LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SCHEME OF INSTRUCTIONS AND EXAMINATIONS [LR-24]

(W.e.f. Academic Year 2026-27)

B.E.V-Semester

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks CIE	Maximum Marks SEE	Duration in Hours	
Theory Course											
1	U24MB501	HSMC	Business Economics and Financial Analysis	3	0	0	3	40	60	3	3
2	U24CM501	PCC	Design and Analysis of Algorithms	3	0	0	3	40	60	3	3
3	U24CM502	PCC	Machine Learning	3	0	0	3	40	60	3	3
4	U24CM503	PCC	Software Engineering	3	0	0	3	40	60	3	3
5	-	PEC	Professional Elective - I	3	0	0	3	40	60	3	3
Practical/Laboratory Course											
6	U24CM5L1	PCC	Machine Learning Lab	0	0	3	3	25	50	3	1.5
7	U24CM5L2	PCC	Design and Analysis of Algorithms Lab	0	0	3	3	25	50	3	1.5
Internship											
8	U24CM5P1	PROJ	Internship /MOOCs	-	-	2	2	50	-	-	1
Skill Development Course											
9	U24MA5L1	BSC	Aptitude and Reasoning	-	-	2	2	25	50	3	1
Total				**15**	**0**	**10**	**25**	**325**	**450**	**-**	**20**

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 **CM:** COMPUTER SCIENCE & ENGINEERING (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

PCC: Program core course **PEC:** Professional Elective Course **PROJ**: Project

MB: Master of Business Administration **BSC**: Basic Science Courses **AM:**AI&ML

HSMC: Humanities & Social Sciences including Management Course

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

LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SCHEME OF INSTRUCTIONS & EXAMINATIONS [LR-24]

PROFESSIONAL ELECTIVE WITH 4 THREADS

S No.	PE1	PE2	PE3	PE4	PE5
1	Data Visualization	Artificial Neural Networks	Computer Vision	Generative AI	Cognitive Computing
2	Object Oriented Analysis and Design	Software Testing Methodologies	Software Project Management	Software quality Assurance	Agile Methodologies
3	Compiler Design	Cloud Computing	Block chain Technology	Information Retrieval Systems	Digital Forensic
4	Computer Graphics	Human Computer Interactions	Multimedia Systems	Virtual Reality	Augmented Reality

LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SCHEME OF INSTRUCTIONS & EXAMINATIONS [LR-24]

PROFESSIONAL ELECTIVE COURSES

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks CIE	Maximum Marks SEE	Duration in Hours	
Theory Course											
1	U24CM503	PEC1	Object Oriented Analysis and Design	3	0	0	3	40	60	3	3
	U24CM504		Data Visualization	3	0	0	3	40	60	3	3
	U24CM505		Compiler Design	3	0	0	3	40	60	3	3
	U24CM506		Computer Graphics	3	0	0	3	40	60	3	3
2	U24CM603	PEC2	Artificial Neural Networks	3	0	0	3	40	60	3	3
	U24CM604		Software Testing Methodologies	3	0	0	3	40	60	3	3
	U24CM605		Cloud Computing	3	0	0	3	40	60	3	3
	U24CM606		Human Computer Interactions	3	0	0	3	40	60	3	3
3	U24CM703	PEC3	Computer Vision	3	0	0	3	40	60	3	3
	U24CM704		Software Project Management	3	0	0	3	40	60	3	3
	U24CM705		Block chain Technology	3	0	0	3	40	60	3	3
	U24CM706		Multimedia Systems	3	0	0	3	40	60	3	3
4	U24CM801	PEC4	Generative AI	3	0	0	3	40	60	3	3
	U24CM802		Software quality Assurance	3	0	0	3	40	60	3	3
	U24CM803		Information Retrieval Systems	3	0	0	3	40	60	3	3
	U24CM804		Virtual Reality	3	0	0	3	40	60	3	3
5	U24CM805	PEC5	Cognitive Computing	3	0	0	3	40	60	3	3
	U24CM806		Agile Methodologies	3	0	0	3	40	60	3	3
	U24CM807		Digital Forensic	3	0	0	3	40	60	3	3
	U24CM808		Augmented Reality	3	0	0	3	40	60	3	3

Course Code	Course Title						Core/Elective
U24MB501	**BUSINESS ECONOMICS AND FINANCIAL ANALYSIS**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Basic Economics	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course will enable students to

1. Understand the concepts of Business and Economics
2. Comprehend the concepts of market dynamics namely elasticity of demand and pricing in different market structures.
3. Gain the knowledge on the production theories and cost analysis while dealing with the production and the concept of breakeven analysis.
4. To acquire the in-depth knowledge on Financial Accounting concepts and principles and preparation of final accounts.
5. To understand the financial statements through ration analysis and cash flow techniques.

Course Outcomes:

On completion of this course, the students are able to

1. Apply the concepts of business and economics during his/her professional and personal life.
2. Understand the elasticity of the demand of the product, different types, and measurement of elasticity of demand and factors influencing on elasticity of demand.
3. Recognize the Production function, features of Iso-Quants and Iso-Costs, different types of internal economies, external economies and law of returns with appropriate examples.
4. Prepare the financial statements of the firm.
5. Analyze the financial statements using ratio analysis and cash flow techniques.

UNIT – I

Introduction to Business and Economics: Business: Structure of Business Firm, Theory of Firm, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company, Non-Conventional Sources of Finance. Economics: Significance of Economics, Micro and Macro Economic Concepts, Concepts and Importance of National Income, Inflation, Money Supply in Inflation, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

UNIT – II

Demand and Supply Analysis: Elasticity of Demand: Elasticity, Types of Elasticity, Law of Demand, Measurement and Significance of Elasticity of Demand, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making, Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting. Supply Analysis: Determinants of Supply, Supply Function & Law of Supply.

UNIT- III

Production, Cost, and Market Structures & Pricing: Production Analysis: Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production Functions. Cost analysis: Types of Costs, Short run and Long run Cost Functions. Market Structures: Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, and Monopolistic Competition. Pricing: Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis, & Cost Volume Profit Analysis.

UNIT – IV

Financial Accounting: Accounting concepts and Conventions, Accounting Equation, Double Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, and Preparation of Final Accounts.

UNIT - V

Financial Analysis through Ratios: Concept of Ratio Analysis, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems), Introduction to Cash Flow Analysis (simple problems).

Suggested Reading:

1. "Managerial Economics and Financial Analysis",A.R. Aryasri, , TMH Publications, 3rd Edition, 2007.
2. "Managerial Economics", D.N. Dwivedi, Vikas Publication House Pvt. Ltd, 2nd Edition, 2012.
3. "Financial Accounting", S.N. Maheshwari & S.K.Maheshwari, , Vikas Publication House Pvt.Ltd, 4th Edition, 2012.
4. "Financial Accounting- A managerial Perspective", R. Narayana Swamy,Pearson publications, 1st Indian Reprint Edition, 2012.
5. "Managerial Economics & Financial Analysis", J.V. Prabhakar Rao & P.V. Rao, Maruthi Publishers, 1st Revised Edition, 2011.
6. "Managerial Economics and Financial Analysis", M.Kasi Reddy & Saraswathi, PHI Publications, New Delhi, 10th Revised Edition, 2012.

Course Code	Course Title						Core/Elective
U24CM501	**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:

This course will enable students to

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:

On completion of this course, the students are 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

Introduction: Algorithm, Fundamentals of algorithmic problem solving,Fundamentals of the analysis of algorithm efficiency, Asymptotic Notations and basic efficiency classes, Mathematical Analysis of Non-Recursive and Recursive Algorithms, The substitute method, Recursion tree method, Master method.

UNIT - II

Divide and conquer: Divide and Conquer: General Method, Binary Search, finding minimum and maximum MergeSort analysis, Quick Sort analysis, Strassen's matrix multiplication.

Transfer and conquer: Introduction, Balanced search trees, Heap and Heap sort.

UNIT - III

The greedy method: The General Method, Knapsack problem, Job Sequencing with Deadlines, Minimum-Cost Spanning Trees: Prim's Algorithm, Kruskal's Algorithm, Single Source Shortest Paths.

Dynamic programming: The General Method, multistage graph, Warshall's Algorithm, Floyd'sAlgorithm for the All-Pairs Shortest Paths Problem, single source shortest path, The Travelling Salesperson problem, optimal binary search,0/1 knapsack.

UNIT – IV

Pattern Matching: The naïve string-matching algorithm, Brute Force String Matching, KMP algorithm.

Np-Hard and Np-Complete Problems: Basic concepts: non-deterministic algorithms, the classes NP - Hard and NP, NP Hard problems, clique decision problem, chromatic number decision problem, Cook's theorem.

UNIT V

Backtracking: n - Queens Problem, Subset – Sum of Subsets Problem, graph coloring.

Branch and bound: Assignment problem, Knapsack problem, 15 puzzle problem, travelling salesman problem

Tries: Standard Tries, Compressed Tries, Suffix Tries, Search Engine Indexing.

Suggested Readings:

1. "Introduction to The Design and Analysis of Algorithms", Anany Levitin (Chapters 1-5,7,9,11), Pearson Education, Delhi, 2nd Edition, 2007, ISBN: 9780321358288.
2. "Fundamentals of Computer Algorithms", Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran: (Chapters 1,3-8,10-12), Universities Press, Hyderabad, 2nd Edition, 2007, ISBN: 10: 8173716129.
3. "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein: PHI, London, England, 3rd Edition, 2010, ISBN: 9780262033848.
4. "Introduction to the Design and Analysis of Algorithms A Strategic Approach", R.C.T. Lee, S.S. Tseng, R.C. Chang and Y.T. Tsai: McGraw-Hill Higher Education, USA, International Edition, 2005, ISBN-13: 978-0071243469.

Course Code	Course Title						Core/Elective
U24CM502	**MACHINE LEARNING**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Artificial Intelligence	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course will enable students to

1. To introduce students to the basic concepts of Data Science and techniques of Machine Learning.
2. To develop skills of using recent machine learning software for solving practical problems.
3. To gain experience of doing independent study and research.
4. To develop an understanding of the role of machine learning in massive scale automation.
5. To design and implement various machine learning algorithms in the range of real world applications

Course Outcomes:

On completion of this course, the students are able to :

1. Design and implement machine learning solutions of classification, regression problems.
2. Evaluate and interpret the results of the machine learning algorithms.
3. Evaluate exploratory data analysis and Data preparation and preprocessing on different datasets.
4. Calculate Statistical measurements of the given data.
5. Analyze and identify the best algorithm matches for a given dataset.

UNIT – I

Introduction: What is Machine Learning, Use Machine Learning, and Types of Machine Learning Systems: supervised, unsupervised, semi-supervised, Reinforcement Learning, Batch and Online Learning, Main Challenges of Machine Learning.

UNIT – II

Descriptive Statistics: Data representation, types of data- nominal, ordinal, interval and continuous, central tendency- calculating mean mode median, mean vs median, variability, variance, standard deviation, Mean Absolute Deviation using sample dataset, finding the percentile, interquartile range, Box Plot, Outlier, whisker, calculating correlation, covariance, causation. Exploratory data analysis, Data preparation and preprocessing, Data visualization.

UNIT – III

Regression: Introduction to Regression analysis, measure of linear relationship, Regression with stats models, Determining coefficient, meaning and significance of coefficients, coefficient calculation with least square method, Types of regression, Simple Linear Regression, Using Multiple features, Polynomial Regression, Metrics for Regression: MSE, RMSE, MAE.

UNIT – IV

Classification: Classification problem, Probability based approach, Logistic Regression- log-odd, sigmoid transformation, Metrics: Confusion Matrix, Accuracy, Error Rate, Precision, Recall, ROC curve, F1 score, and introduction to gradient descent.

UNIT – V

Non Parametric & SVM classification: About Non parametric classification, Decision Trees: Entropy, Gain ratio, Information Gain, Splitting criteria.

Ensemble Method: Introduction to Random Forest, Accuracy measure & performance

Instance based learning- Introduction, KNN algorithm, Distance measures, model building, locally weighted regression, radial basis functions, SVM classifier, hyper-plane, slack variables, geometric transformation kernel trick, kernel transformation.

Suggested Readings:

1. Machine Learning in Action, Peter Harrington, Dreamtech Press India Pvt. Ltd, 1st Edition,2012.
2. The Field Guide to Data Science ,Booz, Allen, Hamilton, Manning Publications 2nd Edition,2018.
3. Hands-On Machine Learning with Scikit-Learn and TensorFlow Aurélien Géron, ,O'Reilly Media, 3rd Edition,2017.

Course Code	Course Title						Core/Elective
U24CM503	**SOFTWARE ENGINEERING**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
PPS	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course will enable students

1. Introduce the basic concepts of software development processes from defining a product to shipping and maintaining
2. Impart knowledge on various phases, methodologies and practices of software development
3. Understand the importance of testing in software development
4. Study various testing strategies along with its relationship with software quality and metrics

Course Outcomes:

On completion of this course, the students are able to :

1. Acquire working knowledge of alternative approaches and techniques for each phase of software development
2. Judge an appropriate process model(s) assessing software project attributes and analyze necessary requirements for project development eventually composing SRS
3. Creation of visual models to describe (non-) algorithmic solutions for projects using various design principles.
4. Acquire skills necessary as an independent or as part of a team for architecting a complete software project by identifying solutions for recurring problems exerting knowledge on patterns.
5. Concede product quality through testing techniques employing appropriate metrics by understanding the practical challenges associated with the development of a significant software system

UNIT-I

Introduction to Software Engineering: A generic view of Process: Software Engineering, Process Framework, CMM Process Patterns, Process Assessment. Process Models: Prescriptive Models, Waterfall Model, Incremental Process Models, Evolutionary Process Models, Specialized Process Models, The Unified Models, Personal and Team Process Models, Process Technology, Product and Process. An Agile view of Process: Introduction to Agility and Agile Process, Agile Process Models.

UNIT-II

Software Engineering Principles: SE Principles, Communication Principles, Planning Principles, Modeling Principles, Construction Principles, Deployment. System Engineering: Computer-based Systems, The System Engineering Hierarchy, Business Process Engineering, Product Engineering, System Modeling. Requirements Engineering: A Bridge to Design and Construction, Requirements Engineering Tasks, Initiating Requirements Engineering Process, Eliciting Requirements, Developing Use-Cases, Building the Analysis Model, Negotiating Requirements, Validating Requirements.

UNIT-III

Building the Analysis Model: Requirements Analysis Modeling Approaches, Data Modeling Concepts, Object-Oriented Analysis, Scenario-based Modeling, Flow-oriented Modeling, Class-based Modeling, Creating a Behavioral Model. Design Engineering: Design within the context of SE, Design Process and Design Quality, Design Concepts, The Design Model, Pattern-based Software Design.

UNIT-IV

Creating an Architectural Design: Software Architecture, Data Design, Architectural Styles and Patterns, Architectural Design. Modeling Component-Level Design: Definition of Component, Designing Class-based Components, Conducting Component-level Design, Object Constraint Language, Designing Conventional Components. Performing User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation.

UNIT-V

Testing: Strategies: A Strategic Approach to Conventional Software Testing, Test Strategies for O-O Software. Tactics: Software Testing Fundamentals, Black-box and White-box Testing, Basis Path Testing, Control Structure Testing, O-O Testing Methods. Debugging: Debugging Techniques, The Art of Debugging. Product Metrics: A Framework Debugging: Debugging Techniques, The Art of Debugging. Product Metrics: A Framework for Product Metrics, Metrics for each phase of software development.

Software Quality: Definition, Quality Assurance: Basic Elements, Formal Approaches, Statistical Software Quality Assurance, Software Reliability, ISO9000 Quality Standards, SQA Plan.

Suggested Readings:

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, 7th Edition, McGraw Hill, 2009

2. Ali Behforooz and Frederick J. Hudson, Software Engineering Fundamentals, Oxford University Press, 1996

3. PankajJalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House, 2008

Course Code	Course Title						Core/Elective
U24CM504	**DATA VISUALIZATION**						**PE-I**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Introduction to Data Science	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students to:

1. Learn different statistical methods for Data Visualization.
2. Understand the basics of R and Python.
3. Learn usage of Watson studio.
4. Understand the usage of the packages like Numpy, pandas and matplotlib.
5. Know the functionalities and usages of Seaborn.

Course Outcomes

After completing this course, the student will be able to

1. Differentiate various statistical methods for data Visualization..
2. Analyze the basics of R and Python Languages
3. Familiarize with Watson studio
4. Understand the usage of the packages like Numpy, pandas and matplotlib.
5. Use the functionalities and applications of Seaborn

UNIT I

Introduction to Statistics : Introduction to Statistics, Difference between inferential statistics and Descriptive statistics, Inferential Statistics- Drawing Inferences from Data, Random Variables, Normal Probability Distribution, Sampling, Sample Statistics and Sampling Distributions. R overview and Installation- Overview and About R, R and R studio Installation, Descriptive Data analysis using R, Description of basic functions used to describe data in R.

UNIT II

Data manipulation with R: Data manipulation packages-dplyr,data.table, reshape2, tidyr, Lubridate, Data visualization with R. Data visualization in Watson Studio: Adding data to datarefinery, Visualization of Data on Watson Studio**.**

UNIT III

Introduction to Jupyter Notebook, Python Scripting basics, Numpy and Pandas-Creating and Accessing Numpy Arrays, Introduction to pandas, read and write csv, Descriptive statistics using pandas, Working with text data and datetime columns, Indexing and selecting data, groupby, Merge / Joindatasets

UNIT IV

Data Visualization Tools in Python- Introduction to Matplotlib, Basic plots using matplotlib, Specialized Visualization Tools usingMatplotlib, Advanced Visualization Tools using Matplotlib Waffle Charts, Word Clouds.

UNIT V

Introduction to Seaborn: Seaborn functionalities and usage, Spatial Visualizations and Analysis in Python with Folium, Case Study.

Suggested Readings:

1. Core Python Programming - Second Edition,R. Nageswara Rao, DreamtechPress.
2. Hands on programming with R by Garrett Grolemund,Shroff/O'Reilly; Firstedition
3. Fundamentals of Mathematical Statistics by S.C. Gupta, Sultan Chand &Son9

Course Code	Course Title						Core/Elective
U24CM505	**OBJECT ORIENTED ANALYSIS AND DESIGN**						**PE-I**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Introduction to Data Science	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students to:

1. To understand the object oriented life cycle.
2. To know how to identify objects, relationships, services and attributes through UML.
3. To understand the use-case diagrams.
4. To know the Object oriented Design process.
5. To know about software quality and usability

Course Outcomes

After completing this course, the student will be able to

1. Apply fundamental concepts of object-oriented analysis and design approach.
2. Understand Unified Modeling Language Notation.
3. Apply models for object-oriented system development
4. Identify system development design patterns.
5. Create use case diagram to represent the scope of development problem domain.

UNIT-I

Introduction to UML, Importance of Modeling, Principles of Modeling, Object oriented modeling, Conceptual model of the UML, Architecture of UML, Software Development Life Cycle.

UNIT-II

Basic Structural Modeling, Classes, Relationships, Common Mechanisms, Basic Diagrams, Advanced Structural Modeling, Advanced Classes, Advanced Relationships, Interfaces, Types and Roles, Packages.

Class and Object Diagrams, Terms, Concepts, Modeling Techniques for Class Diagrams

UNIT-III

Basic Behavioral Modeling-I, Interactions, Interaction Diagrams.

Basic behavioral Modeling-II, Usecases, Use case Diagrams, Activity Diagrams.

UNIT-IV

Advanced Behavioral Modeling, Events and Signals, State Machines, Processes and Threads, Time and Space, State Chart Diagrams.

Architectural Modeling, Component, Deployment, Component Diagrams, Deployment Diagram.

UNIT V

Case Study, The Unified Library application.

REFERENCES :

1. Grady Booch, James Rumbaugh and Ivar Jacobson, “The Unified Modeling Languages User Guide”, Addison Wesley, 2004.
2. Ali Bahrami, “Object Oriented Systems Development”, Tata McGraw Hill, New Delhi.
3. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.

Course Code	Course Title						Core/Elective
U24CM506	**COMPILER DESIGN**						**PE-I**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Automata Theory, Language & Computation	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students to:

1. To understand and list the different stages in the process of compilation.
2. Identify different methods of lexical analysis.
3. Design top-down and bottom-up parsers.
4. Identify synthesized and inherited attributes.
5. Develop syntax directed translation schemes.

Course Outcomes

After completing this course, the student will be able to

1. For a given grammar specification, develop the lexical analyzer.
2. For a given parser specification, design top-down and bottom-up parsers.
3. Develop syntax directed translation schemes.
4. Develop algorithms to generate code for target machine.
5. Develop algorithms to generate code for a target machine

UNIT - I

Introduction: The Structure of a Compiler, Phases of Compilation, The Translation Process, Major Data Structures in a Compiler, Bootstrapping and Porting.

Lexical Analysis (Scanner): The Role of the Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, The Lexical Analyzer Generator Lex.

UNIT – II

SyntaxAnalysis

The Role of the Parser, Syntax Error Handling and Recovery, Top-Down Parsing: Backtracking , Recursive-descent Parsing, Predictive Parsers,LL(1)grammars, Bottom-Up Parsing: Simple LR Parsing, More Powerful LR Parsing, Using Ambiguous Grammars, Parser Generator YACC.

UNIT - III

Syntax-Directed Translation: Syntax-Directed Definitions, S-attributed and L- attributed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation.

Symbol Table: Structure, Operations, Implementation and Management.

UNIT - IV

Intermediate Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Back patching, Switch-statements, Intermediate Code for Procedures.

Run-time environment: Storage Organization, Stack Allocation of Space, Access to Non-local Data on the Stack, Parameter passing, Heap Management and Garbage Collection

UNIT – V

Code Generation: Issues in the Design of a Code Generator, The Target Language, addresses in the Target Code, Basic Blocks and Flow graphs, Optimization of Basic Blocks, Peephole Optimization, Register Allocation and Assignment.

Machine-Independent Optimizations: The Principal Sources of Optimizations, Introduction to Data-Flow Analysis.

Suggested Readings:

1. Compilers Principles, Techniques andTools",AlfredV.Aho,RaviSethi,JeffreyD,Ullman,PearsonEducation,2nd Edition, 2006.
2. Compiler Construction–Principles and Practice",Kenneth C.Louden,Thomson,PWSPublishing,1st Edition,1997.
3. "Modern Compiler Implementation C ,Andrew W.Appel,CambridgeUniversityPress,RevisedEdition,2004.

Course Code	Course Title						Core/Elective
U24CM507	**COMPUTER GRAPHICS**						**PE-1**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Automata Theory, Language and Computation	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students :

1. Understand the fundamental concepts and applications of computer graphics.
2. Learn two-dimensional and three-dimensional transformations and viewing techniques.
3. Study algorithms for drawing, clipping, and hidden surface removal.
4. Understand curve and surface modeling along with illumination and shading techniques.
5. Explore advanced topics such as animation, rendering, and color models.

Course Outcomes

After completing this course, the student will be able to

1. Explain basic concepts, devices, and algorithms used in computer graphics.
2. Apply 2D and 3D transformations and viewing techniques for graphical objects.
3. Implement clipping and hidden surface removal algorithms.
4. Analyze curve, surface representations, and illumination models.
5. Evaluate advanced graphics techniques such as animation, rendering, and color modeling.

UNIT – I: Introduction and Output Primitives

Introduction to computer graphics, applications of computer graphics, raster scan and random scan displays, input and output devices, display devices, frame buffer, points, lines, line drawing algorithms including DDA and Bresenham's algorithm, circle drawing algorithms, ellipse generation, pixel addressing and resolution.

UNIT – II: Two-Dimensional Transformations and Viewing

Two-dimensional transformations, translation, rotation, scaling, reflection, shearing, homogeneous coordinate representation, composite transformations, windowing and viewport transformation, clipping algorithms including point clipping, line clipping using Cohen–Sutherland and Liang–Barsky methods, polygon clipping.

UNIT – III: Three-Dimensional Transformations and Projections

Three-dimensional transformations, translation, rotation, scaling, composite transformations, 3D viewing pipeline, projections, parallel and perspective projections, orthographic projection, viewing transformation, hidden surface removal techniques including back-face detection, depth buffer algorithm, scan-line algorithm.

UNIT – IV: Curves, Surfaces and Illumination Models

Representation of curves and surfaces, Bezier curves, B-spline curves, parametric curves, surface representation, polygon meshes, illumination models, ambient, diffuse and specular reflection, Phong illumination model, shading techniques including flat shading, Gouraud shading and Phong shading.

UNIT – V: Advanced Topics in Computer Graphics

Color models, RGB, CMY, HSV models, animation techniques, keyframe animation, interpolation, basic principles of multimedia and rendering, texture mapping, introduction to ray tracing, fractals, introduction to virtual reality and applications of computer graphics.

Suggested Readings:

1. **Interactive Computer Graphics**, Edward Angel and Dave Shreiner, Addison-Wesley.
2. **Fundamentals of Computer Graphics**, Peter Shirley et al., A K Peters.
3. **Procedural Elements for Computer Graphics**, David F. Rogers, McGraw-Hill.

Course Code	Course Title						Core/Elective
U24CM5L1	**MACHINE LEARNING LAB**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Python Programming	**-**	**-**	**3**	**3**	**25**	**50**	**1.5**

Course Objectives:

This course enable students to:

1. To introduce students to the basic concepts of Data Science and techniques of Machine Learning.
2. To develop skills of using recent machine learning software for solving practical problems.
3. To gain experience of doing independent study and research.
4. Be capable of confidently applying common Machine Learning algorithms in practice and implementing their own
5. Be capable of performing experiments in Machine Learning using real-world data.

Course Outcomes:

On completion of this course, the students are able to:

1. The student must be able to design and implement machine learning solutions to classification, regression problems.
2. Understand complexity of Machine Learning algorithms and their limitations
3. Able to evaluate and interpret the results of the algorithms.
4. Implement Decision trees and various algorithms
5. Implement and Analyze various random forest techniques.

1. Implement a program to demonstrate the following

a) Operation of data types in Python.
b) Different Arithmetic Operations on numbers in Python.
c) Create, concatenate and print a string and access substring from a given string.
d) Append, and remove lists in python.
e) Demonstrate working with tuples in python.
f) Demonstrate working with dictionaries in python.

2. Using python write a NumPy program to compute the

a) Expected Value
b) Mean
c) Standard deviation
d) Variance
e) Covariance
f) Covariance Matrix of two given arrays.

3. For a given set of training data examples stored in a .CSV file, demonstrate Data Preprocessing in Machine learning with the following steps

a) Getting the dataset.
b) Importing libraries.
c) Importing datasets.
d) Finding Missing Data.
e) Encoding Categorical Data.
f) Splitting dataset into training and test set.
g) Feature scaling.

4. Build a linear regression model using python for a particular data set by

a) Splitting Training data and Test data.
b) Evaluate the model (intercept and slope).
c) Visualize the training set and testing set
d) predicting the test set result
e) compare actual output values with predicted values

5. The dataset contains information of users from a company's database. It contains information about UserID, Gender, Age, EstimatedSalary, and Purchased. Use this dataset for predicting that a user will purchase the company's newly launched product or not by Logistic Regression model.

User ID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0
15728773	Male	27	58000	0
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	0
15727311	Female	35	65000	0
15570769	Female	26	80000	0
15606274	Female	26	52000	0
15746139	Male	20	86000	0
15704987	Male	32	18000	0
15628972	Male	18	82000	0
15697686	Male	29	80000	0
15733883	Male	47	25000	1
15617482	Male	45	26000	1
15704583	Male	46	28000	1

6. Implement a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

7. Implement k-nearest neighbor's classification to classify the iris data set using python.

8. Given the following data, which specify classifications for nine combinations of VAR1 and VAR2 predict a classification for a case where VAR1=0.906 and VAR2=0.606, using the result of k-means clustering with 3 means (i.e., 3 centroids)

VAR1	**VAR2**	**CLASS**
1.713	1.586	0
0.180	1.786	1
0.353	1.240	1
0.940	1.566	0
1.486	0.759	1
1.266	1.106	0
1.540	0.419	1
0.459	1.799	1
0.773	0.186	1

9. Evaluate the metrics for all types of machine learning algorithms using sample data.

10. Implement an algorithm to demonstrate the significance of SVM.

Suggested Readings:

1. The Field Guide to Data Science ,Booz, Allen, Hamilton,Manning Publications 2nd Edition,2018.
2. Hands-On Machine Learning with Scikit-Learn and TensorFlow,Aurélien Géron, O'Reilly Media, 1st Edition,2017.
3. Machine Learning in Action, Peter Harrington, Manning Publications.2012.

Course Code	Course Title						Core/Elective
U24CM5L2	**DESIGN AND ANALYSIS OF ALGORITHMS LAB**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Data Structures	**-**	**-**	**3**	**3**	**25**	**50**	**1.5**

Course Objectives:

This course enable students to:

1. Understand problems by applying appropriate algorithms.
2. Analyze the efficiency of various algorithms.
3. Apply techniques of stacks and queues to solve problems.
4. Solve a program in many ways using different techniques.
5. Identify and evaluate complex problems using principles of mathematics and engineering science.

Course Outcomes

On completion of this course, the students are able to:

1. Solve problems by applying appropriate algorithms.
2. Analyze the efficiency of various algorithms.
3. Apply techniques of stacks and queues to solve problems.
4. Develop a program that can be solved in many ways using different techniques.
5. Identify and evaluate complex problems using principles of mathematics and engineering science.

Design, develop and implement the specified algorithms for the following problems using C/Java/python Language

1. Implement a program to sort the elements by using quick sort method.
2. Implement a program to sort the elements by using merge sort method.
3. Obtain the Topological ordering of vertices in a given digraph.
4. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
5. Implement 0/1 Knapsack problem using Dynamic Programming.
6. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.
7. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
8. a. Compute the transitive closure of a given directed graph using Warshall's algorithm.
   b. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm.
9. a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
   b. Check whether a given graph is connected or not using DFS method.
10. Implement N Queen's problem using Back Tracking.

Suggested Readings:

1. "Introduction to the Design and Analysis of Algorithms", Anany Levitin, Pearson Education, Delhi, 2nd Edition, 2007.
2. "Fundamentals of Computer Algorithms", Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran Universities Press, Hyderabad, 2nd Edition, 2007.

Course Code	Course Title						Core/Elective
U24CM5P1	**INTERNSHIP/ MOOCs**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**50**	**-**	**1**

Course Objectives:

This course enable students to:

1. Produce an accurate record of work performed during the Internship/Co-op
2. Apply engineering knowledge to a problem in industry
3. Produce a technical report
4. Discuss work in a team environment, if relevant to the project
5. Conduct herself/himself responsibly, safely, and ethically in a professional environment

Course Outcomes:

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

1. Design/develop a small and simple product in hardware or software.
2. Complete the task or realize a pre specified target, with limited scope, rather than taking up a complex task and leave it.
3. Learn to find alternate viable solutions for a given problem and evaluate these alternatives with reference to pre specified criteria.
4. Implement the selected solution and document the same.

Guidelines:

Internship is introduced as part of the curricula of encouraging students to work on problems of interest to industries. A batch of three students will be attached to a person from the Government or Private Organizations /Computer Industry/Software Companies/R&D Organization for a period of 4 weeks. This will be during the summer vacation following the completion of the III-year Course. One faculty coordinator will also be attached to the group of 3 students to monitor the progress and to interact with the industry co- ordinate (person from industry). The course schedule will depend on the specific internship/training experience. The typical time per topic will vary depending on the internship

1. Overview of company/project
2. Safety training
3. Discussions with project teams
4. Background research, review of documents, white papers, and scientific papers
5. Planning, designing, and reviewing the planned work
6. Executing the plans
7. Documenting progress, experiments, and other technical documentation
8. Further team discussions to discuss results
9. Final report writing and presentation

After the completion of the project, each student will be required to:

1. Submit a brief technical report on the project executed and
2. Present the work through a seminar talk (to be organized by the Department)

Note: Students have to undergo summer internship of 4 weeks at the end of semester IV and credits will be awarded after evaluation in V semester.

Course Code	Course Title						Core/Elective
U24MA5L1	**APTITUDE AND REASONING**						**--**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**2**	**2**	**25**	**50**	**1**

Course Objectives

This course enable students to:

1. Acquire the concepts of mathematical aptitude and reasoning.
2. Develop the innovative and creative thinking through basic mathematical concepts.
3. Improve analytical and problems solving skills.
4. Enhance logical thinking and mathematical ability.
5. Empower with the basic tools of mathematical aptitude.

Course Outcomes

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

1. Build proficiency in quantitative reasoning
2. Improve critical thinking skills
3. Enhance analytical skills
4. Demonstrate quantitative aptitude concepts
5. Adapt principles of quantitative aptitude to achieve qualitative results.

Guidelines

1) **Basic concepts**: combined mean, average principles, wrong values taken, number added or deleted, average speed.

2) **Percentages** -Basic Concepts, conversions, finding percentages from given numbers, quantity increases or decreases by given percentage, population increase by given percentage, comparisons, consumption when a commodity price increase or decrease and applications.

3) **Data Interpretation** - Introduction to Data Interpretation, quantitative and qualitative data, Tabular Data, Line Graphs, Bar Chart, Pie Charts, X-Y Charts. Number Series, Letter Series, Series completion and correction, Coding and Decoding. Word analogy-Applied analogy, Classifications, verbal classification.

5) **Reasoning Logical Diagrams** - Simple diagrammatic relationship, Multi diagrammatic relationship, Venn-diagrams, Analytical reasoning.
Reasoning Ability - Blood Relations, Seating arrangements, Directions, Decision making. Number Systems: Basic Concepts,

6) **Number Systems**: Natural numbers, whole numbers, integers, fractions, Rational Numbers, Irrational Numbers, Real Numbers, Divisibility Rules, Logic Equations, Remainder theorem, Unit digit calculation.
Progressions & Inequalities: Basic Concepts, Types: arithmetic, geometric, harmonic progression and applications.

7) **Profit and Loss**: Basic Concepts, discounts, marked price and list price, dishonest shopkeeper with manipulated weights, successive discounts etc.
Interest (Simple and Compound): Basic Concepts, Yearly, Half-yearly, and quarterly calculations, multiples, differences between simple and compound interest.

8) **Ratio and Proportion**: Basic Concepts of ratio and proportion, continued or equal proportions, mean proportions, invest proportion, alternative proportion, division proportion, compound proportion, duplication of ratio, finding values, coins and currencies, etc.

9) **Speed, Time and Distance**: Basic Concepts, Single train problems, two train problems: some point same side, some point opposite sides, relative speed, different points meeting at common points, different points same side (different timings vs. same timings), ratios, number of stoppages, average speed, etc.

10) **Time and Work**: Basic Concepts, comparative work, mixed work, alternative work, middle leave and middle join, ratio efficiency.

11) **Permutations and combinations**: Basic Concepts, differences between permutations and combinations, alternative arrangement, fixed positions, items drawing from a single group, items drawing from a multiple group, total ways of arrangement with repetitions and without repetitions, handshakes or line joining between two points or number of matches, sides and diagonals, etc.

12) **Clocks and Calendars**: Basic Concepts, Angle between minute hand and hour hand, reflex angle, hours hand angle, time gap between minute hand and hour hand, relative time: coincide, opposite sides and right angle, mirror images, faulty clock (slow/fast), miscellaneous, calendar.

13) **Geometry and Mensuration**: Basic concepts, types of angles.

Plane figures: rectangles, squares, triangles, quadrilateral, areas, perimeters, etc.

Solid figures: cubes, cuboids, cylinders-area (total surface area and lateral surface area), volumes, perimeters.

Others: Parallelogram, Rhombus, Trapezium, Circle, Sector, Segment, Cone, Sphere, Hemisphere, etc.

Suggested Readings:

1. Aptitude and reasoning skills lab Manual, LIET, HYD.

LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SCHEME OF INSTRUCTIONS & EXAMINATIONS [LR-24]

(W.e.f. Academic Year 2026-27)

B.E.VI-Semester

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P / D	Contact Hours/Week	Maximum Marks CIE	Maximum Marks SEE	Duration in Hours	
Theory Course											
1	U24CM601	PCC	Web Design and Development	2	1	0	3	40	60	3	3
2	U24CM602	PCC	Advanced Machine Learning	3	0	0	3	40	60	3	3
3	U24CD601	PCC	Computer Networks	3	0	0	3	40	60	3	3
4	-	PEC	Professional Elective–II	3	0	0	3	40	60	3	3
5	-	OEC	Open Elective-I	3	-	-	3	40	60	3	3
Practical/Laboratory Course											
6	U24CM6L1	PCC	Advanced Machine Learning Lab	-	-	3	3	25	50	3	1.5
7	U24CD6L1	PCC	Computer Networks Lab	-	-	3	3	25	50	3	1.5
8	U24CM6L2	HSMC	Web Design and Development Lab	-	-	2	2	25	50	-	1
Project											
9	U24CM6P1	PROJ	Mini Project	-	-	6	6	50	50	3	2
Total				**14**	**1**	**14**	**29**	**325**	**500**	**24**	**21**

L: Lecture*(Hrs/Wk/Sem)* **T**:Tutorial*(Hrs/Wk/Sem)* **P**:Practical **D**:Drawing*(Hrs/Wk/Sem)*
CIE: Continuous Internal Evaluation **AM:**AI&ML **SEE**: Semester End Examination
PCC: Program core course **PEC:** Professional Elective Course **CM:** COMPUTER SCIENCE & ENGINEERING (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
OEC: Open Elective Course **PROJ**: Project **IT:** Information Technology

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

OPEN ELECTIVE COURSES OFFERED(OE-1)

U24EE608	Fundamental of Power Electronics
U24EE609	Electrical Installation and Safety
U24CS607	Java Programming
U24IT606	Operating Systems
U24ME608	Basics Of 3-D Printing
U24ME609	Optimization Methods for Engineers
U24CE607	Construction Materials
U24CE608	Engineering Geology
U24EC607	Principles of Data Communication and Network
U24EC608	Embedded Systems
U24MB602	Total Quality Management
U24MB603	Innovation Management
U24SH601	Indian Music System
U24SH602	Introduction to Art and Aesthetics
U24CM607	Fundamentals of Machine Learning

LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY
(An Autonomous Institution)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
SCHEME OF INSTRUCTIONS & EXAMINATIONS [LR-24]

OPEN ELECTIVE COURSES

S.No.	Course Code	Category	Course Title
1	U24EE508	OEC1	Non-Conventional Energy Systems
	U24EE509		Energy Conservation and Management
	U24CS508		Database Management Systems
	U24IT506		Data Structures
	U24ME509		Basics of Mechanical Engineering
	U24ME510		Modern Manufacturing Processes
	U24CE510		Disaster Preparedness and Management
	U24CE511		Civil Engineering Principles and Practices
	U24EC507		Principles of Electronic Communication
	U24EC508		Semi-Conductor Devices
	U24MB502		Managerial Communication
	U24MB503		Managerial Science and Theory
	U24SH501		History of Science & Technology
	U24SH502		Economic Policies in India
	U24CM507		Basics of Artificial Intelligence

S.No.	Course Code	Category	Course Title
2	U24EE608	OEC2	Fundamental of Power Electronics
	U24EE609		Electrical Installation and Safety
	U24CS607		Java Programming
	U24IT606		Operating Systems
	U24ME608		Basics Of 3-DPrinting
	U24ME609		Optimization Methods for Engineers
	U24CE607		Construction Materials
	U24CE608		Engineering Geology
	U24EC607		Principles of Data Communication and Network
	U24EC608		Embedded Systems
	U24MB602		Total Quality Management
	U24MB603		Innovation Management
	U24SH601		Indian Music System
	U24SH602		Introduction to Art and Aesthetics
	U24CM607		Fundamentals of Machine Learning

S.No.	Course Code	Category	Course Title
3	U24EE711	OEC3	Introduction to Electrical Vehicles
	U24EE712		Design estimation and Costing of Electrical Systems
	U24CS711		Data Sciences
	U24IT705		Basics of Artificial Intelligence
	U24ME711		Renewable Energy Resources
	U24ME712		Cooling of Electronic Components
	U24CE711		Environmental Systems
	U24CE712		Urban Transportation System
	U24EC703		IOT and its protocols
	U24EC704		Television and Video Engineering
	U24MB702		Logistics Management
	U24MB703		Management of Start Up's
	U24SH701		Display Devices
	U24SH702		Comparative Study of Literature
	U24CM707		Basics of Cyber Security

S.No.	Course Code	Category	Course Title
4	U24EE804	OEC4	Smart Building Systems
	U24EE805		Industrial Automation
	U24CS806		Basics of Machine Learning
	U24IT802		Cloud computing
	U24ME806		Hybrid Vehicle Technology
	U24ME807		Power Plant Engineering
	U24CE806		Green Building Technology
	U24CE807		Environmental Impact Assessment
	U24EC805		Fundamentals of Wireless Communication
	U24EC806		Fundamental Digital Design using Verilog HDL
	U24MB802		Entrepreneurship
	U24MB803		E–Marketing
	U24SH801		Corrosion Science and Technology
	U24SH802		Introduction To Philosophical Thoughts
	U24CM809		Fundamentals of Computer Vision

***Note:** Open Elective Subjects not offered to the students of own department.

Course Code	Course Title						Core/Elective
U24CM601	**WEB DESIGN AND DEVELOPMENT**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**2**	**1**	**0**	**0**	**40**	**60**	**3**

Course Objectives

This course enable students to:

1. Understand the fundamentals of web development including HTML structure, CSS styling, and basic JavaScript concepts.
2. Apply CSS layout techniques such as Flexbox and Grid to design responsive and visually appealing web pages.
3. Develop dynamic and interactive web applications using JavaScript, including DOM manipulation and event handling.
4. Explore modern JavaScript (ES6+) features and asynchronous programming techniques for efficient client-side development.
5. Build full-stack web applications using React, Node.js, Express.js, and MongoDB.

Course Outcomes

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

1. Design structured web pages using HTML and style them effectively using CSS.
2. Implement responsive layouts and animations using advanced CSS techniques like Flexbox, Grid, and media queries.
3. Create interactive web applications using JavaScript with proper use of DOM manipulation and event handling.
4. Utilize ES6+ features and asynchronous programming (callbacks, promises, Fetch API) for modern web development.
5. Develop full-stack applications by integrating React for frontend and Node.js, Express.js, and MongoDB for backend.

UNIT-I

HTML Fundamentals: Basics of Web & Internet, Structure of HTML Document, Basic Tags: Headings, Paragraphs, Line Breaks, Text Formatting Tags, Comments in HTML, HTML Attributes & Lists, Links, Images & Media, Tables, Forms & Input Controls, Semantic Tags

CSS Fundamentals: What is CSS, Types, CSS Syntax & Selectors, Comments in CSS, Pseudo-classes & elements, Colors, Units & Backgrounds,

UNIT-II

CSS Layout Techniques: Box Model Display Property, Positioning, Shadows & Effects, Overflow & Visibility, Responsive Design (Media Queries), Transitions & Animations, Pagination, CSS Flex Container, Flex Items, Flex Properties, Applications, CSS Grid Container, Grid items, Grid properties, Grid Layout Features, Grid vs Flexbox,

UNIT-III

JavaScript: Introduction, Variables, Operators, Data types, Control statements, Functions, Objects, Arrays methods, DOM Manipulation: Selecting elements, changing HTML and Styling, Creating & removing elements, Form Validation, DOM Animations, Events Handling: Event types, Event listeners, Event object, Event bubbling

Advanced JavaScript: Asynchronous JavaScript: Synchronous vs Asynchronous, Callbacks, Promises, ES6+ Features, JS Modules, Classes, Fetch API

UNIT-IV

Basic React JS: Introduction to React, Why React, Single Page Applications, setting up React environment, Folder structure, JSX & Components, Props & State

Advanced React JS: Event Handling & Forms, React Hooks: useState, useEffect, useContext, useRef, API Integration, Styling in React

UNIT-V

Node JS: Introduction to Node.js, Running Node programs, Modules, require vs import, File System – Creating, Writing and Deleting files, Creating Server (HTTP Module)
Express.js: Introduction to Express.js, Routing REST API Development, Middleware, Authentication Basics, File Upload
MongoDB: Introduction to MongoDB, CRUD Operations, Schema & Models, Connecting Node with MongoDB, Querying Data

Suggested Readings :

1. **Programming the World Wide Web**: Robert W. Sebesta, *Programming the World Wide Web*, 4th Edition, Pearson Addison Wesley, 2008.
2. **HTML and CSS: Design and Build Websites**: Jon Duckett, *HTML and CSS: Design and Build Websites*, 1st Edition, Wiley Publications, 2011.
3. **JavaScript and JQuery: Interactive Front-End Web Development**: Jon Duckett, *JavaScript and JQuery: Interactive Front-End Web Development*, 1st Edition, Wiley Publications, 2014.
4. **Learning React**: Alex Banks & Eve Porcello, *Learning React*, 2nd Edition, O'Reilly Media, 2020.
5. **Web Development with Node and Express**: Ethan Brown, *Web Development with Node and Express*, 2nd Edition, O'Reilly Media, 2019.
6. **MongoDB: The Definitive Guide**: Kristina Chodorow, *MongoDB: The Definitive Guide*, 3rd Edition, O'Reilly Media, 2019.

Course Code	Course Title						Core/Elective
U24CM602	**Advanced Machine Learning**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Machine Learning	**3**	**0**	**-**	**-**	**40**	**60**	**3**

Course Objectives

This course enable students :

1. To introduce advanced concepts and methods of Machine learning.
2. To develop an understanding of the role of machine learning in massive scale automation.
3. To design and implement various machine learning algorithms in the range of real world applications.
4. To develop reinforcement Learning through feedback network, function approximation.
5. To understand ensemble methods.

Course Outcomes

After completing this course, the student will be able

1. To introduce advanced concepts and methods of Machine learning.
2. To develop an understanding of the role of machine learning in massive scale automation.
3. To design and implement various machine learning algorithms in the range of real world applications.
4. To understand Reinforcement Learning through feedback network, function approximation.
5. To understand ensemble methods.

Unit I

Artificial neural network: Introduction to ANN, Perceptron, Cost function, Gradient checking, Multi layer perceptron and back propagation algorithm, convergence and local mininma, representational power of feed forward networks, hidden layer representation, generalization, overfitting and stopping criterion.

Unit II

Bayesian learning: Probability theory and Bayes rule, bayes theorem and concept learning, Naive Bayes learning algorithm, Bayesian belief Networks, the EM algorithm.

Unit III

Decision trees: Representing concepts as Decision trees, Recursive induction of Decision trees, Best splitting attribute: Entropy and Information gain, Searching for simple trees and Computational complexity, Overfitting, noisy data and pruning.

Unit IV

Genetic algorithms: genetic algorithms, extensions, hypothesis space search, genetic programming, models of evaluation and learning, parallelizing genetic programming, sequential covering algorithms, learning first order rules, inverting resolution.

Unit V

Reinforcement Learning: learning tasks, Q learning, non deterministic rewards and actions, Reinforcement Learning through feedback network, function approximation

Ensemble methods : Bagging, Boosting and learning with ensembles, Random forests.

Suggested Readings:

1.Machine learning ,Tom Mitchel, Mc Graw Hill,1997.
2. Introduction to Machine learning ,Jeeva Jose, Khanna book publishing,2020
3. Machine Learning ,Rajiv chopra ,Khanna book publishing,2021.
4. Machine Learning and Big Data Concepts, Algorithms, Tools and Applications. Uma N Dulhare, Khaleel Ahmad,Khairol AmaliBin Ahmad, ,Scrivener Publishing,Wiley,2020.
5. Introduction to Machine learning, Ethem Apaydin, the MIT press, 2010.

Course Code	Course Title						Core/Elective
U23CD602	**COMPUTER NETWORKS**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Operating Systems	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives

1. To introduce the fundamental various types of computer networks.
2. To demonstrate the TCP/IP and OSI models with merits and demerits.
3. To explore the IP Addressing Mechanisms
4. To Understand the World Wide Web concepts.
5. Classify the routine protocols and analyze how to assign the IP address for the given network.

Course Outcomes:

1. Explain & design the various reference models and networks.
2. Identify the different types of network devices and Multiple Access Protocols.
3. Use IP addressing Scheme and to interconnect various networks and Routing mechanism
4. Explain transport layer protocols: TCP, UDP.
5. Explain and use various application layer protocols: HTTP, DNS, and SMTP, FTP etc

UNIT – I

Data Communications: Components – Direction of Data flow – Networks, Components and Categories, Network devices, Types of Connections, Topologies, Protocols, OSI model, TCP/IP Protocol Suite, Wireless LAN**,** VPN

Physical layer: Transmission modes, Multiplexing, Transmission Media, Switching, Types of Switching

UNIT – II

Data link layer: Introduction, Framing, Error Detection and Correction – Parity, CRC, Hamming code, Flow and Error Control: Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols.

Medium Access Protocols: Random access, Controlled access, Channelization, LAN– Ethernet IEEE 802.3, IEEE 802.5, IEEE 802.11

UNIT – III

Network layer: Logical Addressing - IPv4, IPv6 protocol, Subnetting, Internetworking, Tunneling, Address mapping, Packet fragmentation, , Delivery, Forwarding, **Routing**-Static routing, Default routing, Dynamic routing, Hierarchical Routing

UNIT – IV

Transport Layer: Process to Process Delivery, UDP and TCP protocols, TCP Services, TCP Segment Format, connection establishment, The TCP connection release, Data Traffic, Congestion, TCP Congestion Control, Quality of Services, Integrated Services, Differentiated Services

UNIT – V

Application Layer: Application layer services**,** Domain name space, electronic mail, FTP, WWW, Uniform Resource Locator, HTTP, SNMP, TELNET, SSH, VOIP, Bluetooth Proxy Server

Suggested Readings

1. Data Communications and Networking, Behrouz A. Forouzan , Fourth Edition TMH, 2006.
2. Computer Networks, Andrew S Tanenbaum, 4th Edition. Pearson Education, PHI.
3. Data communications and Computer Networks, P.C .Gupta, PHI.
4. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.

Course Code	Course Title						Core/Elective
U24CM603	**ARTIFICIAL NEURAL NETWORKS**						**PE-II**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Machine Learning, Artificial Intelligence	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students to:

1. Understand the role of neural networks in engineering, artificial intelligence, and cognitive modeling.
2. Provide knowledge of types of neural networks.
3. Practical approach for using artificial neural networks in various technical, organizatorial and economic applications.
4. Prospects for use of artificial neural networks in products.
5. Understand the mathematical equations, and the role of the various parameters.

Course Outcomes:

After completing this course, the student will be able to

1. Understand the difference between biological neuron and artificial neuron.
2. Understand the application areas of neural networks
3. Understand building blocks of Neural Networks.
4. Develop neural network models.
5. Design and develop applications using neural networks.

UNIT-I

General characteristics of the human brain, Introduction to Biological Neural Networks, Nerve structure and synapse, Basic concepts of Neural Networks, Characteristics of Neural Networks, Terminologies, Applications of the artificial neural networks.

UNIT-II

Structure of a neural net (topology), Directed graphs, Models of Neuron, Neural Network Architectures, Artificial Neuron, Activation functions, Threshold function, Piecewise linear function, Sigmoidal function, Supervised learning, Unsupervised learning, Re-enforcement learning.

UNIT-III

Knowledge Representation, Artificial Intelligence, Learning rules, Error correction learning, Memory based learning, Hebbian learning, Competitive learning, Boltzmann learning, Single layer perceptron, Multilayer perceptron, Back propagation, Recurrent networks, Network Pruning.

UNIT-IV

Adaptive networks, Supervised Learning Neural Networks, Decision-based neural networks, Hierarchical neural networks, Probabilistic neural network, Radial basis function networks, Comparison of RBF Networks and Multilayer perceptron.

UNIT-V

Classification of linearly separable patterns, Boltzmann machine, Sigmoid Belief Networks, Helmholtz machine, Support vector machines, Self organization maps, Genetic Algorithms, Optimization, Prediction Systems, speech and decision-making.

Suggested Readings:

1. "Neural Networks a comprehensive Foundation",S. Haykin, Prentice-Hall India,2nd Edition,1997.
2. "Fundamentals of Neural Networks, Architecture, Algorithms, and Applications", Laurene Fausett, Prentice Hall,ist Edition, 1993.
3. "The Handbook of Brain Theory and Neural Networks", Michael A Arbib, MIT Press, 2ndEdition, 2003.
4. Introduction to Artificial Neural Systems, Jacek M. Zurada, Jaico, Publ. House, 1994.

Course Code	Course Title						Core/Elective
U24CM604	**SOFTWARE TESTING METHODOLOGIES**						**PE-II**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
SE	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students :

1. To understand the fundamental concepts, principles, and processes of software testing.
2. To learn various test case design techniques using black box and white box approaches.
3. To gain knowledge of different levels of testing and their role in software quality assurance.
4. To develop an understanding of test management practices and organizational aspects of testing.
5. To introduce software test automation concepts, tools, and metrics for improving testing efficiency.

Course Outcomes:

After completing this course, the student will be able to

1. Explain software testing concepts, principles, and defect management processes.
2. Apply appropriate test case design techniques for effective software validation.
3. Analyze and perform different levels of testing for software quality assurance.
4. Demonstrate understanding of test management, planning, and reporting activities.
5. Evaluate and apply test automation tools, techniques, and metrics in software testing.

UNIT – I: Introduction

Testing as an engineering activity, testing as a process, testing maturity model, testing axioms, basic definitions, software testing principles, tester's role in a software development organization, origins of defects, cost of defects, defect classes, defect repository and test design, defect examples, developer and tester support in developing a defect repository.

UNIT – II: Test Case Design

Test case design strategies, black box approach to test case design, boundary value analysis, equivalence class partitioning, state-based testing, cause-effect graphing, compatibility testing, user documentation testing, domain testing, random testing, requirements-based testing, white box approach to test design, test adequacy criteria, static testing vs structural testing, code functional testing, coverage and control flow graphs, covering code logic and paths, code complexity testing, additional white box testing approaches, evaluating test adequacy criteria.

UNIT – III: Levels of Testing

Need for levels of testing, unit testing, unit test planning, designing unit tests, test harness, executing unit tests and recording results, integration testing, designing integration tests, integration test planning, scenario testing, defect bash elimination, system testing, acceptance testing, performance testing, regression testing, internationalization testing, ad-hoc testing, alpha and beta testing, testing object-oriented systems, usability and accessibility testing, configuration testing, compatibility testing, testing documentation, website testing.

UNIT – IV: Test Management

People and organizational issues in testing, organizational structures for testing teams, testing services, test planning, test plan components, test plan attachments, locating test items, test management, test process, reporting test results, role of test specialist, skills required for a test specialist, building a testing group, structure of testing group, technical training program.

UNIT – V: Test Automation

Software test automation, skills required for automation, scope of automation, design and architecture for automation, requirements for test tools, challenges in automation, test metrics and measurements, project metrics, progress metrics, productivity metrics.

Suggested Readings:

1. Software Testing: A Craftsman's Approach, CRC Press, 4th Edition.
2. How to Break Software, Addison-Wesley.
3. Agile Testing: A Practical Guide for Testers and Agile Teams, Addison-Wesley.
4. Lessons Learned in Software Testing, Wiley Publications.

Course Code	Course Title						Core/Elective
U24CM605	**CLOUD COMPUTING**						**PE-II**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
SE	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students :

1. This course provides an insight into cloud computing.
2. Topics covered include- Cloud Computing Architecture, Deployment Models, Service Models, Technological Drivers for Cloud Computing, Networking for Cloud Computing and Security in Cloud Computing.

Course Outcomes:

After completing this course, the student will be able to

1. Understand different computing paradigms and potential of the paradigms and specifically cloud computing.
2. Understand cloud service types, cloud deployment models and technologies supporting and driving the cloud.
3. Acquire the knowledge of programming models for cloud and development of software application that runs the cloud and various services available from major cloud providers.
4. Understand the security concerns and issues in cloud computing.
5. Acquire the knowledge of advances in cloud computing.

UNIT – I

Computing Paradigms, Cloud Computing Fundamentals, Cloud Computing Architecture and Management

UNIT – II

Cloud Deployment Models, Cloud Service Models, Technological Drivers for Cloud Computing:SOA and Cloud, Multicore Technology, Web 2.0 and Web 3.0, Pervasive Computing, Operating System, Application Environment

UNIT – III

Virtualization, Programming Models for Cloud Computing: MapReduce, Cloud Haskell, Software Development in Cloud

UNIT – IV

Networking for Cloud Computing:Introduction, Overview of Data Center Environment, Networking Issues in Data Centers, Transport Layer Issues in DCNs, Cloud Service Providers

UNIT – V

Security in Cloud Computing, and Advanced Concepts in Cloud Computing

Suggested Readings:

1. Chandrasekaran, K. *Essentials of Cloud Computing*. CRC Press, 2014
2. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011
3. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
4. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

Course Code	Course Title						Core/Elective
U24CM606	**HUMAN COMPUTER INTERACTION**						**PE-II**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
SE	**3**	**-**	**-**	**-**	**40**	**60**	**3**

Course Objectives:

This course enable students :

1. To understand the fundamentals, principles, and importance of Human Computer Interaction (HCI) and User Interface (UI) design.
2. To analyze human factors, cognitive aspects, and interaction models involved in designing effective user interfaces.
3. To develop skills in screen design, navigation design, multimedia integration, and usability-oriented interface development.
4. To study various HCI design methodologies, evaluation techniques, prototyping methods, and usability standards.
5. To explore advanced concepts such as ubiquitous computing, virtual reality, augmented reality, and information visualization in HCI systems.

Course Outcomes:

After completing this course, the student will be able to

1. Understand the concepts, principles, and significance of Human Computer Interaction and user interface design.
2. Apply human-centered design principles to create effective and user-friendly screen and interface designs.
3. Analyze and evaluate different interaction techniques, usability methods, and interface components for software systems.
4. Design prototypes and perform usability evaluation using standard HCI principles, heuristics, and evaluation techniques.
5. Demonstrate knowledge of advanced HCI applications including ubiquitous computing, augmented reality, virtual reality, and data visualization.

UNIT - I

Introduction: Importance of user Interface – definition, importance of good design. Benefits of good design. A brief history of Screen design.

The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics- Principles of user interface.

UNIT - II

Design process – Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions.

Screen Designing: Design goals – Screen planning and purpose, organizing screen elements, ordering of screen data and content – screen navigation and flow – Visually pleasing composition – amount of information – focus and emphasis – presentation information simply and meaningfully – information retrieval on web – statistical graphics – Technological consideration in interface design.

UNIT- III

Windows – New and Navigation schemes selection of window, selection of devices based and screen-based controls. Components – text and messages, Icons and increases – Multimedia, colors, uses problems, choosing colors.

UNIT- IV

HCI in the software process, The software life cycle Usability engineering Iterative design and prototyping Design Focus: Prototyping in practice Design rationale Design rules Principles to support usability Standards Golden rules and heuristics HCI patterns Evaluation techniques, Goals of evaluation, Evaluation through expert analysis,

Evaluation through user participation, Choosing an evaluation method. Universal design, Universal design principles Multi-modal interaction

UNIT- V

Cognitive models Goal and task hierarchies Design Focus: GOMS saves money Linguistic models The challenge of display-based systems Physical and device models Cognitive architectures Ubiquitous computing and augmented realities Ubiquitous computing applications research Design Focus: Ambient Wood – augmenting the physical Virtual and augmented reality Design Focus: Shared experience Design Focus: Applications of augmented reality Information and data visualization Design Focus: Getting the size right.

Suggested Readings:

1. The essential guide to user interface design, Wilbert O Galitz, Wiley Dream Tech. Units
2. Human – Computer Interaction. Alan Dix, Janet Fincay, Gre Goryd, Abowd, Russell Bealg, Pearson Education

Course Code	Course Title						Core/Elective
U24CM6L1	**ADVANCED MACHINE LEARNING LAB**						**Core**
Prerequisite	**Contact Hours per Week**				**CIE**	**SEE**	**Credits**
	L	**T**	**D**	**P**			
Java/Python	**-**	**-**	**3**	**3**	**25**	**50**	**1.5**

Course Objectives:

This course enable students to:

1. To implement classification algorithms.
2. To implement regression algorithms.
3. To implement clustering techniques.
4. To implement neural networks.
5. Learn to implement the different protocols

Course Outcomes:

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

1. Implement various protocols using classification and regression techniques.
2. Implement clustering mechanisms
3. Implement Decision trees.
4. Implement and Analyze various random forest techniques.
5. Implement Decision trees.

List of Experiments:

1. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
2. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.
3. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
4. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
5. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
6. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.
7. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data setfor clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
8. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem

Suggested Readings:

1. Machine learning, Tom Mitchel, Mc Graw Hill,1997.
2. Introduction to Machine learning ,Jeeva Jose, Khanna book publishing,2020

Course Code	Course Title						Core / Elective
U24CD6L2	**COMPUTER NETWORKS LAB**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
C	**-**	**-**	**-**	**3**	**25**	**50**	**1.5**

Course Objectives:

1. To familiarize students with the basic principles, protocols, and layered architecture of computer networks.
2. To reinforce the theory learned in the classroom via hands-on practice.
3. Develop practical skills in network configuration and management
4. To design and implement client-server applications using protocols such as TCP and UDP.
5. To develop simple socket programming applications in languages like C, Java, or Python.

Course Outcomes:

1. Explain and apply basic networking concepts, layered architecture, and protocol operations in practical scenarios.
2. Configure and troubleshoot network devices and design small-scale network topologies.
3. Develop and implement client–server communication using socket programming in TCP and UDP.
4. Simulate and analyze network protocols and topologies using simulation tools .
5. Capture, inspect, and analyze network traffic using tools such as Wireshark to understand protocol behavior and performance metrics.

Sr No	**Experiment No.**	**NAME OF THE EXPERIMENT**
01	Exp-01	**Network Devices and Networking Commands basics**
02	Exp-02	**<u>Using Cisco Packet Tracer (CPT)</u>** **Build: The Network,** Design the topologies, Connect 4 computers in the LAN using Switch, Configure IP addressing on all PC using 192.168.1.0/24 network, Check connectivity between all the PÇ's using Ping command, traceroute, Configure IP Address as diagram & rules, Verify the interface status
03	Exp-03	**Configuring Static Routing using Packet Tracer** a) Assign IP address according to diagram b) Make sure that interfaces used should be in UP UP state c) Configure static routing d) Verify routing table and reachability between LAN's (Using PING & TRACE commands)

04	Exp-04	**Configuring Default Routing using Packet Tracer** a) Design the topology b) Assign IP address according to diagram c) Make sure that interfaces used should be in UP-UP state d) Configure Default route used on R1 & R3, Static routing on R2 e) Verify routing table and reachability between LAN's (Using PING & TRACE commands)
05	Exp-05	**Configuring Dynamic Routing using Packet Tracer** a) Design the topology b) Assign IP address according to diagram c) Make sure that interfaces used should be in UP UP state d) Configure Dynamic Routing using EIGRP 100 e) Verify routing table and reachability between LAN's (Using PING & TRACE commands)
06	Exp-06	a) Write a Program to demonstrate connectionless (UDP) b) Echo Service Using Well Known / Standard Port. **c)** Write a Program to demonstrate connectionless (UDP) Date and Time Service Using Well Known / Standard Port.
07	Exp-07	a. Write a Program to Demonstrate Connection oriented (TCP) Echo Service Using Well Known / Standard Port b. Write a Program to Demonstrate Connection oriented (TCP) Date and Time Service Using Well Known / Standard Port
08	Exp-08	1. Write a program to implement echo service using RPC. 2. Write a program to implement date and time service using RPC. Program name:day.x 3. Write a program to implement addition of two numbers using RPC.
09	Exp-09	Write a Program to Implement Mini DNS Service
10	Exp-10	Demonstrate by witing a program on Bit Stuffing/ Character stuffing, Link state routing and Distance Vector routing algorithms
11	Exp-Extra	Beyond the syllabus: NS2 simulations for Network analysis

Course Code	Course Title					Core/ Elective
U24CM6L2	**WEB DESIGN AND DEVELOPMENT LAB**					**Core**
Prerequisite	**Contact Hours per Week**			CIE	SEE	Credits
	L	T	D/P			
	-	-	2	25	50	1

Course Objectives:

1. Provide hands-on experience in designing and developing responsive web interfaces using HTML, CSS, and JavaScript.
2. Enable students to implement client-side scripting techniques including DOM manipulation, event handling, and API integration.
3. Develop skills in modern frontend frameworks by building component-based applications using React.
4. Introduce server-side programming concepts using Node.js and Express.js for backend development.
5. Equip students with database integration skills to build full-stack applications using MongoDB.

Course Outcomes:

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

1. Design and develop responsive and interactive web applications using HTML, CSS, and JavaScript.
2. Apply client-side scripting techniques including DOM manipulation, local storage, and API integration.
3. Develop frontend applications using React concepts such as components, props, state, and hooks.
4. Build server-side applications using Node.js and Express.js with basic authentication mechanisms.
5. Implement full-stack web applications by integrating frontend, backend, and MongoDB database operations.

LIST OF EXPERIMENTS

Frontend Development Projects

1. Develop a Responsive Portfolio Website with multiple sections (Home, About, Projects, Contact) using HTML, CSS, and basic animations.
2. Design a Student Management System (Frontend) to add, edit, and delete student details using forms and DOM manipulation.
3. Create a Quiz Application with multiple questions, timer, score calculation, and result display.
4. Build an Image Gallery with Filters including category-based filtering and lightbox preview.

Client-Side Scripting Projects

5. Develop a Weather Application to fetch and display real-time weather data using API.
6. Build an Expense Tracker to manage income and expenses with total balance calculation using local storage.

React-Based Frontend Projects

7. Develop a React To-Do Application with persistent data using local storage and hooks.
8. Create a GitHub Profile Viewer to fetch and display user details (repositories, followers) using API integration.

Server-Side Development (Node.js)

9. Create a Notes Application to add, edit, and delete notes with browser storage support.

Backend Development with Express.js

10. Build a Simple Authentication System with user registration and login functionality.

Database Integration (MongoDB)

11. Create a Basic Blog Application with full CRUD operations (frontend + backend).
12. Implement a Contact Form with Database Storage to store and retrieve user queries using MongoDB.

Suggested Readings:

1. **Jon Duckett** – HTML and CSS: Design and Build Websites, Wiley Publications.
2. **Jon Duckett** – JavaScript and jQuery: Interactive Front-End Web Development, Wiley Publications.
3. **Ethan Brown** – Web Development with Node and Express, O'Reilly Media.
4. **Alex Banks & Eve Porcello** – Learning React, O'Reilly Media.
5. **Brad Dayley** – Node.js, MongoDB and Angular Web Development, Addison-Wesley.