Semester- II

Machine learning and Data analytics using Python			
Course Code	MMC201	CIE Marks	50
Teaching Hours/Week (L:P: SDA/T/T)	2:2:0	SEE Marks	50
Total Hours of Pedagogy	50	Total Marks	100
Credits	04	Exam Hours	03

Course Learning Objectives:

1. Understand foundational concepts in machine learning and data analytics.
2. Gain proficiency in Python for data analysis and machine learning tasks.
3. Learn and apply various machine learning algorithms and techniques.
4. Develop skills in data preprocessing, visualization, and model evaluation.
5. Prepare students for industry roles involving data-driven decision making and predictive modeling.

Module-1	08 Hours

Introduction to Machine Learning and Python:

Introduction to Machine Learning: Definition and importance of machine learning, Types of machine learning: Supervised, unsupervised, and reinforcement learning, Applications of machine learning in various domains.

Python for Data Analysis: Introduction to Python programming, Python libraries for data analysis: NumPy, Pandas, Matplotlib, Data manipulation and visualization using Pandas and Matplotlib.

Data Preprocessing: Data cleaning and transformation, Handling missing values and outliers, Feature scaling and normalization.

Teaching Learning Process:

Lectures with PowerPoint presentations, Hands-on coding exercises using Jupyter notebooks, Interactive discussions and problem-solving sessions, Assignments and quizzes for assessment.

Module-2	08 Hours

Supervised Learning:

Regression: Linear regression, Polynomial regression, Model evaluation metrics: MAE, MSE, RMSE. Classification: Logistic regression, K-Nearest Neighbors (KNN), Decision Trees and **Random Forests, Model evaluation metrics:** Accuracy, precision, recall, F1-score, ROC-AUC.

Model Training and Evaluation: Train-test split and cross-validation, Hyper parameter tuning using GridSearchCV, Overfitting and underfitting.

Teaching Learning Process:

Lab exercises on regression and classification models, Practical coding sessions with real-time problem-solving, Group projects on developing and evaluating supervised learning models, Continuous assessment through quizzes and coding challenges.

Module-3	08 Hours

Unsupervised Learning:

Clustering: K-Means clustering, Hierarchical clustering, Evaluation of clustering results. **Dimensionality Reduction:** Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) t-Distributed Stochastic Neighbor Embedding (t-SNE).

Association Rule Learning: Apriori algorithm, Market Basket Analysis, Evaluation metrics for association rules

Teaching Learning Process:

Lab exercises on clustering and dimensionality reduction, Practical coding sessions with unsupervised learning techniques, Group projects on applying unsupervised learning to real-world data, Continuous assessment through quizzes and practical tests.

Module-4	08 Hours

Advanced Machine Learning Techniques:

Ensemble Methods: Bagging and Boosting, Gradient Boosting Machines (GBM), Extreme Gradient Boosting (XGBoost).

Support Vector Machines (SVM): Linear and non-linear SVM, Kernel trick, Model evaluation and tuning.

Neural Networks and Deep Learning: Introduction to neural networks, Building and training neural networks using TensorFlow and Keras, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN).

Teaching Learning Process:

Practical sessions on advanced machine learning techniques, Interactive coding exercises to implement neural networks, Group projects on applying advanced techniques to complex data problems, Continuous assessment through quizzes and practical tests.

Module-5	08 Hours

Data Analytics and Real-World Applications:

Exploratory Data Analysis (EDA): Data visualization techniques, Statistical analysis and hypothesis testing, Identifying patterns and insights from data.

Time Series Analysis: Introduction to time series data, Time series forecasting using ARIMA and Prophet, Evaluating time series models.

Integrating Machine Learning Models: Deployment of machine learning models, Building web applications with Flask and Django, Case studies on real-world applications of machine learning.

Teaching Learning Process:

Lab exercises on EDA and time series analysis, Practical sessions on deploying machine learning models, Group discussions on real-world case studies, Final project presentation and assessment.

Sl. NO	Experiments
1	Implement and demonstratethe FIND-Salgorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2	For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithmto output a description of the set of all hypotheses consistent with the training examples.
3	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.
4	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.
5	Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions.
6	Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
7	Write a program to demonstrate Regression analysis with residual plots on a given data set.
8	Write a program to compute summary statistics such as mean, median, mode, standard deviationand variance of the given different types of data.
9	Write a program to implement k-Means clustering algorithm to cluster the set of data stored in .CSV file.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. Minimum passing marks in SEE is 40% of the maximum marks of SEE. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 50% (50 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of IPCC

1. Two Tests each of 20 Marks
2. Two assignments each of 10 Marks/One Skill Development Activity of 20 marks
3. Total Marks of two tests and two assignments/one Skill Development Activity added will be CIE for 60 marks, marks scored will be proportionally scaled down to 30 marks.

CIE for the practical component of IPCC

- On completion of every experiment/program in the laboratory, the students shall be evaluated and marks shall be awarded on the same day. The 15 marks are for conducting the experiment and preparation of the laboratory record, the other 05 marks shall be for the test conducted at the end of the semester.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write- ups are added and scaled down to 15 marks.
- The laboratory test at the end /after completion of all the experiments shall be conducted for 50 marks and scaled down to 05 marks. Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for 20 marks.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (duration 03 hours)

1. The question paper will be set for 100 marks and marks scored will be scaled down proportionately to 50 marks.
2. The question paper will have ten questions. Each question is set for 20 marks.
3. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
4. The students have to answer 5 full questions, selecting one full question from each module.

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper shall include questions from the practical component).

- The minimum marks to be secured in CIE to appear for SEE shall be the 15 (50% of maximum marks-30) in the theory component and 10 (50% of maximum marks -20) in the practical component. The laboratory component of the IPCC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included. The maximum of 04/05 questions to be set from the practical component of IPCC, the total marks of all questions should not be more than the 20 marks.

SEE will be conducted for 100 marks and students shall secure 40% of the maximum marks to qualify in the SEE. Marks secured will be scaled down to 50. (Student has to secure an aggregate of 50% of maximum marks of the course(CIE+SEE)

Suggested Learning Resources:

Text Books:

1. Practical Machine Learning for Data Analysis Using Python Abdulhamit Subasi.
2. Advance Machine Learning with Python by John Hearty.

Reference Textbooks:

1. "Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and
Tensor Flow " by Sebastian Raschka and Vahid Mirjalili.
2. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron.
3. "Introduction to Machine Learning with Python".

Web links and Video Lectures (e-Resources):

- https://youtu.be/7eh4d6sabA0?si=JWHNJRVS6NhQOPYe
- https://youtu.be/kqtD5dpn9C8?si=LBep4HWaMFRrAPsD
- https://youtu.be/4SJ7bEILPJk?si=5LurvjzUOuCew1W9

Skill Development Activities Suggested

- The students with the help of the course teacher can take up technical –activities which will enhance their skill or the students should interact with industry (small, medium and large), understand their problems or foresee what can be undertaken for study in the form of research/testing/projects, and for creative and innovative methods to solve the identified problem. The prepared report shall be evaluated for CIE marks.

Course outcome (Course Skill Set)

At the end of the course the student will be able to:

Sl. No.	Description	Blooms Level
CO1	Demonstrate an understanding of machine learning concepts and techniques.	L3

CO2	Perform data preprocessing and exploratory data analysis using Python.	L4
CO3	Develop and evaluate machine learning models using Python libraries.	L5
CO4	Apply machine learning algorithms to real-world data problems.	L4
CO5	Integrate data analytics and machine learning models into practical applications.	L5

Mapping of COS and POs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1		X						X
CO2		X						
CO3			X	X				
CO4		X		X				X
CO5			X					X

Industry-Relevant Key Points:

- Emphasis on coding standards and best practices.
- Integration of version control systems (e.g., Git) in project work.
- Exposure to industry-standard tools and frameworks.
- Real-world application development projects.
- Focus on collaborative development and agile methodologies.

Semester- II

Object Oriented Programming using Java			
Course Code	MMC202	CIE Marks	50
Teaching Hours/Week (L:P:SDA/T)	4:0:0	SEE Marks	50
Total Hours of Pedagogy	50	Total Marks	100
Credits	4	Exam Hours	3

Course Learning objectives:

- This subject will help to improve the analytical skills of object oriented programming Formal introduction to Java programming language
- Overall development of problem solving and critical analysis

Module-1

The History and Evolution of Java: The Byte code, Features of Java An overview of Java: Object-Oriented Programming, Structure of a Java program, Data Types and Variables, Type conversion and casting, Arrays

Classes: Fundamentals, Declaring Objects, Assigning Object Reference Variables, Methods, Constructors, this Keyword, Garbage Collection, Stack application

Methods and Classes: Overloading Methods, Using Objects as Parameters, Argument Passing, Returning Objects, Access Control, static, final, Command-Line Arguments

Module-2

Inheritance: Basic concepts, Member Access and Inheritance, Practical Example Inheritance types, super, constructors, Method Overriding, Dynamic Method Dispatch, Abstract Classes, final with inheritance.

String Handling: String Constructor, String length, Special string Operations, Character Extraction, String comparison, Modifying a string, String Buffer

Generics: About Generics, A simple Generic Example, General class with Two Type Parameters, General form of generic class

Module-3

Packages and Interfaces: Packages, Packages and member access, Importing packages, Interfaces, Default interface methods, Use static methods in an interface, Private Interface methods.

Exception handling: Fundamentals, Exception types, uncaught exceptions, try and catch, multiple catch clauses, nested try statements, throw, throws, finally, Java's built-in exceptions, User-defined exceptions.

Module-4

Multithreaded Programming: Java thread model, main thread, creating thread, creating multiple threads, isalive( ) and Join( ), thread priorities, synchronization

Input/Output: Exploring java.io - The I/O Classes and Interfaces, The Byte Streams

Module-5

Event Handling: Two Event Handling Mechanisms, The Delegation Event Model, EventsEvent Sources, Event Listeners, Event Classes- The MouseEventClass, Event Listener Interfaces-The MouseListener Interface, the MouseMotionListener Interface, Delegation
Event Model – Handling Mouse Events.
AWT: Working with Windows, Graphics and Text
AWT Classes, Window Fundamentals, Working with Frame Windows, Graphic

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. Minimum passing marks in SEE is 40% of the maximum marks of SEE. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 50% (50 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

1. Two Unit Tests each of **25 Marks**
2. Two assignments each of **25 Marks**or **oneSkill Development Activity of 50 marks** to attain the COs and POs
3. The sum of two tests, two assignments/skill Development Activities, will be **scaled down to 50 marks**
4. **CIE methods /question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

Semester-End Examination:

1. The SEE question paper will be set for 100 marks and the marks scored will be proportionately reduced to 50.
2. The question paper will have ten full questions carrying equal marks.
3. Each full question is for 20 marks. There will be two full questions (with a maximum of four sub-questions) from each module.
4. Each full question will have a sub-question covering all the topics under a module.
5. The students will have to answer five full questions, selecting one full question from each module

Suggested Learning Resources:

Prescribed Text Book:: Java the Complete Reference Eleventh Edition by Herbert Schildt ,Tata McGraw-hill Edition , 2019

Reference Text Books:

1) Introduction to JAVA Programming 9th Edition by Y. Daniel Liang , Pearson education, 2012
2) Programming in JAVA 5.0 1st Edition by James P Cohoon, Jack W Davidson, TATA McGraw hill,2006

Web links and Video Lectures (e-Resources):

1) https://ia800303.us. archive.org/ 26/items/ JavaJavaJavaObjectorientedProblemSolving/ jjj-os.pdf
2) http://people.reed.edu/ ~jerry/121/materials/ artsciencejava.pdf
3) https://upload.wikimedia. org/wikipedia/ commons/e/e7/ Java_Programming.pdf
4) https://onlinecourses.swayam2.ac.in/aic20_sp1 3/preview
5) https://onlinecourses.swayam2.ac.in/aic20_sp1
6) https://www.classcentral.com/course/coursera-object-oriented-programming-in-java-4212

Skill Development Activities Suggested

- The students with the help of the course teacher can take up technical –activities which will enhance their skill or the students should interact with industry (small, medium and large), understand their problems or foresee what can be undertaken for study in the form of research/testing/projects, and for creative and innovative methods to solve the identified problem. The prepared report shall be evaluated for CIE marks.

Course outcome (Course Skill Set)

At the end of the course the student will be able to :

1. Understand the basic principles of the object-oriented programming
2. Demonstrate an introductory understanding of graphical user interfaces, multi
3. Apply the knowledge of Java concepts to find the solution for a given problem.
4. Analyse the given Java application for correctness/functionalities.
5. Develop Java programs / applications for a given requirement.

-

Sl. No.	Description	Blooms Level
CO1	Solve the given problem by applying OOP using java	L2
CO2	Apply the fundamentals of Inheritance	L2
CO3	Identify and list the different applications of event handling	L3

Mapping of COS and POs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1			**X**					
CO2								**X**
CO3				**X**				
CO4			**X**					
CO5				**X**				

Semester- II

Data Structure and Algorithms			
Course Code	MMC203	CIE Marks	50
Teaching Hours/Week(L:P: SDA/T)	4:0:0	SEE Marks	50
Total Hours of Pedagogy	50	Total Marks	100
Credits	04	Exam Hours	03

Course Learning Objectives:

1. To understand and implement fundamental data structures.
2. To develop efficient algorithms for solving problems.
3. To analyze the time and space complexity of algorithms.
4. To gain practical experience in applying data structures and algorithms to real-world problems.
5. To prepare students for industry roles requiring strong foundations in data structures and algorithmic thinking.

Module-1 **10 Hours**

Introduction to Data Structures and Algorithms:

Basic Concepts: Definition and importance of data structures, Abstract Data Types (ADTs), Algorithm analysis: Time and space complexity, **Big O notation. Arrays:** Definition and operations: Insertion, deletion, traversal, Multidimensional arrays, Applications of arrays. **Linked Lists:** Singly linked list: Creation, insertion, deletion, traversal, Doubly linked list and circular linked list, Applications of linked lists.

Teaching Learning Process:

Lectures with PowerPoint presentations, Hands-on coding exercises in C, Interactive discussions and problem-solving sessions, Assignments and quizzes for assessment.

Module-2 **10 Hours**

Stacks, Queues, and Recursion:

Stacks: Definition and operations: Push, pop, peek, Applications: Expression evaluation, backtracking, function calls. **Queues:** Definition and operations: Enqueue, dequeue, front, rear, Types: Circular queue, priority queue, double-ended queue (deque), Applications of queues. **Recursion:** Definition and principles of recursion, Recursive algorithms: Factorial, Fibonacci series, Tower of Hanoi, Analysis of recursive algorithms.

Teaching Learning Process:

Case studies and real-world examples, Practical coding sessions, Group discussions and problem-solving exercises, Mid-term project focusing on stack and queue applications.

Module-3 **10 Hours**

Trees and Graphs:

Trees: Definition and terminology: Root, leaf, internal node, height, depth, Binary trees: Traversal (preorder, inorder, postorder), creation, insertion, deletion, Binary search trees (BST), AVL trees, B-trees. **Graphs:** Definition and terminology: Vertices, edges, adjacency list, adjacency matrix, Graph traversal algorithms: Depth-first search (DFS), breadth-first search (BFS), Shortest path algorithms: Dijkstra's algorithm, Floyd-Warshall algorithm.

Teaching Learning Process:

Lab exercises on tree and graph implementations, Interactive coding sessions with real-time problem-solving, Group projects to develop tree and graph applications, Continuous assessment through quizzes and coding challenges.

Module-4 **10 Hours**

Sorting and Searching Algorithms:

Sorting Algorithms: Basic concepts and classification, Comparison-based sorting: Bubble sort, selection sort, insertion sort, quicksort, mergesort, heapsort, Non-comparison-based sorting: Radix sort, counting sort. **Searching Algorithms:** Linear search and binary search, Search in linked lists, trees, and graphs, Hashing: Hash functions, collision resolution techniques (chaining, open addressing).

Teaching Learning Process:

Demonstrations and hands-on coding practice, Problem-solving sessions with practical use cases, Case studies on the application of sorting and searching algorithms, Assignments and group activities to solidify understanding.

Module-5 **10 Hours**

Advanced Data Structures and Applications:

Advanced Data Structures: Heaps: Definition, operations, heap sort, applications, Trie: Definition, operations, applications in dictionary and spell-checking, Segment trees and Fenwick trees: Definition, operations, range queries. **Algorithm Design Techniques:** Divide and conquer, Greedy algorithms, Dynamic programming. **Industry Applications:** Real-world applications of data structures and algorithms, Best practices in data structure and algorithm implementation, Case studies of complex problem-solving using advanced data structures.

Teaching Learning Process:

Practical sessions on advanced data structures, Industry guest lectures, Project-based learning with real-world applications, Final project presentation and assessment.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. Minimum passing marks in SEE is 40% of the maximum marks of SEE. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 50% (50 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

1. Three Unit Tests each of **20 Marks**
2. Two assignments each of **20 Marks** or **one Skill Development Activity of 40 marks** to attain the COs and POs

The sum of three tests, two assignments/skill Development Activities, will be **scaled down to 50 marks CIE methods /question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course**.

Semester End Examination:

1. The SEE question paper will be set for 100 marks and the marks scored will be proportionately reduced to 50.
2. The question paper will have ten full questions carrying equal marks.
3. Each full question is for 20 marks. There will be two full questions (with a maximum of four sub-questions) from each module.
4. Each full question will have a sub-question covering all the topics under a module.

Suggested Learning Resources:

Text Books:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

Reference books:

1. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014.
2. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
3. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2 nd Ed, McGraw Hill, 2013
4. A M Tenenbaum, Data Structures using C, PHI, 1989
5. Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.
6. Introduction to the Design and Analysis of Algorithms, Anany Levitin: 2nd Edition, 2009. Pearson.
7. Computer Algorithms/C++, Ellis Horowitz, SatrajSahni and Rajasekaran, 2nd Edition, 2014, Universities Press.

8. Algorithms, Kenneth A Berman and Jerome L Paul, Cengage Learning India Pvt Ltd, 2002 edition.

Web links and Video Lectures (e-Resources):

- https://www.youtube.com/watch?v=BBpAmxU_NQo
- https://www.youtube.com/watch?v=8hly31xKli0
- https://archive.nptel.ac.in/courses/106/106/106106127/

Skill Development Activities Suggested

- The students with the help of the course teacher can take up technical –activities which will enhance their skill or the students should interact with industry (small, medium and large), understand their problems or foresee what can be undertaken for study in the form of research/testing/projects, and for creative and innovative methods to solve the identified problem. The prepared report shall be evaluated for CIE marks.

Course outcome (Course Skill Set)

At the end of the course the student will be able to:

Sl. No.	Description	Blooms Level
CO1	Demonstrate an understanding of fundamental data structures and algorithms.	L2
CO2	Implement and manipulate data structures such as arrays, linked lists, stacks, queues, trees, and graphs.	L4
CO3	Develop algorithms for searching, sorting, and optimization problems.	L2
CO4	Analyze the efficiency and correctness of algorithms.	L2
CO5	Apply data structures and algorithms to solve complex problems in various domains.	L4

Industry-Relevant Key Points:

- Emphasis on coding standards and best practices.
- Integration of version control systems (e.g., Git) in project work.
- Exposure to industry-standard tools and frameworks.
- Real-world application development projects.
- Focus on collaborative development and agile methodologies.

Mapping of COS and POs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	**X**	X						
CO2			X					
CO3				X				
CO4								X
CO5								X

Semester- II

Software Engineering			
Course Code	MMC204	CIE Marks	50
Teaching Hours/Week (L:P:SDA/T)	2:0:1	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03

Course Learning objectives:

- Understand the importance of various Software Engineering Lifecycle Models.
- Document the Software Requirements Specification (SRS) for the identified system.
- Gain knowledge of the System Analysis and Design concepts using UML.

Module-1

SOFTWARE PROCESS AND AGILE DEVELOPMENT: Introduction to Software Engineering, Software Process, Perspective and Specialized Process Models –Introduction to Agility-Agile process-Extreme programming-XP Process-Case Study.

Module-2

REQUIREMENTS ANALYSIS AND SPECIFICATION: Requirement analysis and specification **–** Requirements gathering and analysis **–** Software Requirement Specification **–** Formal system specification – Finite State Machines – Petrinets –Object modelling using UML – Use case Model – Class diagrams – Interaction diagrams – Activity diagrams – State chart diagrams – Functional modelling – Data Flow Diagram- CASE TOOLS.

Module-3

SOFTWARE DESIGN: Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence – Design patterns – Model-view-controller – Publish-subscribe – Adapter – Command – Strategy – Observer – Proxy – Facade – Architectural styles – Layered - Client Server - Tiered - Pipe and filter- User interface design-Case Study

Module-4

TESTING AND MAINTENANCE : Testing – Unit testing – Black box testing– White box testing – Integration and System testing–Regression testing – Debugging - Program analysis – Symbolic execution – Model Checking Case Study.

Module-5

PROJECT MANAGEMENT: Software Project Management- Software Configuration Management - Project Scheduling- DevOps: Motivation-Cloud as a platform-Operations- Deployment Pipeline:Overall Architecture Building and Testing-Deployment- Tools- Case Study.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. Minimum passing marks in SEE is 40% of the maximum marks of SEE. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 50% (50 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

1. Two Unit Tests each of **25 Marks**
2. Two assignments each of **25 Marks** or **one Skill Development Activity of 50 marks** to attain the COs and POs
3. The sum of two tests, two assignments/skill Development Activities, will be **scaled down to 50 marks**
4. **CIE methods /question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

Semester-End Examination:

1. The SEE question paper will be set for 100 marks and the marks scored will be proportionately reduced to 50.
2. The question paper will have ten full questions carrying equal marks.
3. Each full question is for 20 marks. There will be two full questions (with a maximum of four sub-questions) from each module.
4. Each full question will have a sub-question covering all the topics under a module.
5. The students will have to answer five full questions, selecting one full question from each module

Suggested Learning Resources:

Books

1. Bernd Bruegge and Allen H. Dutoit, "Object-Oriented Software Engineering: Using UML, Patterns and Java", Third Edition, Pearson Education, 2009
2. Roger S. Pressman, Object-Oriented Software Engineering: An Agile Unified Methodology, First Edition, Mc Graw-Hill International Edition, 2014.
3. Len Bass, Ingo Weber and Liming Zhu, —DevOps: A Software Architect's Perspective, Pearson Education, 2016.
4. Rajib Mall, —Fundamentals of Software Engineering, Third Edition, PHI Learning Private Limited, 2009
5. PankajJalote, —Software Engineering, A Precise Approach, Wiley India, 2010.

Skill Development Activities Suggested

- The students with the help of the course teacher can take up technical –activities which will enhance their skill or the students should interact with industry (small, medium and large), understand their problems or foresee what can be undertaken for study in the form of research/testing/projects, and for creative and innovative methods to solve the identified problem. The prepared report shall be evaluated for CIE marks.

Course outcome (Course Skill Set)

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

Sl. No.	Description	Blooms Level
CO1	Describe the fundamental software process models and the principles of requirement analysis.	L2
CO2	Implement design patterns and architectural styles to construct software systems.	L3
CO3	Utilize the software testing methodologies to ensure the quality and reliability of software.	L3
CO4	Analyze the integration of software project management practices with DevOps principles to improve software delivery and operational efficiency.	L4

Mapping of COS and POs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1				**X**				
CO2								X
CO3				**X**				
CO4								X

Semester- II

Web Application Development			
Course Code	MMC205	CIE Marks	50
Teaching Hours/Week (L:P: SDA/T)	2:1:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03

Course Learning Objectives:

1. To understand the fundamental concepts and technologies of web application development.
2. To gain proficiency in front-end and back-end web development.
3. To learn and apply modern web frameworks and libraries.
4. To develop skills in creating responsive and dynamic web applications.
5. To prepare students for industry roles requiring expertise in web development.

Module-1 08 Hours

Introduction to Web Development and HTML5:

Web Development Basics: Introduction to web technologies and protocols, Client-server architecture, Overview of front-end and back-end development

HTML5 Fundamentals: HTML5 elements and attributes, Semantic HTML5 tags, Forms and input types, Multimedia elements (audio, video)

Advanced HTML5: Canvas and SVG for graphics, HTML5 APIs(Geolocation, Web Storage, Web Workers), Offline web applications using AppCache.

Teaching Learning Process:

Lectures with PowerPoint presentations, Hands-on coding exercises in HTML5, Interactive discussions and problem-solving sessions, Assignments and quizzes for assessment.

Module-2 08 Hours

CSS3 and Responsive Web Design:

CSS3 Basics: Introduction to CSS3, Selectors, properties, and values, Box model, layout, and positioning, Flexbox and Grid layouts.

Responsive Web Design: Media queries, Responsive design principles, Fluid grids and flexible images, Mobile-first design approach.

CSS Frameworks: Introduction to Bootstrap, Bootstrap components and utilities, Customizing Bootstrap with Sass.

Teaching Learning Process:

Practical sessions on CSS3 and responsive design, Interactive coding exercises to implement responsive layouts, Group projects on developing responsive web pages, Continuous assessment through quizzes and assignments.

Module-3 08 Hours

JavaScript and DOM Manipulation:

JavaScript Basics: Introduction to JavaScript, Variables, data types, and operators, Control structures (if-else, loops), Functions and scope

Document Object Model (DOM): DOM structure and manipulation, Event handling and event listeners, Creating and modifying DOM elements, Form validation using JavaScript

Advanced JavaScript: Asynchronous JavaScript (callbacks, promises, async/await), AJAX and Fetch API, Introduction to JavaScript libraries (e.g., jQuery).

Teaching Learning Process:

Lab exercises on JavaScript and DOM manipulation, Practical coding sessions with real-time problem-solving, Group projects on creating interactive web applications, Continuous assessment through quizzes and coding challenges.

Module-4	08 Hours

Front-End Frameworks and AngularJS:

Introduction to Front-End Frameworks: Importance of front-end frameworks, Overview of popular frameworks (React, Angular, Vue)

AngularJS Basics: Introduction to AngularJS, Modules, controllers, and scope, Directives, expressions, and filters

Advanced AngularJS: Services and dependency injection, Routing and single-page applications (SPAs), Data binding and form handling, Custom directives and components.

Teaching Learning Process:

Practical sessions on AngularJS basics and advanced topics, Interactive coding exercises to build AngularJS applications, Group projects on developing single-page applications, Continuous assessment through quizzes and practical tests.

Module-5	08 Hours

Back-End Integration and Deployment:

Back-End Development: Introduction to server-side programming, Overview of server-side languages (Node.js, PHP, Python), RESTful web services and APIs, Database integration (SQL, NoSQL)

Full-Stack Development: Integrating front-end and back-end technologies, Developing full-stack web applications, Case studies on full-stack applications

Deployment and Security: Web application deployment (cloud platforms, hosting services), Security best practices for web applications, Authentication and authorization, Performance optimization.

Teaching Learning Process:

Lab exercises on back-end development and integration, Practical sessions on deploying web applications, Group discussions on web application security, Final project presentation and assessment.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. Minimum passing marks in SEE is 40% of the maximum marks of SEE. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 50% (50 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

1. Two Unit Tests each of **25 Marks**
2. Two assignments each of **25 Marks** or **one Skill Development Activity of 50 marks**
   a. to attain the COs and POs
3. The sum of two tests, two assignments/skill Development Activities, will be **scaled down to 50 marks**
4. **CIE methods /question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.**

Semester-End Examination:

1. The SEE question paper will be set for 100 marks and the marks scored will be proportionately reduced to 50.
2. The question paper will have ten full questions carrying equal marks.
3. Each full question is for 20 marks. There will be two full questions (with a maximum of four sub-questions) from each module.
4. Each full question will have a sub-question covering all the topics under a module.
5. The students will have to answer five full questions, selecting one full question from each module

Suggested Learning Resources:

Text Books:

1. Web Programming By Chris Bates , Wiley Publications
2. HTML5 Black Book by Dreamtech
3. Angular JS By Krishna Rungta
4. Bootstrap essentials by Snig by Packt-open source .

Skill Development Activities Suggested

- Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

The students with the help of the course teacher can take up activities which will enhance their activity based learning like Quizzes, Assignments and Seminars.

Course outcome (Course Skill Set)

At the end of the course the student will be able to:

Sl. No.	Description	Blooms Level
CO1	Demonstrate an understanding of web technologies and protocols.	L1

CO2	Develop and deploy web applications using HTML5, CSS3, JavaScript, and modern frameworks.	L1,L2
CO3	Apply responsive design principles using frameworks like Bootstrap.	L3
CO4	Implement dynamic web applications using AngularJS.	L4,L5
CO5	Integrate front-end and back-end technologies to create full-stack web applications.	L5

Industry-Relevant Key Points:

- Emphasis on coding standards and best practices.
- Integration of version control systems (e.g., Git) in project work.
- Exposure to industry-standard tools and frameworks.
- Real-world application development projects.
- Focus on collaborative development and agile methodologies.

Mapping of COS and POs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	**X**							
CO2								X
CO3			**X**					
CO4								X
CO5				**X**				

Semester- II

Object Oriented Programming using JAVA Laboratory			
Course Code	**MMCL206**	CIE Marks	50
Teaching Hours/Week (L:P: SDA/T)	0:2:0	SEE Marks	50
Credits	2	Exam Hours	03

Course objectives:

- Using java programming to develop programs for solving real-world problems.
- Reinforce the understanding of basic object-oriented programming concepts.

Sl.NO	Experiments
1	Write a Java program to print the following triangle of numbers1 1 2 1 2 3 1 2 3 4 1 2 3 4 5
2	Write a Java program to list the factorial of the numbers 1 to 10. To calculate the factorial value, use while loop. (Hint Fact of 4 = 4*3*2*1)
3	Write a Java program • To find the area and circumference of the circle by accepting the radius from the user. • To accept a number and find whether the number is Prime or not
4	Write a Java program to demonstrate a division by zero exception
5	Write a Java program to implement Inner class and demonstrate its Access protection.
6	Write a Java program to demonstrate Constructor Overloading and Method Overloading.
7	Write a JAVA program to demonstrate Inheritance. Simple Program on Java for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
8	Write a Java applet program, which handles keyboard event.
	Demonstration Experiments (For CIE) if any
9	Write a Java Program to create a window when we press ✓ M or m the window displays Good Morning ✓ A or a the window displays Good After Noon ✓ E or e the window displays Good Evening ✓ N or n the window displays Good Night
10	Write a Java program to implement a Queue using user defined Exception Handling (also make use of throw, throws). a. Complete the following: b. Create a package named shape. c. Create some classes in the package representing some common shapes like Square, Triangle, and Circle. d. Import and compile these classes in other program.

Course outcomes (Course Skill Set):

At the end of the course the student will be able to:

- Demonstrate the fundamental data types and constructs of Java Programming by writing executable/interpretable programs.
- Illustrate the object oriented principles with the help of java programs.
- Develop reusable and efficient applications using inheritance concepts of java.
- Learn the object oriented concepts and its implementation in Java.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 40%of maximum marks in the semester- end examination(SEE). In total of CIE and SEE student has to secure 50% maximum marks of the course.

Continuous Internal Evaluation (CIE):

CIE marks for the practical course is **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled downed to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.
- In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The average of 02 tests is scaled down to **20 marks** (40% of the maximum marks).

The Sum of **scaled-down** marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

SEE marks for the practical course is 50 Marks.

SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University.

All laboratory experiments are to be included for practical examination.

(Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.

Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.

Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)

Change of experiment is allowed only once and 10% Marks allotted to the procedure part to be made zero.

The duration of SEE is 03 hours

Semester- II

Data Structures with Algorithms Laboratory			
Course Code	**MMCL207**	CIE Marks	50
Teaching Hours/Week (L:P: SDA/T/T)	0:2:0	SEE Marks	50
Credits	2	Exam Hours	03

Course objectives:

- Evaluate the Expressions like postfix, prefix conversions.
- Implementing various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.

Sl.NO	Experiments
1	Implement a Program in C for converting an Infix Expression to Postfix Expression.
2	Design, develop, and execute a program in C to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide).
3	Design, develop, and execute a program in C to simulate the working of a queue of integers using an array. Provide the following operations: a. Insert b. Delete c. Display
4	Write a C program to simulate the working of a singly linked list providing the following operations: a. Display& Insert b. Delete from the beginning/end c. Delete a given element
5	Write a C program to Implement the following searching techniques a. Linear Search b. Binary Search.
6	Write a C program to implement the following sorting algorithms using user defined functions: a. Bubble sort (Ascending order) b. Selection sort (Descending order).
7	Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm (C programming)
8	From a given vertex in a weighted connected graph, find shortest paths to other vertices Using Dijkstra's algorithm (C programming)
	Demonstration Experiments (For CIE) if any
9	Using circular representation for a polynomial, design, develop, and execute a program in C to accept two polynomials, add them, and then print the resulting polynomial.
10	Design, develop, and execute a program in C to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide).

Course outcomes (Course Skill Set):

At the end of the course the student will be able to:

- Implement the techniques for evaluating the given expression.
- Implement sorting / searching techniques, and validate input/output for the given problem.
- Implement data structures (namely Stacks, Queues, Circular Queues, Linked Lists, and Trees), its operations and algorithms.
- Implement the algorithm to find whether the given graph is connected or not and conclude on the performance of the technique implemented.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 50% of the maximum marks. A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 40%of maximum marks in the semester-end examination(SEE). In total of CIE and SEE student has to secure 50% maximum marks of the course.

Continuous Internal Evaluation (CIE):

CIE marks for the practical course is 50 Marks.

- The split-up of CIE marks for record/ journal and test are in the ratio 60:40.
- Each experiment to be evaluated for conduction with observation sheet and record write- up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled downed to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.
- In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The average of 02 tests is scaled down to 20 marks (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

SEE marks for the practical course is 50 Marks.

SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University.

All laboratory experiments are to be included for practical examination.

(Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation rubrics shall be decided jointly by examiners.

Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.

Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks.

SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners) Change of experiment is allowed only once and 10% Marks allotted to the procedure part to be made zero. The duration of SEE is 03 hours