<table>
<tr><th>Course Code</th><th colspan="6">Course Title</th><th>Core / Elective</th></tr>
<tr><td>U24MA301</td><td colspan="6">MATHEMATICS – III
(PROBABILITY AND STATISTICS)

Branch – CSE-AIML ,IT, AIML , ECE & CSE</td><td>Core</td></tr>
<tr><td rowspan="2">Prerequisite</td><td colspan="4">Contact Hours Per Week</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td><td rowspan="2">Credits</td></tr>
<tr><td rowspan="2">--</td><td>L</td><td>T</td><td>D</td><td>P</td></tr>
<tr><td>3</td><td>-</td><td>-</td><td>-</td><td>40</td><td>60</td><td>3</td></tr>
</table>

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 Find the parameters and concepts of correlation, regression and obtain the knowledge of sampling Theory with context to test of hypothesis.
4. 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).

Textbooks:

1. Advanced Engineering Mathematics, R.K.Jain & Iyengar, Narosa Publications.
2. B.S.Grewal,"Higher Engineering Mathematics",Khanna Publishers,2000.
3. Engineering Mathematics,P.SivaramakrishnaDas&C.Vijaya Kumar,Pearson India Education Services Pvt.Ltd.
4. Engineering Mathematics, SS Sastry, PHI Learning, Private Limited

REFERENCE BOOKS:

1. Fundamentals of Mathematical Statistics, S.C.Gupta & V.K.Kapoor, S.Chand Publication.
2. P.G.Hoel, S.C.Portand C.J.Stone, "Introduction to Probability Theory", Universal Book Stall, 2003.
3. W.Feller,"An Introduction to Probability Theory and its Applications",Vol.1, Wiley, 1968.
4. N.P.Baliand M.Goyal,"A textbook of Engineering Mathematics",Laxmi Publications, 2010.

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

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

Stacks and Queues:

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

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

UNIT – III

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

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

UNIT – IV

Searching Techniques: Linear search and Binary Search algorithms.

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

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:

Text Books:

1. “Fundamentals of Data Structures in C”, Ellis Horowitz, Sartaj Sahani, Susan Anderson Freed, Computer Science Press, 2004
2. S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2008.

Reference Books:

3. D. Samanta, “Classic Data Structures”, PHI Learning, 2nd Edition, 2004.
4. Mark A Weiss, “Data Structures and Algorithm Analysis In C”, SecondEdition (2002), Pearson
5. “Data Structures and Algorithms in C++”, second Edition by Michael T. Goodrich and Roberto Tamassia.

Course Code	Course Title						Core/Elective
U24EC304	**Digital Electronics & Computer Organization**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**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

Number Systems: Number systems, Complements of Numbers, Codes- Weighted and Non-weighted codes and its Properties, Parity check code and Hamming code. Boolean algebra: Basic Theorems and Properties, Switching Functions- Canonical and Standard Form, Algebraic Simplification, Fundamentals of Digital Logic Gates.

UNIT – II

Combination Circuits: Implementation of logic functions using K-Map Quine- McCluskey Tabular method, Adders: half adder, full adder, Subtractors, Comparators, Multiplexers, De-multiplexers, Encoders and Decoders.

UNIT-III

Sequential circuits: Basic Architectural Distinctions between Combinational and Sequential circuits, SR Latch, Flip Flops: SR, JK, JK Master Slave, D and T Type Flip Flops. Excitation Table of all Flip Flops, Conversion from one type of Flip-Flop to another.

UNIT-IV

Basic Structure of Computers: Computer Types, Block diagram of Digital computer, Basic Operational Concepts, Bus Structures, Stored program organization and computer registers, Instruction formats, Input/output Organization: Block diagram of I/O organization, Interrupts, direct memory access. Asynchronous data transfer: strobe control and hand shaking.

UNIT -V

The Memory System: Basic concepts, Semiconductor RAM memories, Read-Only memories, Speed, Size and Cost, Primary memory ,Auxiliary memory, Associative memory, Cache memories, Virtual Memories, Memory management requirements.

Text Books:

1. Morris Mano M. and Michael D. Ciletti, "Digital Design, With an Introduction to Verilog HDL", Pearson 5 th edition, 2013.
2. 2 RP Jain " Modern Digital Electronics" , Fourth Edition Mcgraw hill education (India) Pvt Limited, 2003
3. Carl Hamacher, ZvonkoVranesic, SafwatZaky, Computer Organization, 5th Edition, McGraw Hill, 2002.
4. Suggested Reading:
5. Ronald J Tocci, Neal Widmer, Greg Moss, "Digital Systems: Principles and Applications", Pearson 11th Edition, 2011.
6. Computer Architecture a quantitive approach, Jhon L. Hennessy and David A. Patterson, Fourth Edition Elsevier.
7. Computer Architecture: Fundamentals and principles of Computer Design, Joseph D. Dumas II, BS Publication.

<table>
<tr><td>Course code</td><td colspan="6">Course title</td><td>Core/Elective</td></tr>
<tr><td>U24EN301</td><td colspan="6">ENGLISH FOR TECHNICAL COMMUNICATION</td><td>--</td></tr>
<tr><td>Pre-requisites</td><td colspan="4">Contact Hours Per Week</td><td rowspan="2">CIE</td><td rowspan="2">SEE</td><td rowspan="2">Credits</td></tr>
<tr><td rowspan="2">English for Professional Communication</td><td>L</td><td>T</td><td>D</td><td>P</td></tr>
<tr><td>2</td><td>-</td><td>-</td><td>-</td><td>40</td><td>60</td><td>2</td></tr>
<tr><td colspan="8">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</td></tr>
</table>

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, Business letters (all types), Business proposals.

UNIT-III

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

UNIT-IV

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

UNIT-V

Information Transfer and Presentations: Non-verbal (bar diagram, flow chart, pie chart,tree diagram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oraland 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
U24IT301	**Database Management Systems**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

The objectives of this course is to impart knowledge

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:

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, 6th Edition, 2017.
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamental Database Systems", Pearson Education, 6th Edition, 2014.
3. Raghu Ramakrishnan, "Database Management System", Tata McGraw-Hill Publishing Company, 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
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. Analyze and implement programming skills to implement sorting and searching algorithms
3. Apply the suitable data structures for the given real world problems.
4. Acquire knowledge in practical applications of data structures.
5. Provide solutions for various graphical concepts.

Course Outcomes:

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

1. Write programs in various data structures using arrays and linked lists.
2. Develop ADT necessary for solving problems based on Stacks and Queues
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 forsolving a given problem

List of Experiments:

1. Implementation of Stacks and Queues using Arrays.
2. Solving Towers of Hanoi problem
3. Implementation of Circular Queue.
4. Solving tic-tac -toe problem
5. Implementation of Infix to Postfix Conversion, Postfix ExpressionEvaluation.
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 and Hashing
11. Implementation of Operations on Binary Tree (Delete Entire Tree, Copy Entire Tree, Mirror Image, Level Order, Search for a Node etc.)
12. Implementation of Tree Traversals on Binary Trees.
13. Implementation of Binary Search Tree. (Insertion, Deletion and Search operations)
14. Implementation of operations on AVL Trees.
15. Implementation of Traversal on Graphs.
16. Implementation of Selection, Merge, Quick and InsertionSort.
17. Implementation of Prims and Kruskals Algorithm.

Suggested Readings:

1.S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2008.
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

U24IT3L1	Database Management Systems Lab						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	3	25	50	1.5

Course Objectives:

1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation
3. To 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

After completing this course, the student will 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. To 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. Write a PL/SQL program to swap two numbers.
2. Write a PL/SQL program to find the largest of three numbers
3. Write a PL/SQLprogramtofindthetotalandaverageof6subjectsanddisplaythegrade.
4. Write a PL/SQL program to find the sum of digits in a given number.
5. Write a PL/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. Write a PL/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, 3rd Edition, 2008.
2. Silberschatz, Korth, ―Database System Concepts‖, McGraw Hill, V edition, 2005.
3. Rick F. Vander Lans, ―Introduction to SQL‖, Pearson education, 2007.
4. B. Rosenzweig and E. Silvestrova, ―Oracle PL/SQL‖, Pearson education, 2004.
5. Dr. P. S. Deshpande, ―SQL & PL/SQL for Oracle 10g‖, Black Book, Dream Tech, 2006.
6. M. Mc Laughlin, ―Oracle Database 11g PL/SQL Programming‖, TMH, 2017

Course code	Course title						Core/Elective
U24EN3L1	**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 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
U24DT3L1	Design and Thinking Lab						**Core**
Prerequisite	Contact Hours per Week				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives:

Develop ability to

1. Understand grassroots realities and local community structures.
2. Identify real-world problems using empathy and mapping tools.
3. Design and test innovative solutions aligned with SDGs.
4. Implement, evaluate, and present a measurable impact project.
5. Reflect on ethics, sustainability, and inclusive community engagement.

Course Outcomes

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

1. Live social impact project implemented
2. Pitch-ready project for incubation or scaling
3. Opportunities for publication, patent, or grant
4. Potential nomination for DOEI's pre-incubation track

List of Experiments:

Week	**Theme**	**Activities**	**Output/Assessment**
1	Orientation & SDG Awareness	Field ethics, intro to CESE, local context sensitization	Quiz + Reflection Journal
2–3	Community Immersion	Field visit, stakeholder interviews, Participatory Rural Appraisal (PRA)	Field Report
4	Problem Mapping	Problem tree, empathy mapping, root cause analysis	Community Challenge Brief
5–6	Ideation & Co-creation	Design thinking workshop, idea sprints, stakeholder feedback	Innovation Canvas + Poster
7	Feasibility Analysis	Resource and implementation planning	Mini-pitch to mentors
8–10	Pilot Implementation	Field-based solution deployment, stakeholder engagement	Prototype/Program Implementation

Week	Theme	Activities	Output/Assessment
11	Feedback & Impact Evaluation	Feedback surveys, journaling, reflection meeting	Impact Survey + Pivot Report
12	Documentation & Showcase Preparation	Storyboard, final impact report, pitch prep	Report + Presentation Deck
13–14	CESE Expo & Final Jury Evaluation	Public expo + jury panel feedback	Final Pitch + Community Acknowledgment

.

LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY
(An Autonomous Institution)
DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
SCHEME OF INSTRUCTIONS & EXAMINATIONS [LR-24]
(W.e.f Academic Year 2025-26)

B.E.IV-Semester

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks		Duration in Hours	
								CIE	SEE		
Theory Course											
1	U24CM401	PCC	Introduction to Data Science	3	0	-	3	40	60	3	3
2	U24CM402	PCC	Artificial Intelligence #	3	0	0	3	40	60	3	3
3	U24IT402	PCC	Java Programming	3	0	0	3	40	60	3	3
4	U24IT403	PCC	Automata Theory, Languages and Computation	3	1	0	3	40	60	3	4
5	U24CD401	PCC	Operating Systems	3	0	0	3	40	60	3	3
Practical/Laboratory Course											
6	U24CD4L1	PCC	Operating Systems Lab	0	0	2	2	25	50	3	1.5
7	U24IT4L2	PCC	JAVA Programming Lab	0	0	3	3	25	50	3	1.5
8	U24CM4L2	PCC	Artificial Intelligence Lab	0	0	3	3	25	50	3	1.5
Total				**15**	**1**	**9**	**24**	**275**	**450**	**-**	**20.5**

 # - Term Work

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
MC: Mandatory Courses **BSC**: Basic Science Courses **ESC**: Engineering Science Courses
HSMC: Humanities & Social Sciences including Management Course **MA**: Mathematics
CH: Chemistry **EN:** English **ME**: Mechanical Engineering. **EE:** Electrical Engineering

Note:
1. Each contact hour is a Clock Hour.
2. 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						Course Category
U24CM401	INTRODUCTION TO DATA SCIENCE						PCC
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	40	60	3

Course Objectives:

1. An understanding of the data operations
2. An overview of simple statistical models and the basics of machine learning techniques of regression.
3. An understanding good practices of data science
4. Skills in the use of tools such as python, IDE
5. Understanding of the basics of the Supervised learning

Course Outcomes:

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

1. Describe what Data Science is and the skill sets needed to be a data scientist
2. Explain the significance of exploratory data analysis (EDA) in data science
3. Ability to learn the supervised learning, SVM
4. Apply basic machine learning algorithms (Linear Regression)
5. Explore the Networks, PageRank

UNIT-I

Introduction: Toolboxes: Python, fundamental libraries for data Scientists, Integrated development environment (IDE).

Data operations: Reading, selecting, filtering, manipulating, sorting, grouping, rearranging, ranking, and plotting.

UNIT-II

Descriptive statistics, data preparation. Exploratory Data Analysis data summarization, data distribution, measuring asymmetry. Sample and estimated mean, variance and standard score. Statistical Inference frequency approach, variability of estimates, hypothesis testing using confidence intervals, using p values.

UNIT-III

Supervised Learning: First step, learning curves, training-validation and test. Learning models generalities, support vector machines, random forest.

UNIT-IV

Regression analysis, Regression: linear regression simple linear regression, multiple & Polynomial regression, Sparse model. Unsupervised learning, clustering, similarity and distances, quality measures of clustering, case study.

UNIT-V

Network Analysis, Graphs, Social Networks, centrality, drawing centrality of Graphs, PageRank, Ego-Networks, community Detection**.**

Text Book

1. Introduction to Data Science a Python approach to concepts, Techniques and Applications, Igual, L;Seghi', S. Springer, ISBN:978-3-319-50016-4
2. Data Analysis with Python A Modern Approach, David Taieb, Packt Publishing, ISBN-9781789950069

Reference Books

1. Introduction to Data Science a Python approach to concepts, Techniques and Applications, Igual, L;Seghi', S. Springer, ISBN:978-3-319-50016-4 2. Data Analysis with Python A Modern Approach, David Taieb, Packt Publishing, ISBN-978178995

Course Code	Course Title						Core/Elective
U24CM402 **U24CM502**	**ARTIFICIAL INTELLIGENCE**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Data structures	**3**	**-**	**-**	**-**	**40**	**60**	**03**

Course Objectives

The course will introduce the students to

1. Understand the various characteristics of Intelligent agents.
2. Learn the different search strategies in AI.
3. Learn to represent knowledge in solving AI problems.
4. Understand the different ways of designing software agents.
5. Know about the various applications of AI.

Course Outcomes

After successful completion of the course the students will be able to

1. Use appropriate search algorithms for any AI problem.
2. Represent a problem using first order and predicate logic.
3. Provide the apt agent strategy to solve a given problem.
4. Design software agents to solve a problem.
5. Design applications for NLP that use Artificial Intelligence.

UNIT – I

Introduction–Definition - Future of Artificial Intelligence –Characteristics of Intelligent Agents– Typical Intelligent Agents – Problem Solving Approach to Typical AI problems.

UNIT – II

Problem Solving Methods

Problem solving Methods - Search Strategies- Uninformed - Informed - Heuristics - Local Search Algorithms and Optimization Problems - Searching with Partial Observations - Constraint Satisfaction Problems – Constraint Propagation - Backtracking Search - Game Playing - Optimal Decisions in Games – Alpha - Beta Pruning - Stochastic Games

UNIT – III

Knowledge Representation

First Order Predicate Logic – Unification – Forward Chaining-Backward Chaining – Resolution – Knowledge Representation - Ontological Engineering-Categories and Objects – Events - Mental Events and Mental Objects - Reasoning Systems for Categories - Reasoning with Default Information.

UNIT – IV

Software Agents

Architecture for Intelligent Agents – Agent communication – Negotiation and Bargaining – Argumentation among Agents – Trust and Reputation in Multi-agent systems.

UNIT – V

Applications of AI

AI applications – Language Models – Information Retrieval- Information Extraction – Natural Language Processing - Machine Translation – Speech Recognition – Robot – Hardware – Perception – Planning – Moving

Suggested Readings:

1. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach‖, Prentice Hall, Third Edition, 2009.
2. I. Bratko, ―Prolog Programming for Artificial Intelligence‖, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.
3. M. Tim Jones, ―Artificial Intelligence: A Systems Approach(Computer Science)‖, Jones and Bartlett Publishers, Inc.; First Edition, 2008
4. Nils J. Nilsson, ―The Quest for Artificial Intelligence‖, Cambridge University Press, 2009.
5. William F. Clocksin and Christopher S. Mellish,‖ Programming in Prolog: Using the ISO Standard‖, Fifth Edition, Springer, 2003.
6. Gerhard Weiss, ―Multi Agent Systems‖, Second Edition, MIT Press, 2013.
7. David L. Poole and Alan K. Mackworth, ―Artificial Intelligence: Foundations of Computational Agents‖, Cambridge University Press, 2010.

Course Code	**Course Title**						Core/Elective
U24IT402	**JAVA Programming**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			

PPS	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

Develop ability to

1. Understand 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, exception handling, multi-threading.
3. Understand fundamentals of object-oriented programming in Java.
4. Define classes, invoking methods, difference between applet and application programs, using class libraries
5. Use Collection framework, AWT and event handling to solve real world problems.

Course Outcomes

At the end of the Course, Student would be:

1. Achieve proficiency in object-oriented concepts and also learns to incorporate the same into the Java programming language.
2. Create Java application programs using sound OOP practices e.g. Inheritance, interfaces and proper program structuring by using packages, access control specifiers.
3. Understand and Implement the concepts of Exception Handling in JAVA.
4. Develop the ability to solve real-world problems through software development in high-level programming language using Large APIs of Java as well as the Java standard class library.
5. Understand File, Streams, Input and Output Handling in java.

UNIT – I

Object Oriented Programming: Benefits of Object Oriented Programming.

Introduction to Java: Java buzzwords, bytecode. Java Programming Fundamentals, data types, variables, arrays, operators, expressions, control statements, concepts of classes, objects, constructors, methods, access control, overloading methods and constructors, introducing access control, static, final, exploring string class.

Principles of OOPS: Data Abstraction, Data Encapsulation, Polymorphism, and Inheritance.

UNIT – II

Interfaces: Defining an interface, implementing interfaces, extending interface. **Packages:** Defining, Creating and Accessing a Package, importing packages

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

UNIT – III

Multithreading: Java Thread Model, 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.

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.

UNIT – IV

GUI Programming with java: The AWT class hierarchy, MVC architecture. Applet Revisited: Basics, architecture and skeleton, simple applet program.

Event Handling: Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces. Handlingmouse and keyboard events, Adapter classes.

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

UNIT – V

Exploring Swing: JLabel, Image Icon, JTextField, the Swing buttons, JTabbedpane, JScrollPane, JList,JComboBox.

Servlet: Life cycle, using tomcat, simple servlet, servlet API, javax. servlet package, reading servlet parameters,javax. servlet. http package, handling HTTP requests and responses

Text Books:

1. Herbert Scheldt, "The Complete Reference Java, 7th Edition, Tata McGraw Hill,2006.
2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning,2002.

Reference Books:

3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 5th Edition, McGraw Hill Publishing,2010.
4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education /PHI.

Course Code	**Course Title**			Core/Elective
U24IT403	**Automata Theory, Languages and Computation**			**Core**
	Contact Hours per Week			

Prerequisite	L	T	D	P	CIE	SEE	Credits
PPS	3	-	-	-	40	60	3

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:

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 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: 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, 3rd Edition,2008.
2. Automata and Computability, Undergraduate Texts in Computer Science,Dexter C. Kozen, Springer, 2007.
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						Course Category
U24CD401	Operating Systems						PCC
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
DLD	L	T	D	P			
	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: Computer System organization & Architecture, Operating System Structure & Operations, Process, Memory and Storage Managements, Protection and Security, Distributed and Special-Purpose Systems, Computing Environments.

System Structures: System calls, Types of System Calls, System Programs.

Process Concept: Overview, Process Scheduling, Operations on Processes, Inter process communication, Communication in Client/Server Systems, Operating System Examples.

UNIT-II

Process 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.

Text Book

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Principles, ninth Edition, John Wiley & Sons Publication,2012
2. A.Tanenbaum-Modern Operation Systems. Third edition, Pearson Education, 2008.

Reference Books

1. William Stallings - Operating Systems, Fifth Edition, Pearson Education, 2005.
2. Ida M.Flynn, Understanding Operating Systems, Sixth Edition, Cengage, 2011
3. Operating Systems: Principles and Practice, Thomas Anderson and Michael Dahlin, Recursive Books, 2014.
4. The Design of Unix Operating System, Maurice Bach, Prentice Hall, 1988.

Course Code	Course Title			Core/Elective
U23CD5L2	**OPERATING SYSTEMS LAB**			**Core**
	Contact Hours per Week			

Prerequisite	L	T	D	P	CIE	SEE	Credits
-	-	-	-	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

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**			
-	**-**	**-**	**-**	**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 multi-threading.
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 specifiers.
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 adatabase with JDBC.

List of Experiments:

1. (a) Write a Java program to illustrate the concept of class with method overloading
   (b) Write a Java program to illustrate the concept of class with method overriding
2. (a) 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)

   (b) Write a Java program to illustrate the concept of Single level and Multi levelInheritance.

3. (a) write a Java program to demonstrate polymorphism
   (b) Write a Java program to demonstrate the Interfaces & Abstract Classes.
4. (a) Write a Java program to implement the concept of exception handling.
   (b) Write a Java program to illustrate the concept of threading using Thread Classand runnable Interface.
5. (a) Write a Java program to illustrate the concept of Thread synchronization.
6. (a) Write a Java program that correctly implements producer consumer problemusing the concept of inter thread communication.
   (b) 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.
7. (a) Write a Java program to illustrate the concept of I/O Streams
   (b) Write a Java program to implement serialization concept

8. (a) Write a Java applet program to implement Colour and Graphics class
   (b) Write a Java applet program for handling mouse & key events
9. (a) Write a Java applet program to implement Adapter classes
   (b) 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.
10. (a) Write an example for JDBC prepared statement with ResultSet
    (b) Write a Java Program to get primary key value (auto-generated keys) frominserted queries using JDBC
11. (a) Write a Java Program to create a simple JList
    (b) Write a Java Program to create a simple checkbox using JCheckBox
12. (a) Write a Java Program to create a checkbox and Item Listener to it.
    (b) Write Servlet application for following
        i. Html & Servlet Communication
        ii. Auto refresh a page
        iii. Demonstrate session tracking
        iv. Select record from database
        v. Application for login page
        vi. Insert record into database
        vii. Count the visits on webpage
        viii. Insert teacher record in Database

Suggested Readings:

1. Herbert Scheldt, “The Complete Reference Java, 7th Edition, Tata McGraw Hill,2006.
2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning,2002.
3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 5th Edition, McGraw Hill Publishing,2010.
4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education.

Course Code	Course Title						Core/Elective
U24CM4L2 **U24CM5L2**	**ARTIFICIAL INTELLIGENCE LAB**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Python	**-**	**-**	**-**	**3**	**25**	**50**	**1.5**

Course Objectives:

The objective of this lab is to get an overview of the various Artificial Intelligence techniques and can able to demonstrate those using python.

1. To introduce students to the basic concepts of AI Searching techniques
2. To develop skills for solving practical problems.
3. To gain experience neural networks.

Course Outcomes:

After the completion of the course the student can able to:

1. After learning the AI concepts the student must be able to design and implement AI solutions searching techniques using AI.
2. Able to know about facts of querying.
3. Be capable of confidently applying tree mechanism using AI with Neural Network
4. Be capable of performing experiments in Machine Learning using real-world data. ·
5. Be capable to implement classifiers and Regression algorithm

List of Experiments:

1. Develop a Python program to perform Breadth-First Search (BFS).
2. Create a Python program that carries out Depth-First Search (DFS).
3. Build a console-based Tic-Tac-Toe game using Python.
4. Write a Python script to solve the 8-Puzzle problem.
5. Design a Python program to address the classic Water Jug puzzle.
6. Implement the Travelling Salesman Problem (TSP) using Python.
7. Construct a recursive Python solution for the Tower of Hanoi problem.
8. Simulate the Monkey and Banana problem through a Python implementation.
9. Write a Python solution for the Missionaries and Cannibals puzzle.
10. Implement the N-Queens problem using backtracking in Python.
11. Write a Python program to train and evaluate the following machine learning models using scikit-learn:
    (i) Decision Tree Classifier
    (ii) Multi-layer Perceptron (Feedforward Neural Network)
    (iii) Gaussian Naive Bayes classifier using scikit-learn (Implement any two classifiers) .
12. Create a Python program that performs Linear Regression using any two different algorithms.

Suggested Readings:

1. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach‖, Prentice Hall, Third Edition, 2009.
2. M. Tim Jones, ―Artificial Intelligence: A Systems Approach (Computer Science), Jones and Bartlett Publishers, Inc.; First Edition, 2008.
3. Elaine Rich, Kevin Knight, and Shivashankar B. Nair, *Artificial Intelligence*, McGraw-Hill Education, Third Edition, 2008.
4. Gerhard Weiss, ―Multi Agent Systems‖, Second Edition, MIT Press, 2013.
5. David L. Poole and Alan K. Mackworth, ―Artificial Intelligence: Foundations of Computational Agents‖, Cambridge University Press, 2010.