



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course: Artificial Intelligence and Machine Learning	Semester	II	
Course Code:	25MMC201	CIE Marks	50
Teaching Hours/Week (L:P:T)	3:2:0	SEE Marks	50
Total Hours of Pedagogy	40(Theory) +10 (Lab)	Total Marks	100
Credits	04	Exam Hours	03
Examination type (SEE)	Theory		

Course Objectives (Course Skill Set)

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

Module-1

Introduction to AI: What is AI? Foundations and History of AI Intelligent Agents: Agents and environment, Concept of Rationality, The nature of environment, The structure of agents.

Problem-solving: Problem-solving agents, Example problems.

Uninformed Search Strategies: Breadth First search, Depth First Search, Iterative deepening depth first search

Number of Hours:8

Module-2

Informed Search Strategies: Heuristic functions, Greedy best first search, A*search. Heuristic Functions **Logical Agents:** Knowledge-based agents, The Wumpus world, Logic, Propositional logic, Reasoning patterns in Propositional Logic

Number of Hours:8

Module-3

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.

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

Data Privacy: Introduction to Data Privacy, Data Privacy Issues in Machine Learning, Case Studies and Applications.

Number of Hours:8

Module-4

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.

Number of Hours:8

Module-5

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

Number of Hours:8

Sl. No	Experiments
1	Implement A* Search algorithm.
2	Implementation of TSP using heuristic approach.
3	Define a function which takes TWO objects representing complex numbers and returns new complex number with a addition of two complex numbers. Define a suitable class 'Complex' to represent the complex number. Develop a program to read N ($N \geq 2$) complex numbers and to compute the addition of N complex numbers.
4	Develop a program to print 10 most frequently appearing words in a text file. [Hint: Use dictionary with distinct words and their frequency of occurrences. Sort the dictionary in the reverse order of frequency and display dictionary slice of first 10 items]
5	Write a function named DivExp which takes TWO parameters a, b and returns a value c ($c=a/b$). Write suitable assertion for $a > 0$ in function DivExp and raise an exception for when $b=0$. Develop a suitable program which reads two values from the console and calls a function DivExp.
6	Write a program to compute summary statistics such as mean, median, mode, standard deviation and variance of the given different types of data.
7	Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions.
8	Write a program to demonstrate Regression analysis with residual plots on a given data set.
9	Write a program to implement k-Means clustering algorithm to cluster the set of data stored in .CSV file.
10	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

Course outcome (Course Skill Set)

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

CO1: Utilize AI intelligent agents and uninformed search strategies to develop solutions for a real-world scenario.

CO2: Make use of heuristic functions and logical agents to solve computational problems

CO3: Build an understanding of Machine Learning and various Data Pre-processing techniques.

CO4: Apply supervised learning technique to solve real world problems.

CO5: Apply unsupervised learning technique to solve real world applications

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year) Text

Books:

1. Advance Machine Learning with Python by John Hearty.
2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron (O'Reilly, 3rd Edition) – **Chapter sections on Responsible AI and Privacy-Aware ML.**
3. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning.
4. Artificial intelligence, A modern Approach , 2nded, Stuart Russel, Peter Norvig, PEA.

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. "Introduction to Machine Learning with Python".

Privacy-Preserving Machine Learning by Vishnu Kanhere & Jaydip Sen (Springer, 2021).

Web links and Video Lectures (e-Resources):

- <https://youtu.be/kqtD5dpm9C8?si=LBep4HWaMFRrAPsD>
- <https://youtu.be/4SJ7bEILPJK?si=5LurvjzUOuCew1W9>
- <https://youtu.be/7eh4d6sabA0?si=JWHNJRVS6NhQOPYe>

Teaching-Learning Process (Innovative Delivery Methods)

The following are sample strategies that educators may adopt to enhance the effectiveness of the teaching- learning process and facilitate the achievement of course outcomes.

1. Lectures with PowerPoint presentations, Interactive discussions and problem-solving sessions, Assignments and quizzes for assessment.
2. 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.
3. 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.
4. 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.
5. Lectures with PowerPoint presentations, Lab exercises on EDA and Python programs, Assignments and quizzes for 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.

CIE for the theory component of IPCC

1. Three Tests each of 25 Marks (scaled down to 25)
2. Two assignments each of 10 Marks/One Skill Development Activity of 20 marks (scaled down to 5 marks)
3. Total Marks of three tests and two assignments/one Skill Development Activity added will be CIE for 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 10 marks are for conducting the experiment and preparation of the laboratory record,
- 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.
- The Two laboratory test at the end /after completion of all the experiments shall be conducted for 50 marks. The test marks will be scaled down to 10 marks. Sum of Observation and Lab test will be the CIE marks for the laboratory component of IPCC for 20 marks.

SEE for IPCC

- The question paper will be set for 100 marks and marks scored will be scaled down proportionately to 50 marks.
- The question paper will have 2 Parts: Part A and Part B. In part B, each question is set for 16 marks.
- There will be 2 questions from each module in Part A. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
- The students have to answer 5 full questions in Part B, 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))



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course : Object Oriented Programming using Java	Semester	II	
Course Code	25MMC202	CIE Marks	50
Teaching Hours/Week (L:T:P)	4:0:0	SEE Marks	50
Total Hours of Pedagogy	50	Total Marks	100
Credits	04	Exam Hours	03
Examination type (SEE)	Theory		

Course Objectives (Course Skill Set)

- 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

Number of Hours:10

Module-2

Encapsulation- Introduction & examples, **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

Number of Hours:10

Module-3

Generics: About Generics, A simple Generic Example, General class with Two Type Parameters, General form of generic class Java.util: The Collections Framework: Collections Overview

The Collection Interfaces: List, Set, Queue, Deque. The Collection Classes: Array List, Linked List, HashSet, Linked HashSet.

Number of Hours:10

Module-4

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.

Number of Hours:10

Module-5

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.

Number of Hours:10

Course outcome (Course Skill Set)

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

CO1: Develop Java based solution to the given problem.

CO2: Apply inheritance, string handling and generic classes to build object-oriented programs.

CO3: Make use of collections to solve real world problems.

CO4: Build Java programs using packages, interfaces & exception handling.

CO5: Construct Java programs using the concept of multithreading.

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year) Text Books:

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
<https://www.classcentral.com/course/coursea-object-oriented-programming-in-java-4212>

Teaching-Learning Process (Innovative Delivery Methods)

The following are sample strategies that educators may adopt to enhance the effectiveness of the teaching- learning process and facilitate the achievement of course outcomes.

Lectures with PowerPoint presentations, Hands-on coding exercises using any IDE's

Interactive discussions and problem-solving sessions, Assignments and quizzes for 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 25 Marks (scale down to 25 Marks)
2. Two assignments each of 25 Marks or one Skill Development Activity of 50 marks to attain the COs and POs which will be scale down to 25 marks.

The sum of **three**-unit tests, two assignments/Skill Development Activities (CIE), will be 50 marks.

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 consists of Part A and Part B. Part A consists of 10 questions from 5 modules, each carrying 2 marks.
3. Part B consists of 10 questions. Each full question is for 16 marks. There will be two full questions (with a maximum of three 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



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course : Data Structure and Algorithms	Semester	II	
Course Code	25MMC203	CIE Marks	50
Teaching Hours/Week (L:P:T)	4:0:0	SEE Marks	50
Total Hours of Pedagogy	50	Total Marks	100
Credits	04	Exam Hours	03
Examination type (SEE)	Theory		

Course Objectives (Course Skill Set)

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

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.

Number of Hours:10

Module-2

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.

Number of Hours:10

Module-3

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, Floyd -Warshall algorithm.

Number of Hours:10

Module-4

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).

Number of Hours:10

Module-5

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.

Number of Hours:10

Course outcome (Course Skill Set)

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

CO1: Apply fundamental data structures to develop efficient algorithms.

CO2: Apply linear data structures to solve a given problem.

CO3: Apply Non- linear data structures to solve a given problem.

CO4: Develop algorithm for sorting and searching problems.

CO5: Apply advanced data structures and algorithms to solve complex problems.

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year) 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.
3. Algorithms, Kenneth A Berman and Jerome L Paul, Cengage Learning India Pvt Ltd, 2002 edition.

Reference books:

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

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/>

Teaching-Learning Process (Innovative Delivery Methods)

The following are sample strategies that educators may adopt to enhance the effectiveness of the teaching- learning process and facilitate the achievement of course outcomes.

1. Lectures with PowerPoint presentations, Hands-on coding exercises in C, Interactive discussions and problem-solving sessions, Assignments and quizzes for assessment.
2. Case studies and real-world examples, Practical coding sessions, Group discussions and problem-solving exercises, Mid-term project focusing on stack and queue applications.
3. 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.
4. 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.
5. 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 25 Marks (scale down to 25 Marks)
2. Two assignments each of 25 Marks or one Skill Development Activity of 50 marks to attain the COs and POs which will be scale down to 25 marks.

The sum of **three**-unit tests, two assignments/Skill Development Activities (CIE), will be 50 marks.

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 consists of Part A and Part B. Part A consists of 10 questions from 5 modules, each carrying 2 marks.
3. Part B consists of 10 questions. Each full question is for 16 marks. There will be two full questions (with a maximum of three 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



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course : Software Engineering		Semester	II
Course Code	25MMC204	CIE Marks	50
Teaching Hours/Week (L:P:T)	2:0:2	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		

Course Objectives (Course Skill Set)

- 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, Agile Methodologies: Scrum, Kanban, Domain engineering.

Number of Hours: 8

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.

Number of Hours: 8

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

Number of Hours: 8

Module-4

TESTING AND MAINTENANCE: Testing - Unit testing - Black box testing- White box testing - Integration and System testing-Regression testing - testing for specialized environment-testing GUI-testing client-server architectures-testing for documentation - Debugging - Program analysis - Symbolic execution - Model Checking Case Study.

Number of Hours: 8

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.

Number of Hours: 8

Course outcome (Course Skill Set)

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

CO1: Identify the principles of software process and Agile methodologies.

CO2: Determine the requirement specification for a given scenario.

CO3: Construct a software design for a given case study.

CO4: Utilize the software testing methodologies to evaluate the quality & reliability of software.

CO5: Utilize the principles of project management to build a solution for a given case study.

Suggested Learning Resources:

1. Bernd Bruegge and Allen H Dutoit, "Object-Oriented Software Engineering : using UML , patterns and Java"
2. Roger S Pressman, Object-Oriented Software Engineering: an Agile unified methodology, first edition, McGraw-Hill International edition, 2014.
3. Len Bass, Ingo Weber and Liming Zhu, DevOps: A software architect's perspective, Person Education 2016
4. Rajib Mall – Fundamentals of Software Engineering, 3rd edition, PHI learning private Limited, 2009.
5. Pankajjalote- Software Engineering, A Precise Approach, Wiley India, 2010

Teaching-Learning Process (Innovative Delivery Methods)

The following are sample strategies that educators may adopt to enhance the effectiveness of the teaching- learning process and facilitate the achievement of course outcomes.

Lectures with PowerPoint presentations, Interactive discussions and case study sessions, Assignments and quizzes for 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 25 Marks (scale down to 25 Marks)
2. Two assignments each of 25 Marks or one Skill Development Activity of 50 marks to attain the COs and POs which will be scale down to 25 marks.

The sum of **three**-unit tests, two assignments/Skill Development Activities (CIE), will be 50 marks.

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 consists of Part A and Part B. Part A consists of 10 questions from 5 modules, each carrying 2 marks.

3. Part B consists of 10 questions. Each full question is for 16 marks. There will be two full questions (with a maximum of three 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



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course : Python Programming		Semester	II
Course Code	25MMC205	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		

Course Objectives (Course Skill Set)

- Learn the syntax and semantics of the Python programming language.
- Illustrate the process of structuring the data using lists, tuples.
- Appraise the need for working with various documents like Excel, PDF, Word and Others.
- Demonstrate the use of built-in functions to navigate the file system.
- Implement the Object Oriented Programming concepts in Python

Module-1

Python Basics: Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables, You're First Program, Dissecting Your Program.

Flow control: Boolean Values, Comparison Operators, Boolean Operators, Mixing Boolean and Comparison Operators, Elements of Flow Control, Program Execution, Flow Control Statements, Importing Modules, Ending a Program Early with sys.exit()

Functions: def Statements with Parameters, Return Values and return Statements, The None Value, Keyword Arguments and print(), Local and Global Scope, The global Statement, Exception Handling, A Short Program: Guess the Number.

Number of Hours: 8

Module-2

Lists: The List Data Type, Working with Lists, Augmented Assignment Operators, Methods, Example Program: Magic 8 Ball with a List, List-like Types: Strings and Tuples, References.

Dictionaries and Structuring Data: The Dictionary Data Type, Pretty Printing, Using Data Structures to Model Real-World Things. **Fundamental Libraries:** NumPy, Pandas.

Number of Hours: 8

Module-3

Manipulating Strings: Working with Strings, Useful String Methods, Project: Password Locker, Project: Adding Bullets to Wiki Markup

Debugging: Raising Exceptions, Getting the Traceback as a String, Assertions, Logging, IDLE's Debugger

Number of Hours: 8

Module-4
<p>Organizing Files: The shutil Module, Walking a Directory Tree, Compressing Files with the zipfile Module, Project: Renaming Files with American-Style Dates to European-Style Dates, Project: Backing Up a Folder into a ZIP File.</p> <p>Reading and Writing Files: Files and File Paths, The os.path Module, The File Reading/Writing Process, Saving Variables with the shelve Module, Saving Variables with the print.format() Function, Project: Generating Random Quiz Files, Project: Multiclipboard</p> <p style="text-align: right;">Number of Hours: 8</p>
Module-5
<p>Classes and objects: Programmer-defined types, Attributes, Rectangles, Instances as return values, Objects are mutable, Copying.</p> <p>Classes and functions: Time, Pure functions, Modifiers, Prototyping versus planning.</p> <p>Classes and methods: Object-oriented features, Printing objects, Another example, A more complicated example, The init method, The <code>_str_</code> method</p> <p style="text-align: right;">Number of Hours: 8</p>
<p>Course outcome (Course Skill Set) At the end of the course, the student will be able to:</p> <p>CO1: Develop simple applications in python using basic constructs</p> <p>CO2: Design and implement applications using lists and dictionaries</p> <p>CO3: Apply debugging skills to identify and resolve errors in string manipulation programs.</p> <p>CO4: Build simple applications using files.</p> <p>CO5: Make use of object-oriented concepts to develop real-world applications.</p>
<p>Suggested Learning Resources:</p> <ol style="list-style-type: none"> 1. Al Sweigart, “Automate the Boring Stuff with Python”, 1st Edition, No Starch Press, 2015. (Available under CC-BY-NC-SA license at https://automatetheboringstuff.com/) (Chapters 1 to 18, except 12) for lambda functions use this link: https://www.learnbyexample.org/python-lambda-function 2. Allen B. Downey, “Think Python: How to Think like a Computer Scientist”, 2nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license at http://greenteapress.com/thinkpython2/thinkpython2.pdf (Chapters 13, 15, 16, 17, 18) (Download pdf/html files from the above link)
<p>Teaching-Learning Process (Innovative Delivery Methods) The following are sample strategies that educators may adopt to enhance the effectiveness of the teaching- learning process and facilitate the achievement of course outcomes. Lectures with PowerPoint presentations, Hands-on coding exercises, Assignments and quizzes for assessment</p>
<p>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.</p>

Continuous Internal Evaluation:

1. Three Unit Tests each of 25 Marks (scale down to 25 Marks)
2. Two assignments each of 25 Marks or one Skill Development Activity of 50 marks to attain the COs and POs which will be scale down to 25 marks.

The sum of **three**-unit tests, two assignments/Skill Development Activities (CIE), will be 50 marks.

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 consists of Part A and Part B. Part A consists of 10 questions from 5 modules, each carrying 2 marks.
3. Part B consists of 10 questions. Each full question is for 16 marks. There will be two full questions (with a maximum of three 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



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course : Object Oriented Programming using JAVA Laboratory	Semester	II	
Course Code	25MML206	CIE Marks	50
Teaching Hours/Week (L:P:T)	0:2:2	SEE Marks	50
Total Hours of Pedagogy	42	Total Marks	100
Credits	02	Exam Hours	03
Examination type (SEE)	LAB		

Course Objectives (Course Skill Set)

- 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 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)
2	Write a Java program <ul style="list-style-type: none">• 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
3	Write a Java program that demonstrates the concept of encapsulation by creating an Employee class with employee data such as name and designation, and retrieve their values using getter and setter methods.
4	Write a Java program to demonstrate Constructor Overloading and Method Overloading
5	Write a JAVA program to demonstrate Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
6	Write a Java program to demonstrate the use of the ArrayList class with generics by performing addition on an ArrayList of five integers.
7	Write a Java program to demonstrate the use of the linked list class with generics by creating a linked list that stores natural numbers and perform multiplication operation on the elements of the linked list
8	Write a Java program to demonstrate the use of the Stack class with generics by creating a stack that stores five numbers which are multiples of 5, and perform an addition operation on the elements of the stack.
9	Write a Java program to demonstrate a division by zero exception using Try, catch, throw and finally
10	1) Develop a Java program to create package named “tpack” and import and implement it in a suitable class. 2) Write a program to illustrate creation of threads using runnable class.

Course outcome (Course Skill Set)

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

CO1: Apply conditional looping statements to solve computational decision making problems.

CO2: Develop efficient applications using exception handling and packages.

CO3: Make use of concept of encapsulation and polymorphism to construct Java programs.

CO4: Apply the concept of multithreading to develop Java programs.

CO5: Develop efficient applications using the concept of Inheritance.

CO6: Make use of Linear Data structures to construct Java programs

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. 10 marks will be allocated for mini project.

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 down 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 50 marks each, 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 marks for the practical course is 100 marks scaled down to 50 Marks.

SEE shall be conducted jointly by the two examiners. (One Internal and One External)

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 for SEE is 03 hours.



K. S. INSTITUTE OF TECHNOLOGY

An Autonomous Institution under VTU, Approved by AICTE
Department of Master of Computer Applications
SECOND SEMESTER SYLLABUS

Course Data Structures with Algorithms Laboratory		Semester	II
Course Code	25MML207	CIE Marks	50
Teaching Hours/Week (L:P:T)	0:2:2	SEE Marks	50
Total Hours of Pedagogy	42	Total Marks	100
Credits	02	Exam Hours	03
Examination type (SEE)	Lab		

Course Objectives (Course Skill Set)

1. Evaluate the Expressions like postfix, prefix conversions.
2. 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	Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers . a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 b. Traverse the BST in Inorder, Preorder and Post Order c. Exit.
6	Write a C program to sort n numbers using merge sort.
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
9	Develop a Program in C for the following operations on Graph (G) of Cities a. Create a Graph of N cities using Adjacency Matrix. b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method
10	Implement 0/1 Knapsack problem using Dynamic Programming method

Course outcome (Course Skill Set)

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

CO1: Select an appropriate data structure to solve real world problems

CO2: Develop C programs using Linear data structures.

CO3: Make use of Non-Linear data structures to develop C programs.

CO4: Apply Sorting and Searching techniques to construct C programs

CO5: Apply Dynamic Programming and Greedy techniques to solve real world problems.

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. 10 marks will be allocated for mini project.

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 down 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 50 marks each, 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 marks for the practical course is 100 marks scaled down to 50 Marks.

SEE shall be conducted jointly by the two examiners. (One Internal and One External)

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 for SEE is 03 hours.