
2. Create Java application programs using sound OOP practices such as interfaces and inheritance
3. Analyze the concepts of packages, exceptions.
4. Use Knowledge of Collection framework, AWT and event handling to solve real-world problems.
5. Apply the concepts of JDBC to organize data efficiently.

Course Outcomes

At the end of the Course, Student would be:

1. Familiarization of OOP concepts and basics of java programming.
2. Describe the concept of interfaces and inheritance, how to solve real world problems.
3. Choose a suitable package to develop the inter process communication using multithreading.
4. Build GUI applications using AWT and Swings.
5. Describe the connectivity to database and java programming using JDBC Connectivity.

UNIT – I

Object Oriented Programming and principles: definition of OOP, Benefits of Object Oriented Programming. **Introduction to Java:** Java buzzwords, byte code. Java Programming Fundamentals, data types, variables, arrays, operators, expressions, control statements, concepts of classes, objects, constructors, methods, access control, overloading methods, and constructors, static, final, exploring string class.

UNIT – II

Inheritance: Inheritance concept, types of inheritance, Member access rules, use of super and final.
Polymorphism - dynamic binding, method overriding, abstract classes and methods.
Interfaces: Defining an interface, implementing interfaces, extending interfaces.
Packages: Defining, Creating and Accessing a Package, importing packages

UNIT – III

Exception handling: Benefits of exception handling, classification, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, built-in exceptions, creating own exception sub classes.
Multithreading: Java Thread Model-life cycle, The Main Thread, creating a Thread, creating multiple threads, using is Alive() and join(), thread priorities, synchronization, inter-thread communication, deadlock.
Collections: Overview of Java Collection frame work, commonly used Collection classes- ArrayList, LinkedList, HashSet, TreeSet. Iterator, Working with Maps. Legacy classes and interface.

UNIT –IV

Other Utility classes: String Tokenizes, Scanner Java Input/output: exploring java.io, Java I/O classes and interfaces, File, Stream classes, byte stream, character stream, serialization.
GUI Programming with Java: The AWT class hierarchy, MVC architecture. AWT classes and Interfaces, Layout Manager, Applet Revisited: Basics, architectureand skeleton, simple applet program.
Event Handling: Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces. Handlingmouse and keyboard events and adapter classes.

UNIT – V

Exploring Swings: JLabel, JTextField, JcomboBox. **Servlet :** Introduction**,** Servlet Lifecycle, **JSP**: Introduction, JSP Lifecycle**,** JSP vs servlet.

Database Programming using JDBC: Introduction to JDBC, JDBC Drivers & Architecture, CURD operation Using JDBC.

Suggested Readings:

1.Herbert Schildt, Dr.Danny Coward "Java, The Complete Reference", 13th Edition, Tata McGraw Hill,2021.
2.James M Slack, "Programming and Problem Solving with JAVA", Thomson Learning,2023.
3.C Thomas Wu, "An Introduction to Object Oriented Programming with Java", 5th Edition, McGraw Hill Publishing,2010.
4. Paul Dietel and Harvey. Dietel, "Java How to Program", 11th Edition, Pearson Education/PHI.

Course Code	Course Title						Core/Elective
U24IT3L1/ U24IT4L1	DATABASE MANAGEMENT SYSTEMS LAB						Core
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving Lab	-	-	-	3	25	50	1.5

Course Objectives:

Develop ability to

1. Learn ER data model, database design and normalization.
2. Learn SQL basics for data definition and data manipulation.
3. Understand the basic concepts and the applications of database systems.
4. Be acquainted with the basics of transaction processing and concurrency control.
5. Learn the concepts of Views, Stored Procedure and Triggers.

Course Outcomes

At the end of the course, student would be able to

1. Design database schema for a given application and apply normalization.
2. Gather skills in using SQL commands for data definition and data manipulation.
3. Demonstrate creation and usage of Views and Stored Procedures using SQL.
4. Develop solutions for database applications using procedures, cursors and triggers
5. Design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

LIST OF EXPERIMENTS

Scenario: Product-Sales database: South wind database is a sample database used by Organization. The database contains the sales data for South wind Traders; it is foods export-import Company. Using this schema to demonstrate the how customer can choose and order products, how orders are placed and how those products get delivered to the customer. Products: This Entity will have all the products details where suppliers will supply products based on customers demand. Supplies: This Entity will supply the products demanded by the customers. Shippers: This Entity will take the orders from suppliers and deliver to customers. Employees: Employees will monitor the orders placed by customers. Invoices: This Entity will take care of billing process based on customer order. Etc...Identify some more entities and find out relationship between them. Product-sales the above process involves many steps like

1. Analyzing the problem and identifying the Entities and Relationships,
2. E-R Model
3. Relational Model
4. Normalization
5. Creating the database
6. Querying.

Experiment 1: E-R Model

Analyze and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc. Identify the primary keys for all the entities. Identify the other keys like Foreign Key and constraints like NULL, NOT NULL, CHECK etc. Example to create for products, customers, suppliers, orders, , employees, order details, categories, among others. Students should submit E-R diagrams using the above tables.

Experiment 2: DDL

How to create tables, altering the database or tables, dropping tables if not required. You will also try truncate, rename commands etc. Data Definition Language (DDL) : create , alter, drop.

Experiment 3: DML

Data Manipulation Language Commands (DML) commands are used to for managing data within schema objects. Exercising the commands using DML: insert, delete, update on the following tables : products, customers, suppliers, orders, , employees, order details, categories.

- INSERT – insert data into a table.
- UPDATE – updates existing

Experiment 4: Querying

data within a table.

- DELETE – deletes single or all records from a table.

Data Query Language – Select Populate all the tables designed in experiment: 2 with appropriate data.

Practice queries on Aggregate functions like count, max , min ,avg ,sum Practice queries like nested queries/co-related queries using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, groupby ,having etc.

Joins: Join , Left Outer Join, Right Outer Join, Self Join

Experiment 5 : Querying(continued…)
Some example to practice the queries:
1. Display all the order details of given a customer.
2. Display all the products.
3. Get the highest sold product from given supplier ID
4. List all products grouped by category
5. List the products, whose products unit price is greater then all the products of average.
6. List Details of order and customer of each order
7. List the products which were sold in year 1997
8. Display the total amount for each order
9. Display Order Details for given an order ID
10. Order Details: product name and unit price for given order ID Exercising Simple to complex
11. Queries using joins, nested and co-related queries.

Experiment 6 : Programs on Pl/Sql
1. WriteaPL/SQL program to swap two numbers.
2. WriteaPL/SQL program to find the largest of three numbers
3. WriteaPL/SQLprogramtofindthetotalandaverageof6subjectsanddisplaythegrade.
4. WriteaPL/SQL program to find the sum of digits in a given number.
5. WriteaPL/SQL program to display the number in reverse order.
6. Write a PL/ SQL program to check whether the given number is prime or not.
7. WriteaPL/SQL program to find the factorial of a given number.

Experiment 7 : Stored Procedures :
1. Create a stored procedure, Alter and Drop a procedure, IN, OUT, IN & OUT parameters
2. Create a Procedure to display order details of given customer ID like ordered, order Date ,
Required Date, Shipped Date
3. Create a procedure to accept a customer ID and display the customer order history (product name and how much quantity ordered for that particular product)
Ex: product name, Total quantity he/she ordered.
4. Create a procedure to display Ten Most Expensive Products Columns should be displayed
Product name & Unit price

Experiment 8: Views
1. Create a view to display the current product list which is available (not discontinued)
2. Create a view to display the products by category
3. Display product name, quantity Per Unit, units In Stock, Discontinued
4. Create a view as ―Invoices‖ to display all the information from order, customer, and shipper for each Order Details

Experiment 9: Triggers
Demonstrate Create Trigger, Alter Trigger, Drop Trigger, Row Level, Table Level triggers, Before Insert, After Insert, Before Update, After Update, Before Delete, After Delete

Experiment 10: Case study: Book Publishing Company
A publishing company produces scientific books on various subjects. The books are written by authors who specialize in one particular subject. The company employs editors who, not necessarily being specialists in a particular area, each take sole responsibility for editing one or more publications.
A publication covers essentially one of the specialist subjects and is normally written by a single author. When writing a particular book, each author works with on editor, but may submit another work for publication to be supervised by other editors. To improve their competitiveness, the company tries to employ a variety of authors, more than one author being a specialist in a particular subject for the above case study, do the following:
a. Analyze the data required.
b. Normalize the attributes.
Create the logical data model using E-R diagrams.

Experiment 11: Case Study: General Hospital
A General Hospital consists of a number of specialized wards (such as Maternity, Pediatric, Oncology, etc). Each ward hosts a number of patients, who were admitted on the recommendation of their own GP and confirmed by a consultant employed by the Hospital. On admission, the personal details of every patient are recorded. A separate register is to be held to store the information of the tests undertaken and the results of a prescribed treatment. A number of tests may be conducted for each patient. Each patient is assigned to one leading consultant but may be examined by another doctor, if required. Doctors are specialists in some branch of medicine and may be leading consultants for a number of patients, not necessarily from the same ward. For the above case study
For the above case study, do the following.
a. Analyze the data required.
b. Normalize the attributes.
Create the logical data model using E-R diagrams.

Suggested Readings
1. Raghurama Krishnan, Johannes Gehrke, ―Database Management Systems‖, Tata McGraw Hill, 4th Edition, 2018.
2. Silberschatz, Korth, ―Database System Concepts‖, McGraw Hill, VI edition, 2014.
3. M. Mc Laughlin, ―Oracle Database 11g PL/SQL Programming, TMH, 2017
4. Dr. P. S. Deshpande, ―SQL & PL/SQL for Oracle 10g‖, Black Book, Dream Tech, 2006.

Course Code	Course Title						Core/Elective
U23CD5L3, U24CD4L1	OPERATING SYSTEMS LAB						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS	-	-	-	3	25	50	1.5

Course Objectives:

1. Work with unix/linux commands.
2. Learn working with files and use file access permissions.
3. Implement process scheduling algorithms & deadlock management.
4. Implement page replacement algorithms.
5. Implement disk scheduling algorithms.

Course Outcomes:

At the end of the course, student would be able to

1. Work with UNIX commands and Shell Programming.
2. Implement and analyze the performance of different algorithm of operating.
3. Implement CPU scheduling algorithm,
4. Implement Page Replacement Algorithm
5. Implement Deadlock handling mechanism.

List of Experiments

1. Practice of Linux Commands.
i) File Commands
ii) Process Management commands
iii) File Permissions
iv) System Information
2. Implementation of Fork() system call of Unix operating system.
3. Implementation of two process communication for the following
i) Pipes ii) Shared Memory
4. Implementation of Processor Scheduling Algorithms for the following
i) FCFS ii) SJF iii) Priority iv) Round Robin
5. Implementation of Producer Consumer Problem
6. Implementation of Dining Philosophers problem.
7. Simulation of Bankers algorithm for deadlock avoidance
8. Simulation of Bankers algorithm for deadlock Prevention.
9. Implementation of Page Replacement Algorithm for the following
i) FIFO ii) LRU
10. Implementation of Disk Scheduling Algorithm for the following
i) FCFS ii) SCAN iii) C-SCAN

Suggested Readings:

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Principles, Ninth Edition, John Wiley & Sons Publication,2012
2. The Design of Unix Operating System, Maurice Bach, Prentice Hall.

Course Code	Course Title						Core/Elective
U24IT4L2	JAVA PROGRAMMING LAB						Core
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Python Programming	-	-	-	3	25	50	1.5

Course Objectives:

Develop ability to

1. Build software development skills using java programming for real world applications.
2. Implement frontend and backend of an application
3. Create Java application programs using sound OOP practices such as interfaces,exception handling multithreading.
4. Understand fundamentals of object-oriented programming in Java.
5. Implement classical problems using java programming.

Course Outcomes

After completing this course, the student will be able to:

1. Develop Java applications using the concepts of Inheritance, interfaces, packages,access control specifies.
2. Implement the concepts of Exception Handling in java Applications.
3. Read and write data using different Java I/O streams.
4. Create graphical user interfaces and Applets by applying the knowledge of EventHandling.
5. Create robust applications using Java standard class libraries and retrieve data from a database with JDBC.

List of Experiments:

1. a) Write a Java sample program to implement class and object concepts.
   b) Write a Java program to illustrate types of constructors.
2. a) Write a Java program to illustrate the concept of Single level and Multi level Inheritance.
   b) Write a Java program to illustrate the concept of class with method overloading and method overriding
3. a) Write a Java program to demonstrate the Interfaces & Abstract Classes.
   b) Write a Java program to implement the concept of exception handling.
   c) Write a Java program to illustrate the concept of threading using Thread Classand runnable Interface.
4. Write a Java program to illustrate the concept of Thread synchronization.
5. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java.util)
6. a)Write a Java program that reads a file name from the user, and then displays inform action about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
   b) Write a Java program to illustrate the concept of I/O Streams.
7. a) Write a Java applet program to implement Color and Graphics class
   b) Write a Java program to implement AWT class
8. Write a Java applet program for handling mouse & key events
9. Write a Java applet program to implement Adapter classes
10. Write a JDBC program to implement CURD operation
11. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field todisplay the result.

Suggested Readings:

1. Herbert Schildt, Dr.Danny Coward “Java, The Complete Reference”, 13th Edition, Tata McGraw Hill,2021.
2. James M Slack, “Programming and Problem Solving with JAVA”, Thomson Learning,2023.

Course Code	Course Title						Core/Elective
U24IT403/ U24IT501/ U24IT602	AUTOMATA THEORY, LANGUAGES AND COMPUTATION						Core
Prerequisite	Hours Per Week				CIE	SEE	Credits
Discrete Mathematics	L	T	D	P			
	3	1	-	-	40	60	4

Course Objectives

Develop ability to

1. Develop a formal notation for strings, languages and machines.
2. Understand Regular Expression and algebraic laws.
3. Design context free grammars and PDA, to generate strings from a context free language and Convert them into normal forms.
4. Identify the hierarchy of formal languages, grammars and machines.
5. Distinguish between computability and non-computability and Decidability and undecidability.

Course Outcomes

At the end of the course, student would be able to

1. Gain knowledge of the various abstract machines.
2. Use regular languages and regular expression for constructing different finite state machines.
3. Understand and design different types of grammars.
4. Construct Push down Automata.
5. Construct Turing Machine.

UNIT-I

Introduction: Introduction to Finite Automata, Structural Representations, Basic Concepts of Automata Theory – Symbol, Alphabets, Strings, Languages, Kleene Closure, Positive Closure.

Finite Automata without Output: Deterministic Finite Automata, Nondeterministic Finite Automata, Finite Automata with Epsilon-Transitions, Conversion of NFA to DFA

UNIT-II

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Properties of regular sets, Conversion of Finite Automata to Regular Expressions.

Pumping Lemma for Regular Languages: Pigeon Hole Principle, Statement of the pumping lemma, Applications of the Pumping Lemma.

Closure Properties of Regular Languages: Closure properties of Regular languages, Decision Properties of Regular Languages, Minimization of Automata- Equivalence, Myhill-Nerode Theorem

UNIT-III

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Trees, Ambiguity in Grammars.

Push down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state & empty stack, Deterministic PDA Conversion of CFG to PDA, Conversion of PDA to CFG

UNIT-IV

Normal Forms for Context- Free Grammars: Grammar Simplification, Chomsky Normal form, Greibach Normal form.

Context-Free Languages: Statement of pumping lemma, Applications pumping lemma

Properties of Context-Free Languages: Closure properties of CFL's, Decision Properties of CFL's

UNIT-V

Turing Machines: Introduction, Techniques for construction of Turing machines, Modifications of TM, TM as enumerator, Restricted TM.

Undecidability: Recursive and Recursively enumerable languages, Halting problem, Rice Theorem, Post's correspondence problem, P,NP,NP-Complete, NP-Hard Problems, Chomsky's Hierarchy–Regular grammars, Unrestricted grammar, CSL

Suggested Reading:

1. Introduction to Automata Theory, languages, and Computation, John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Pearson Education India, 5th Edition,2017.
2. Automata and Computability, Undergraduate Texts in Computer Science,Dexter C. Kozen, Springer, 2015.
3. Introduction to the Theory of Computation, Michael Sipser, PWS Publishing 3rd Edition,2014.
4. Introduction to Languages and the Theory of Computation, John Martin, Tata McGraw Hill, 3rd Edition, 2002.

Course Code	Course Title						Core/Elective
U24EP4L1	Design Thinking Skills Lab (Common to all Branches)						Core
Pre-requisites	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
--	-	-	-	02	25	50	1.0

Course Objectives:

Students are able

1. Sensitize students to real-world, community-based challenges
2. Encourage empathy-driven problem identification
3. Promote creative ideation and co-creation with stakeholders
4. Guide students to develop low-cost, scalable prototypes
5. Encourage alignment with Sustainable Development Goals (SDGs)
6. Build confidence in innovation, teamwork, and social impact

Course Outcomes:

On completion of this course, the student will be able to:

1. Listen and interpret spoken language productively.
2. Understand and apply the design thinking process
3. Conduct field research and empathize with user needs
4. Define clear and relevant problem statements
5. Generate and evaluate innovative ideas collaboratively
6. Create and test functional prototypes
7. Gather feedback and refine solutions iteratively
8. Communicate project outcomes through storyboards and pitches
9. Contribute to community development through ethical, sustainable design

LIST OF ACTIVITIES

1. Community Sensitization & Field Immersion
2. Stakeholder Interviewing & Empathy Mapping
3. Defining Problem Statements Using CESE Framework
4. Brainstorming & Ideation Workshops
5. Developing Innovation Canvases
6. Co-Creation with Local Stakeholders
7. Prototype Building and Field Testing
8. Feedback Gathering and Iterative Improvement
9. Planning Implementation Strategies
10. Preparing Impact Pitch Presentations
11. Storyboarding Project Journeys
12. Jury Presentation and Public Exhibition

Suggested Readings:

1. **Brown, Tim** – *Change by Design: How Design Thinking Creates New Alternatives for Business and Society*
2. **IDEO.org** – *The Field Guide to Human-Centered Design* (Free PDF at: www.designkit.org)
3. **Rolf Faste / Stanford d.school** – *Bootcamp Bootleg* (Design Thinking tools and templates)
4. **Jeanne Liedtka & Tim Ogilvie** – *Designing for Growth: A Design Thinking Tool Kit for Managers*
5. **Anand, Natarajan** – *Social Innovation: Concepts, Cases, and Impact*
6. **Datar, Ashish, Karamchandani** – *From Blueprint to Scale: The Case for Philanthropy in Impact Investing*
7. **United Nations Development Programme (UNDP)** – *SDG Accelerator and Innovation Tools*
8. **National Innovation Foundation – India** – *Grassroots Innovations: Minds on the Margin are not Marginal Minds*