Detailed Syllabus

Lecture-wise Breakup

Subject Code	17M11CS111	**Semester (specify Odd/Even)**	**Semester** Odd **Session 2024-2025 Month from July 24 to December 24**
Subject Name	Data Structure & Algorithms for Big Data		
Credits	**3**	**Contact Hours**	**3(L)**

Faculty (Names)	**Coordinator(s)**	Shikha Jain
	Teacher(s) (Alphabetically)	Shikha Jain

COURSE OUTCOMES		**COGNITIVE LEVELS**
C110.1	Understand the importance of data structure and algorithm for Big Data	Understand Level (Level 2)
C110.2	Apply appropriate data structure for the big data problems.	Apply Level (Level 3)
C110.3	Analyze various algorithms required to solve problems from the domain of big data.	Analyze Level (Level 4)
C110.4	Design and evaluate an efficient solution to a given real world problem using Big data based data structures and algorithms	Create Level (Level 6)

S.N.	**Subtitle of the Module**	**Topics in the module**	**No. of Lectures for the module**	**Remarks**
1.	Introduction to Big Data	Big Data and its characteristics, Type of data, Motivation, Applications of Big Data, Domains for Big Data, Various tools and services	2	
2.	Basic Data Structures Concepts	Array: searching, sorting; Trees: Binary Tree, AVL, B-tree; Graph: BFS, Spanning Tree	3	
3.	Parallel Basic Algorithms	Brent's Theorem, Sum of n numbers, Prefix scan, Pointer Jumping, Rank of list, Pointer to root, Suffix sum, Preorder traversal of binary tree.	4	
4.	Parallel advance Algorithms	Parallel Sorting (Merge Sort, Quick Sort, Odd even transposition sort), Parallel shortest Path Algorithm, Parallel Matrix Algorithms	5	

5.	Indexing strategies Trees	R and R+ Trees, Prefix Trees, LSM trees	5	
6.	Big Data Databases	MongoDB, Accumulo, BigTable	5	
7.	Map Reduce	MapReduce, Mapreduce Job scheduling	4	
8.	Hash and membership	Hashing, Approximate Membership, Bloom Filter, Counting Bloom Filter	5	
9.	Cardinality and Frequency	LogLog, HyperLogLog, Count Sketch, Count–2 min sketch	5	
10.	Big Data Framework	Hadoop HDFS, Read and write operation, Fault Tolerance-Failures and Recovery	4	
Total number of Lectures			**42**	

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25Attendance (10 Marks), Assignment/Quiz/Mini-project (15 Marks)
Total	**100**

Project based learning: Students in group of 3 to 4 students are required to develop mini-project based on the concepts taught in this course. In mini-project, students need to create the distributed environment either using Hadoop framework/Multithreading using OpenMP/ Matlab. Problem statements need to be formulated in various applications domains of big data, proposing the solution approach and implemented over the created distributed environment.

Text Books	
1.	Algorithms and Data Structures for Massive Datasets by Dzejla Medjedovic, Emin Tahirovic, and Ines Dedovic, MEAP began July 2020
2.	Data Algorithms: Recipes for Scaling Up with Hadoop and Spark by Mahmoud Parsian, O'Reilly Media, 2015
Reference Books	
1.	Probabilistic Data Structures and Algorithms in Big Data Applications by Andrii Gakhov, 2022
2.	Sequential and Parallel Algorithms and Data Structures by Roman Dementiev, Martin Dietzfelbinger, Peter Sanders, Kurt Mehlhorn, 2019
3.	Big Data with Hadoop MapReduce A Classroom Approach By Rathinaraja Jeyaraj, Ganeshkumar Pugalendhi, Anand Paul, 2021

COs	PO1	PO2	PO3	PSO 1	PSO2
C110.1	1 Students will understand the existing algorithms to solve various open problems in the domain.		2 Towards the end of the semester, students will submit a mini-project taken from the domain of Big Data		
C110.2	2 Students will design algorithms to solve various open problems in the domain.	1 Students will submit a mini project report	2 Towards the end of the semester, students will submit a mini-project taken from the domain of Big Data	2 Various real-world problems in the domain will be discussed and given in assignments/exam	
C110.3	2 Students will design algorithms to solve various open problems in the domain.	1 Students will submit a mini project report	2 Towards the end of the semester, students will submit a mini-project taken from the domain of Big Data	2 Various real-world problems in the domain will be discussed and given in assignments/exam	
C110.4	2 Students will design algorithms to solve various open problems in the domain.	1 Students will submit a mini project report	2 Towards the end of the semester, students will submit a mini-project taken from the domain of Big Data	2 Various real-world problems in the domain will be discussed and given in assignments/exam	1 Students will work on mini project to provide ethical solution to the real world problem
Avg.	2	1	2	2	1

Detailed Syllabus
Lecture-wise
Breakup

Subject Code	17M11CS112	**Semester (specify Odd/ Even):** Odd	**Semester:** Odd **Session** 2024-2025 **Month from** July to December
Subject Name	Machine Learning and Data Mining		
Credits	3	**Contact Hours**	3

Faculty (Names)	**Coordinator(s)**	Anita Sahoo
	Teacher(s)	Anita Sahoo

COURSE OUTCOMES		**COGNITIVE LEVELS**
C112.1	Explain different techniques used in machine learning and data mining.	Level-2- (Understanding)
C112.2	Identify and apply a suitable technique to solve the given problem in the domain of data mining and machine learning.	Level-3 (Apply)
C112.3	Derive implications by applying pre-processing techniques on datasets for machine learning problems.	Level-4 (Analyze)
C112.4	Solve to provide the complete solution to a given knowledge discovery/ prediction problem and evaluate its performance using suitable metric(s).	Level-5 (Evaluate)

Module No.	**Subtitle of the Module**	**Topics in the module**	**No. of Lectures for the module**
1	Introduction	Introduction to Machine Learning, Data Mining and Knowledge Discovery in Databases, Data Types, EDA	4
2	Classification	Introduction to classification, k-Nearest Neighbours, Naïve Bayes, Decision Trees, Support Vector Machine, Back-propagation Neural Network	8
3	Regression	Linear Regression with One Variable, Linear Regression with Multiple Variables, Logistic Regression	4
4.	Clustering	Introduction, Different type of Clustering Methods, Partitioning Clustering Methods, Hierarchical Clustering Methods, k-means, k-medoids, density based clustering, Self-Organizing Map, cluster validation	6
5.	Association Rules	Support, Confidence, Lift, Conviction; Apriori algorithm, Eclat algorithm, FP-growth algorithm	5
6.	Dimensionality Reduction	Introduction, Subset Selection, PCA, SVD, Factor Analysis, Multidimensional Scaling, Linear Discriminant Analysis	4
7.	Ensemble Methods	Ensemble methods of classification-Bagging, Boosting, and Random Forest	4
8.	Quantum Machine Learning	Fundamentals of quantum computing, quantum states, quantum gates, interference, superposition, entanglement, measurements, variational quantum circuit using Qiskit	7
		Total number of Lectures	42

Evaluation Criteria	
Components	**Maximum**
Marks T1	**20**
T2	**20**
End Semester Examination	**35**
TA	**25 (Attendance (10), Mini-project/Assignment (15))**
Total	**100**
Project based learning: Each student in a group of 3-4 will have to develop a mini project based on association mining, classification and clustering approaches. The students can choose any real-world application that requires some decision-making. The students have to implement the mini-project using any open-source programming language. Project development will enhance the knowledge and employability of the students in IT sector.	
Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc.)	
1	Han, Jiawei, Jian Pei, and Micheline Kamber. Data mining: concepts and techniques. Elsevier, 3rd edition ,2012
2	Kimball R. and Ross M ,The Data Warehouse Toolkit", Wiley, 3rd edition,2013
3	Pujari, Arun K, Data mining techniques , Universities press, 3rd edition , 2013
4	Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining, second edition, 2019
5	Soumen Chakrabarti, Mining the Web: Discovering knowledge from hypertext data", Morgan Kaufmann, Elsevier
6	Mitchell, Tom, and Machine Learning McGraw-Hill. "Edition." (1997).
7	Wittek, Peter. Quantum machine learning: what quantum computing means to data mining. Academic Press, 2014.
8	Anahory S. and Murray D, Data Warehousing in the Real World, Addison- Wesley
9	Dunham, Margaret H. Data mining: Introductory and advanced topics. Pearson Education India, 2006.
10	Mattison R. ,Web Warehousing and Knowledge Management", Tata McGraw- Hill.
11	David Hand, Heikki Mannila and Padhraic Smyth ,Principles of Data Mining,PHI
12	Transactions on Database Systems (ACM)
13	IEEE Transactions on Knowledge & Data Engineering
14	The VLDB Journal The International Journal on Very Large Data Bases

Machine Learning and Data Mining Lab (17M15CS112)
Detailed Syllabus

Course Code	17M15CS112	**Semester:** Odd 2024	**Semester: I Session** 2024 -2025 Month from: July – December 2024
Course Name	Machine Learning and Data Mining Lab		
Credits	1	**Contact Hours**	**2**

Faculty (Names)	**Coordinator(s)**	Dr. Amit Mishra
	Teacher(s) (Alphabetically)	Dr. Anita Sahoo, Dr. Amit Mishra

COURSE OUTCOMES: Students will be able to		**COGNITIVE LEVELS**
C173.1	Perform data preprocessing, data sampling and visualization.	Understanding (Level-2)
C173.2	Apply Linear regression, Logistic regression, kNN, k Means, SVM and ID3 on different datasets.	Apply (Level-3)
C173.3	Implement Apriori algorithm and Eclat algorithm in R.	Apply (Level-3)
C173.4	Apply neural networks such as ANN, BPN and CNN to different datasets.	Apply (Level-3)
C173.5	Evaluate and analyze different machine learning models on the basis of their performances.	Evaluate (Level-5)

Module No.	**Title of the Module**	**List of Experiments**	**CO**
1.	Python for data sampling and Visualization	a. To write a program for writing the pixel values of an image b. Write programs for Data Sampling (given dataset).	1
2.	Python for text processing	Use IPython (a web version provided by Jupyter notebook) to write a word count program. Your program should read a text document (download from https://raw.githubusercontent.com/python/cpython/master/	1
3.	Classification-1	Implement kNN algorithm using Python. Consider the iris dataset and report the accuracy of classification. [May take help from : https://machinelearningmastery.com/tutorial-to-implement-k-nearest-neighbors-in-python-from-scratch/]	2
4.	Clustering	Clustering: Implement kMeans Algorithm	2
5.	Classification-2	Classify the wine dataset of UCI Repository by ID3.	2
6.	Data Mining-1	Implement Logistic Regression on a sample dataset	2
7.	Data Mining-2	Implement apriori and Eclat algorithm for association rule mining in R	3
8.	SVM-1	Apply Support Vector Machine on the dataset of question the Parkinson dataset given in https://archive.ics.uci.edu/ml/datasets/Parkinson+Dataset+with+replicated+acoustic+features+.	2
9.	Comparison of Classification algorithms	Compare the classification of Iris dataset by different algorithms namely kNN, ID3 and SVM. Report accuracy and other performance measures.Implement neural networks for Classification of *four* character patterns	5

10.	ANN	Apply Multi Layer Percepron for supervised learning (problem statement to be given individually)	4
11.	BPN	Use back propagation for supervised learning . For the data based on 1990 census data from California.Evaluate the accuracy of a model's predictions using RMSE.	4
12.	CNN	Implement CNN using TensorFlow for classifying MNIST images	4

Evaluation Criteria

Components	**Maximum Marks**
Lab Test1	20
Lab Test2	20
D2D	50
Attendance	10
Total	100

PBL- Students in a group of 4-5 will be designing an efficient solution to a given problem / case-studies using appropriate Machine Learning and Data mining Technique studies in the course.

Recommended Reading material:

Text Books:

1.	Jiawei, Han, and Kamber Micheline. *Data mining: concepts and techniques*. Morgan kaufmann, 2006.
2.	Chakrabarti, Soumen, Richard E. Neapolitan, Dorian Pyle, Mamdouh Refaat, Markus Schneider, Toby J. Teorey, Ian H. Witten et al. *Data mining: know it all*. Morgan Kaufmann, 2008.
3.	Trueblood, Robert P., and John N. Lovett. *Data mining and statistical analysis using SQL*. Vol. 1. Berkeley, CA: Apress, 2001.
4.	Kimball, Ralph, and Margy Ross. The data warehouse toolkit: the complete guide to dimensional modeling. John Wiley & Sons, 2011.
5.	Pujari, Arun K. Data mining techniques. Universities press, 2001.

Reference Books:

1.	Mining, What Is Data. *Introduction to data mining*. New Jersey: Pearson Education, Inc, 2006.
2.	Chakrabarti, Soumen. *Mining the Web: Discovering knowledge from hypertext data*. Morgan Kaufmann, 2002.
3.	Berson, Alex, and Stephen J. Smith. *Data warehousing, data mining, and OLAP*. McGraw-Hill, Inc., 1997.
4.	Inmon, William H. Building the data warehouse. John wiley & sons, 2005.
5.	Anahory, Sam, and Dennis Murray. Data warehousing in the real world: a practical guide for building decision support systems. Addison-Wesley Longman Publishing Co., Inc., 1997.
6.	Dunham, Margaret H. *Data mining: Introductory and advanced topics*. Pearson Education India, 2006.
7.	Subasi, Abdulhamit. Practical machine learning for data analysis using python. Academic Press, 2020.
8.	Putatunda, Sayan. *Practical Machine Learning for Streaming Data with Python*. Berkeley, CA: Apress, 2021.

Detailed Syllabus
Lab-wise Breakup

NOTE: All the entries (…) must be in Times New Roman 11.

Course Code	17M15CS113	**Semester** **Odd 2024**	**Semester** ... **Session** 2024-25 **Month from** July to Dec, 2024
Course Name	**Cloud Technology Lab**		
Credits	1	**Contact Hours**	2 Hours

Faculty (Names)	**Coordinator(s)**	Dr Prakash Kumar
	Teacher(s) (Alphabetically)	Dr. Prakash Kumar Mr. Prashant Kaushik

COURSE OUTCOMES		**COGNITIVE LEVELS**
C171.1	Demonstrate the architecture and layers of Cloud Service Models, Deployment models etc.	Understand (level 2)
C171.2	Provisioning of Data Centers, Virtual Machines (VMs) and cloudlet allocations on CloudSim using various scheduling algorithms.	Apply (level 3)
C171.3	Analyze various Scheduling techniques and resource allocations, compare their performances on different Cloud Platforms, like, CloudSim, Amazon Web Services (AWS).	Analyze (level 4)
C171.4	Evaluate the various Cloud Services provisioning and their performances using AWS platforms, Containers and Dockers.	Evaluate (level 5)

Module No.	**Title of the Module**	**List of Experiments**	**CO**
1.	CloudSim installations, VM creation and usage	Understand the Cloud Service Models, Deployment Models, Various Cloud Layers, Data Centers, Virtualization Technology, Virtual Machines (VMs), Virtual Machine Monitors (VMMs).	CO1
2.		Provisioning of Data Centers, Virtual Machines (VMs) on CloudSim. Allocate different Cloudlets to VMs and Data Centers using different scheduling algorithms.	CO2
3.	Analyze various Scheduling algorithms in different scenarios on cloudsim, AWS	Create different Data Centers and allocate the VMs to them and analyze the outcomes	CO3
4.		Analyze various Scheduling techniques and resource allocations supported by Cloud Platforms, e.g. CloudSim and AWS., Their performance evaluations on different Cloud Platforms, like, CloudSim and Amazon Web Services (AWS).	CO3
5.	Evaluate Cloud Service provision on AWS, Containers and Dockers.	Evaluate the various Cloud Services provisioning and their performance evaluations using AWS like EC2, RDS, Simple Storage Service, Containers and Dockers.	CO4
***n*.**	…	…	…

Evaluation Criteria

Components	**Maximum Marks**
Lab Test# 1	20
Lab Test# 2	20
D2D work	60 (D2D: 30 marks, PBL: 20 marks, Attendance: 10 marks)
Total	**100**

Project Based Learning: A group of maximum 2 students are to be formed. Each group shall choose a Cloud based project. The project shall be designed and/or modeled based on Cloud Platform like AWS, Google cloud, Eucalyptus, CloudSim, iFogSim or any other Cloud Platform, preferably open source platforms and tools. The project shall function and run as per the objective of the project. Live demonstration of the project shall be shown during their presentation. The project evaluation shall be done based on the quality, innovation, relevance and creativity involved.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)	
1.	K. Hwang, Geoffrey C. Fox, Jack J. Dongarra, "Distributed and Cloud Computing- From Parallel Processing to the Internet of Things", Morgan Kauffman Publishers, Elsevier.
2	George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" O'REILLY publication.
3	"Virtualization Overview", White paper, VM Ware.
4.	Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar A. F. De Rose, and Rajkumar Buyya, CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms, Software: Practice and Experience, Volume 41, Number 1, Pages: 23-50, ISSN: 0038-0644, Wiley Press, New York, USA, January 2011.
5.	Tom Guérout, Thierry Monteil, Georges Da Costa, Rodrigo Neves Calheiros, Rajkumar Buyya, Mihai Alexandru, Energy-aware Simulation with DVFS, Simulation Modelling Practice and Theory, Volume 39, No. 1, Pages: 76-91, ISSN: 1569-190X, Elsevier Science, Amsterdam, The Netherlands, November 2013.
6.	Rajkumar Buyya, Rajiv Ranjan and Rodrigo N. Calheiros, Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities, Proceedings of the 7th High Performance Computing and Simulation Conference (HPCS 2009, ISBN: 978-1-4244-4907-1, IEEE Press, New York, USA), Leipzig, Germany, June 21 - 24, 2009. - **Keynote Paper.**
7.	https://www.docker.com
***m*.**	…

COs	PO 1	PO 2	PO 3	PSO1	PSO2
C171.1	2	2	2	1	1
C171.2	2	2	2	1	2
C171.3	2	1	1	1	1
C171.4	1	1	2	2	2
AVG.	2	2	2	1	2

1. CO-PO and CO-PSO Mapping (M. Tech- CSE) I sem:

COs	PO 1	PO 2	PO 3	PSO1	PSO2
C171.1	2 Basics of Cloud services demonstrated	2 Basic principles and architectures of Cloud model is demonstrated	2 Various Cloud Service types and deployment models are demonstrated	1 Role of Virtualization Technology in Cloud Model is demonstrated	1 Data Center, Virtual Machine creation and usage is demonstrated
C171.2	2 Provisioning of Data Center and VMs on CloudSim platform	2 Allocation of Virtual Machines to Data Centers and Hosts	2 Applying various scheduling algorithms for VM provisioning and cloudlet allocations	1 Allocate Cloudlets to VM and Data Centers	2 Applying various scheduling algorithms for Cloudlet allocations on VMs
C171.3	2 Creating VMs and Instances on Amazon Web Services (AWS)	1 Analysing the behaviour of scheduling techniques	1 Analysis of instances on AWS, Elastic Compute Cloud (EC2) etc.	1 Analysis of Simple Storage Service (S3)	1 Analysis of other AWS Services, viz, Relational Database Service (RDS).
C171.4	1 Evaluation of AWS, Elastic Compute Cloud (EC2) features	1 Evaluation of AWS storages and their features, namely, Simple Storage Service (S3), Relations Database Services (RDS)	2 Performance evaluations of instances on AWS, EC2, storage and other services.	2 Performance Evaluation of Containers and their benefits over Virtual Machines.	2 Performance Evaluation of Dockers and their applications.
AVG.	2	2	2	1	2

Detailed Syllabus
Lecture-wise Breakup

Course Code	17M17CS121	**Semester** Odd **(specify Odd/Even)**	**Semester** 3rd Session2024 -2025 **Month** from July to Dec
Course Name	Project Based Learning-II		
Credits	4	**Contact Hours**	0-0-8
Faculty (Names)	**Coordinator(s)**	Dr. Kashav Ajmera	
	Teacher(s) (Alphabetically)	Dr. Kashav Ajmera, Dr. Shikha Jain	

COURSE OUTCOMES At the completion of the course, Students will be able to		**COGNITIVE LEVELS**
C210.1	Identify live problems that would be solved through automated software development process.	Apply Level (C3)
C210.2	Confront the issues related to development of project which includes team work, test driven design, data collections, implementations etc.	Apply Level (C3)
C210.3	Develop oral communication skill and prepare a technical report	Apply Level (C3)
C210.4	Critically review the projects and can skilfully map each stage in software development cycle.	Apply Level (C3)

SN	**Activity**	**Details**	**No. of labs**
1	Group Allocationa nd Literature Survey	a) 3 – 6 students in a batch and a maximum of 5 – 6 batches b) average CGPA of the batches should be roughly same	**2**

2	Literature Survey&Problem Identification	a) Automation Problems (live problem relevant to the Indian society) b) Economic considerations c) Aim d) Scope e) Open Source Automation Building & Testing Tools: E.g.: JUnit is an open source unit testing tool for Java programming language	**2**
3	Reviews-1		**2**
4	Problem Formulation and Gantt Chart	a) Design and Implementation Constraints b) Assumptions and Dependencies c) Functional Requirements d) Non-functional Requirements	**2**
5	Lab Class	Implementation, Testing and Analysis	**2**
6	Lab Class	Implementation, Testing and Analysis	**2**
7	Mid Term Viva	a) Presentation by Students b) Viva	**2**
8	Lab Class	Implementation, Testing and Analysis	**2**
9	Reviews -4		**2**
10	Lab Class	Implementation, Testing and Analysis	**2**
11	Reviews -5		**2**
12	Lab Class	Testing, Analysis, and Report Preparation	**2**
13	Reviews -6		**2**
14	End Term	a) Presentation by Students b) Viva c) Report Submission d) Self-Assessment Report Submission e) Peer Evaluation	**2**

Total number of labs	**28**

Evaluation Scheme:

Parameters	Marks
6-Reviews (8 Marks each)	48
Report	10
Presentation	10
Viva	16
Peer Assessment	8
Self-Assessment	8

Total Marks	**100**

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc.)	
Text Book	
1.	…
References	
1.	…

CO-PO Mapping:

COs	PO1	PO2	PO3	PSO1	PSO2
C210.1	2 Study literature to identify live problem to pursue	2 Based on study list out functional and non-functional requirements, and include in technical reports	1 Study existing literature and gain expertise to address the identified problem	2 Develop research aptitude to address the identified problem	2 Learn art of picking prominent societal problems which needs to be addressed urgently
C210.2	3 Investigate approaches to solve project problems and make effective plan for project development	2 Prepare technical reports for stage wise project development	3 Use team's diverse knowledge to solve project problems	3 Work on identified issues in project development using existing technologies and resources	2 Use ethical principles for project development
C210.3	0	3 Learn ways to write a technical documents and skills to present work	2` Conduct team meetings and identify skillset for the project development	0	2 Learn ways to cite other's work in technical report
C210.4	2 Identify existing solutions to project problem and inspect scope of improvements in them	1 Learn ways to comprehend existing SDF models	2 Review the existing tools and technology and identify suitable one	2 Visualize and analyze the existing solutions to given problem	2 Critically review existing literature and identify art of claiming work for contribution
Avg.	1.7	2	2	1.7	2

ORDINANCE

3.3A **Project Based Learning**

(a) In PBL (Project Based Learning) Courses, students will learn a new subject through execution of project(s).

(b) Students will be divided into batches ranging from **3 – 6 students in a batch** and a **maximum of 5 – 6 batches**for the whole class. The students in batches will be decided by the instructor. Choice of batch formation shall not be given to the students. The **average CGPA of the batches should be roughly**

samemeaning thereby that each batch will consist of students with high average and low CGPA.

(c) The projects to be given shall be decided by the instructor in such a manner that it involves gaining knowledge of the subject and additionally forces students to demonstrate skill acquisition at least in the following areas:

(i) Problem solving
(ii) Team working
(iii) Communication skills (both oral and written)
(iv) Economic considerations
(v) Acquisition of knowledge in allied areas as required by the Project

The Project should preferably be a **live problem relevant to Indian society**.

(d) The instructor shall help the students in developing the project by giving hints and suggestions, but normally should refrain from giving ready-madesolutions. If need be, the instructor may deliver short lectures.

(e) In order to force the students to work consistently throughout the semester, an assessment-cum-assistance session should be carried out on a fortnightly basis or more frequently, if felt necessary by the instructor.

(f) The evaluation scheme for Project Based Learning courses shall be as under:

(i) Each fortnightly assessment - 8%
(First assessment should be at the end of 3rd week from the beginning of the semester and thereafter fortnightly assessment. A total of six assessments giving a total percentage 6 x 8 = 48%) - 48%
(ii) Report at the end of the semester - 10%
(iii) Semester end presentation by the students - 10%
(iv) Viva-voce at the end of the semester - 16%
(v) Peer group evaluation (i.e. evaluation by the fellow - 8% students not belonging to the same batch)
(vi) Self-assessment by the student concerned (can be - 8% moderated by the instructor by discussing with the student concerned)

RUBRICS for Evaluation

Assessment-1	Exemplary (>=80%)	Competent (>=50% &<80%)	Unsatisfactory (<50%)

Literature Survey	Insightful and in-depth background information is provided to illuminate the issues through inclusion of history relevant to the presentation, the "big picture" and a succinct description of the significance of the project.	Background information is provided, including references to the work of others and an explanation of why the project was undertaken, to help put the presentation in context.	Little or no background information is presented to help the audience understand the history and significance of the project.
Problem Identification	The problem has been shown (not just stated) to exist with supporting factual evidence.	The problem has stated but has weak support.	Problem has not been stated clearly and lacks supporting evidence.

Assessment-2	**Exemplary (>=80%)**	**Competent (>=50% &<80%)**	**Unsatisfactory (<50%)**
Literature Survey	Existing solutions to the problem, including their good and bad points, have been stated.	Existing solutions have been stated. Additional discussion may be warranted in places.	Connection between references and what is written is not clear. Little investigation has been done.
Problem Formulation	The project's objectives are clearly stated. Motivation for pursuing the project and its relevance are clearly established. There are clear expectations of the specific outputs or deliverables for the project. A set of measurable performance requirements has been created.	The project's objectives are presented. The motivation for pursuing the project and its relevance are addressed. Expectations have been stated. Some objectives may not be measurable.	The project's objectives are missing or incomplete. There is little or no discussion of motivation or relevance. Expectations have been stated but needs clarity. Most objectives are not measurable.
Gantt Chart	A plan stating the completion date, and required resources has been presented. Gantt chart has been generated.	Some aspects of the plan have not been fully developed.	Lack of planning is evident.

Assessment-3	**Exemplary (>=80%)**	**Competent (>=50% & <80%)**	**Unsatisfactory (<50%)**

Methodology	A system block diagram has been developed to assist the team in solving the design. All blocks have been broken down to a manageable level. ***For web/ mobile applications:*** Pages are attractive and consistent in style throughout the site. Site is well organized and is easily navigated from any page. Graphic elements are appropriate, of high quality, and are creatively used to enhance content.	A system block diagram has been developed to assist the team in solving the design. Not all blocks have been broken down to a manageable level. ***For web/ mobile applications:*** Pages are attractive, but not consistent in style throughout the site. Site is well organized. Graphic elements are appropriate and are of acceptable quality to enhance content.	A system block diagram has not been fully developed. Problem has not been broken down to manageable tasks and blocks. ***For web/ mobile applications:*** Pages are unattractive Site is not organized or consists of a single page. Graphic elements are not appropriate or not used, or are of such poor quality that they detract from content.
Coding/ Implementation	All major points of the project were completed as per planning.	Most points of the project were completed as per planning.	Little or none of the project was completed as per planning.

Assessment-4	**Exemplary (>=80%)**	**Competent (>=50% &<80%)**	**Unsatisfactory (<50%)**
Coding/ Implementation	All major points of the project were completed as per planning.	Most points of the project were completed as per planning.	Little or none of the project was completed as per planning.

Assessment-5	**Exemplary (>=80%)**	**Competent (>=50% &<80%)**	**Unsatisfactory (<50%)**
Coding/ Implementation	All major points of the project were completed as per planning.	Most points of the project were completed as per planning.	Little or none of the project was completed as per planning.
Presentation	Clearly heard and polished. Attitude indicates confidence and enthusiasm and audience attention is constantly maintained. Presenters demonstrate full knowledge of the material and can explain and elaborate on expected questions.	Clearly heard but not polished. Attitude indicates confidence but not enthusiasm and audience attention aremostly maintained. Presenters have sufficient knowledge of the material to answer expected questions.	Difficult to hear and/or moments of awkwardness. Attitude indicates some lack of confidence and/or disinterest in subject and audience attention is minimally maintained. Presenters cannot answer expected questions.
Peer Evaluation	To greatest extent	To great extent	To some extent or no contribution

Assessment-6	**Exemplary (>=80%)**	**Competent (>=50% &<80%)**	**Unsatisfactory (<50%)**

Coding/ Implementation	All major points of the project were completed as per planning.	Most points of the project were completed as per planning.	Little or none of the project was completed as per planning.

End Term Assessment	**Exemplary (>=80%)**	**Competent (>=50% &<80%)**	**Unsatisfactory (<50%)**
Viva	Answers the questions to greatest extent	Answers the questions to a great extent	Answers the questions to some extent
Report	Addresses all specified content areas. Material abundantly supports the topic. All items are labelled in accordance with engineering standards and are referred to in the text. Prior work is acknowledged by referring to sources for theories, assumptions, quotations, and findings. References are in IEEE format.	Addresses most of the specified content areas. Material minimally supports the topic. Use of engineering terms and jargonwith some minor exceptions, references are in IEEE format.	Addresses few of the content areas. Material does not support the topic. There is no corresponding explanatory text for included items. Little attempt is made to acknowledge the work of others. Most references that are included are inaccurate or unclear.
Presentation	Clearly heard and polished. Attitude indicates confidence and enthusiasm and audience attention is constantly maintained. Presenters demonstrate full knowledge of the material and can explain and elaborate on expected questions.	Clearly heard but not polished. Attitude indicates confidence but not enthusiasm and audience attention aremostly maintained. Presenters have sufficient knowledge of the material to answer expected questions.	Difficult to hear and/or moments of awkwardness. Attitude indicates some lack of confidence and/or disinterest in subject and audience attention is minimally maintained. Presenters cannot answer expected questions.
Peer Evaluation	To greatest extent.	To great extent.	To some extent or no contribution.

Software development automation

The automated software development process is characterized by the following characteristics:

1. **A single common code repository** is put in place. All developers place the code they write in the repository. Currently, Git is the most popular version control system. The code in the repository is the sole source of software in the project.
2. **There is the so-called "build process"** in place. The build process is a standardized method for creatingand building subsequent software copies. Every developer, tester, testing script and mechanism uses the exact same process.

3. **The build process is automated**. Obtaining the current version of the software does not require anybody to perform a large number of manual actions. In an ideal situation, the build process is another script or a piece of software, which is also versioned in the code repository. A developer downloads the latest code from the repository, starts the build process (for example by starting a script) and obtains the current state of the application. The same script should be used by all the testing tools and testing environments, as well as for building demo versions.
4. **The build process is fast**. Building the software package does not last too long. This allows for testing results and implementing fixes multiple times.
5. The team commits changes often, every day or several times per day at best. The working code is pushed to the master branch in the version control system on an ongoing basis.
6. **The testing environment should resemble the production** environment as closely as possible. In an ideal situation, it would be a direct copy of a production environment.
7. **The process of pushing software to production is automated.** In a best-case scenario, pushing new changes to production should be done by clicking a single button or running a single script.

Detailed Syllabus

Lecture-wise Breakup

Subject Code	23M12CS111	**Semester (specify Odd/Even):** Odd	**Semester:** Odd **Session** 2024-2025 **Month from** July to December
Subject Name	Advanced Programming with Python and R		
Credits	3	**Contact Hours**	3

Faculty (Names)	**Coordinator(s)**	Dr. Megha Rathi
	Teacher(s)	Dr. Megha Rathi

COURSE OUTCOMES		**COGNITIVE LEVELS**
C122.1	Understand the concept of advance R and python programming	Level-2- (Understanding)
C122.2	Apply R and Python libraries and modules for data analysis	Level-3 (Apply)
C122.3	Examine performance of statistical model	Level-4 (Analyze)
C122.4	Evaluate performance of models developed in R and Python	Level-5 (Evaluate)
C122.5	Develop Applications using advance programming concepts	Level-6 (Create)

CO-PO-PSO Mapping:

COs	**PO1:** An ability to independently carry out research/investigation and development work to solve practical problems	**PO2:** An ability to write and present a substantial technical report/document	**PO3:** Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program	**PSO 1:** Students should be able to develop and implement the solution of real life computing problems using contemporary technologies	**PSO2:** Students should be able to apply ethical principles and commit to professional and social responsibilities
C122.1	**1** Fundamentals of R & Python			**1** Learn concepts of R & Python to solve computing problems using advanced technologies	
C122.2	**2** Apply R & Python Libraries to solve real-world problems	**1** Students will submit a mini project report	**2** Apply R/Python techniques or packages to solve domain problems	**2** Make use of R/Python techniques to solve domain problems	
C122.3	**2** Derive implications of various statistical models using R & Python	**1** Students will submit a mini project report	**2** Derive implications from statistical modeling	**2** Students will submit a mini-project at the end of semester	
C122.4	**2** Evaluate the performance of mathematical model.	**1** Students will submit a mini project report	**3** Solve a domain specific problem and evaluate it's performance	**3** Students will submit a mini-project at the end of semester	
C122.5	**3** Students will create	**1** Students will	**3** Students will	**2** Students will	

	applications to open problems.	submit a mini project report	create applications to real-world problems.	submit a mini-project at the end of semester	
AVG.	**2.00**	**1.00**	**2.50**	**2.00**	**0.00**

Module No.	Subtitle of the Module	Topics in the module	No. of Lectures for the module
1.	R and Python Basics	Data Types, Input and Output, Operators, Keywords, Identifiers, Output Formatting, String Handling, Control Structure, Conditional Statements, Introduction to the Standard Libraries, Debugger.	4
2.	Functional Programming	Data Structures- Array, Matrices, Tuple, Vector, Data Frame, List Comprehension, Set Comprehension, Dictionary Comprehension, String Handling and manipulation, indexing, slicing, Range.	5
3.	Advanced R and Python Libraries for Data Engineering	Introduction to standard libraries for Data Engineering in R and Pyhton, data frames, Data loading, Data analysis; Create, access, modify, and sort multidimensional arrays, slicing, Boolean indexing, Data Cleaning, Data Wrangling (Join/Combine/ Reshape/Transform), Data Aggregation, Handling Missing& Redundantrecords	7
4.	Import & Export	File Descriptors, Files & Directories, Saving & Loading data, Import and Export to different file formats, Python SQL Database Access using PySQL/RSQL Import/Export Structured data, Querying data, DDL & DML operations, Handling Errors, No-SQL database access/manipulation with Python	7
5.	Regular Expression & Pattern Matching	Regular Expression,RegEx, String handling and manipulation, quantifiers, meta-characters, sequences, Text matching, Repetition, Branching, Pattern-composition	6
6.	R and Python for Data Intelligence	Feature Engineering, Time Series, Predictive Analytics using R and Python, Regression, Decision Tree. Dimensionality Reduction with Principal component analysis, Clustering, Hypothesis Testing, Performance evaluation metrics for supervised and unsupervised learning models	7
7.	Exploratory Data Analysis	Visual Representation of statistical analysis, Exploring univariate and multivariate data with Line plot, Heat Map, QQ, Pie chart, Box/Whisker plot, Scatter plots, Histograms, and Bubble charts using advanced libraries, Geospatial analysis.	6

Total number of Lectures	42

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25Attendance (10 Marks), Assignment/Quiz/Mini-project (15 Marks)
Total	**100**

Project based learning: Students in group of 3 to 4 students are required to develop mini-project based on the concepts taught in this course. In mini-project, students need to create the solution for real-world problems in R/Python. Mini project will enhance statistical skills, data analysis skills, and EDA skills in both R and Python. Students will gain experience in data preprocessing, visualization, and drawing meaningful insights from real-world data.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

1.	Beazley, D. M. (2000). Advanced Python Programming. *Department of Computer Science, University of Chicago*.
2.	Hunt, J. (2019). *Advanced guide to Python 3 programming*. Springer International Publishing.
3.	Kuhlman, D. (2009). *A python book: Beginning python, advanced python, and python exercises* (pp. 1-227). Lutz: Dave Kuhlman.
4.	Hill, C. (2020). *Learning scientific programming with Python*. Cambridge University Press.
5.	Jaworski, M., &Ziadé, T. (2016). *Expert Python Programming*. Packt Publishing Ltd.
6.	Jaworski, M., &Ziadé, T. (2019). *Expert Python Programming: Become a master in Python by learning coding best practices and advanced programming concepts in Python 3.7*. Packt Publishing Ltd.

Detailed Syllabus

Course Code	23M12CS112	**Semester** Odd **Course** MTech	**Semester** 1st **Session** 2024 - 2025 **Month from** Jul. 2024 to Dec.- 2024
Course Name	**Object oriented programming using JAVA**		
Credits	3	**Contact Hours**	3-0-0

Faculty (Names)	**Coordinator(s)**	Dr. Ashish Singh Parihar
	Teacher(s) (Alphabetically)	Dr. Ashish Singh Parihar

COURSE OUTCOMES		**COGNITIVE LEVELS**
C123.1	Understand object-oriented principles in Java to design efficient software solutions.	Understand (Level 2)
C123.2	Implement advanced Java concepts for code modularity and organization.	Apply (Level 3)
C123.3	Illustrating effective error-handling and concurrency strategies.	Analyze (level 4)
C123.4	Analyze data movements through I/O operations and database connectivity.	Analyze (level 4)
C123.5	Create dynamic web applications through Java enterprise standard techniques.	Level-6 (Create)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	***Unit-1*** ***OOPS Concepts and Java Programming***	**OOP concepts:** Classes and objects, data abstraction, encapsulation, inheritance, polymorphism, procedural and object-oriented programming paradigm. **Java programming:** History of Java, comments, Data types, Variables, Constants, Scope and Lifetime of variables, Operators, Type conversion and casting, Enumerated types, Control flow- block scope, conditional statements, loops, break and continue statements, arrays, simple java stand-alone programs, class, object, and its methods constructors, methods.	**10**
2.	***Unit-2*** ***Inheritance, Interfaces and Packages***	**Inheritance:** Inheritance types, super keyword, preventing inheritance through final classes and methods. **Interfaces:** Interfaces Vs Abstract classes, defining an interface, implement interfaces, accessing implementations through interface references, extending interface, inner class. **Packages:** Defining, creating and accessing a package, importing packages.	**6**
3.	***Unit-3*** ***Exception Handling and Multithreading***	**Exception handling:** Benefits of exception handling, the classification of exceptions - exception hierarchy, checked exceptions and	**9**

		unchecked exceptions, usage of try, catch, throw, throws and finally, creating own exception subclasses. **Multithreading:** Differences between multiple processes and multiple threads, thread life cycle, creating threads, interrupting threads, thread priorities, synchronizing threads, inter-thread communication, producer consumer problem.	
4.	***Unit-4*** ***Files and Connecting to Database***	**Files:** Streams- Byte streams, Character streams, Text input/output, Binary input/output, File management using File class. **Connecting to Database**: JDBC Type 1 to 4 drivers, Connecting to a database, querying a database and processing the results, updating data with JDBC,Data Access Object (DAO).	**8**
5.	***Unit-5*** ***Servlets and JSP***	**Servlet:** Introduction to Servlet, Servlet API, Servlet Interface, GenericServlet, HttpServlet, Servlet Life Cycle, RequestDispacher, Cookies in Servlet. **JSP:** JSP Introduction, Life cycle of JSP, JSP API, JSP scripting elements (scriptlet tag, expression tag, declaration tag), JSP Directive Elements (page directive, include directive, taglib directive), JSP Exception.	**9**
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**		
T1	20		
T2	20		
End Semester Examination	35		
TA	25 (	Project	:15
		Attendance	:10)
Total	**100**		

Project based learning: Each student works on different java project. They utilize the concepts taught in lecture and develop project.

The course aims to equip students with a strong foundation in object-oriented programming principles using JAVA, enabling them to design and develop efficient, modular, and scalable software applications, fostering code reusability and maintainability.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)	
Text Book(s):	
1.	Samanta, D., & Sarma, M. (2023). Joy with Java: Fundamentals of Object Oriented Programming. Cambridge University Press.
2.	Ullenboom, C. (2024). Java Programming Exercises: Volume Two: Java Standard Library. CRC Press.
3.	Sharan, K., & Davis, A. L. (2021). Beginning Java 17 Fundamentals: Object-Oriented Programming in Java 17. Apress.
Reference Book(s):	
1.	Kumar, T. S., Reddy, B. E., & Raghavan, P. (2023). Programming with Java. Pearson Education India.
2.	Cosmina, I., & Cosmina, I. (2022). An Introduction to Java and Its History. Java 17 for Absolute Beginners: Learn the Fundamentals of Java Programming, 1-31.
3.	Dingle, A. (2021). Object-Oriented Design Choices. Chapman and Hall/CRC.

CO-PO and CO-PSO Mapping:

COs	**PO1:** An ability to independently carry out research/ investigation and development work to solve practical problems	**PO2:** An ability to write and present a substantial technical report/document	**PO3:** Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program	**PSO 1:** Students should be able to develop and implement the solution of real life computing problems using contemporary technologies	**PSO2:** Students should be able to apply ethical principles and commit to professional and social responsibilities
CO1 Understand object-oriented principles in Java to design efficient software solutions.	**2** Understand OOPs fundamental and features.	**1** Student able to demonstrate the key features of OOPs and submit a mini report.	-	**2** Learn JAVA basics to solve real time computing problems.	-
CO2 Implement advanced Java concepts for code modularity and organization.	**2** Apply code reusability techniques to develop corelated applications.	-	-	**1** Explain code organization and abstraction to solve nested domain problems.	-
CO3 Illustrating effective error-handling and concurrency strategies.	**2** Analyze error case scenarios during the code development.	**1** Student will submit a mini report on concurrency techniques.	**1** Explain the real time synchronous application behavior to students.	**2** Implement concurrent real time application and handle errors during the development.	-
CO4 Analyze data movements through I/O operations and database connectivity.	**2** Illustrate the file movement techniques from local to server through I/O operations during application development.	**1** Student will submit a mini project report on database connectivity through java program.	-	**1** Student will able to organize and manipulate the data through Java program during the development of real-time applications.	-
CO5 Create dynamic web applications through Java enterprise standard techniques.	**2** Design and develop real time dynamic web applications through java enterprise strategies.	**2** Student will submit a mini project report including javadocs.	-	**3** Student will create a real time web application for different domains.	**2** Plan and learn the code ethical properties as it impacts the entire dynamic web application using java enterprise standards. (like, copying etc.)

Detailed Syllabus

Lecture-wise Breakup

Subject Code	23M12CS113	**Semester Odd (specify Odd/Even)**	**Semester Odd Session 2024-2025 Month from July 24 to December 24**
Subject Name	Software Quality and Testing		
Credits	**3**	**Contact Hours**	**3-0-0**

Faculty (Names)	**Coordinator(s)**	Dr. Indu Chawla
	Teacher(s) (Alphabetically)	Dr. Indu Chawla

COURSE OUTCOMES		**COGNITIVE LEVELS**
1.	Describe software quality management processes in the context of Software Development and Engineering.	Understand Level (Level 2)
2.	Utilize quality standards, factors, metrics and models for quality improvement.	Apply Level (Level 3)
3.	Infer the defects and manipulate them for improvement in quality for given Software.	Apply Level (Level 3)
4.	Examine the different testing processes for appropriate testing strategy.	Analyze Level (Level 4)

S.N.	**Subtitle of the Module**	**Topics in the module**	**No. of Lectures for the module**
1.	Overview and Challenges	Overview of Software quality in the context of software development, quality frameworks, perspectives and expectations. Software errors: causes and classification	3
2.	Software quality models and factors	Software quality models: generalized, product specific, their comparison and interactions, Software quality factors: Product operations, revision and transition.	4
3.	Software quality Metrics	Software quality Metrics such as product quality metrics, in process quality metrics, metrics for software maintenance	4
4.	Software quality standards	Scope of quality management standards, SPI, CMMI and six sigma certifications	3
5.	Quality Assurance	Quality assurance techniques and comparisons, Defect prevention and process improvement.	6

6	Quantifiable Quality improvement	Quality assurance monitoring and measurement, immediate follow up actions and feedback.	4
7.	Software testing	Test activities, management and automation, Input domain partitioning and Boundary testing, Control flow, data Dependency and Interaction testing	6
8.	Software testing	Goals of Testing Software, Model-Driven Test Design, Test Automation, Input Space Partitioning, Graph Coverage, Logic Coverage, Syntax-based Testing	6
9.	Coverage and usage testing	Coverage and usage testing based on checklists, partitions, Finite state machines and Markov Chains	6
Total number of Lectures			**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25 Assignment/Quiz/Mini-project (15 Marks) Attendance (10 Marks)
Total	**100**

Project based learning: Each Students in group of 3 to 4 will study about implications of software quality and testing in open source projects. They will present a detailed report or demonstrate the solution proposed. This detailed study using Software quality and testing techniques will help their employability into IT sector.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

Text Books

1.	Daniel Galin, Software Quality: Concepts and Practice, Wiley, 2018
2.	Paul Ammann and Jeff Offutt, Introduction to Software Testing, Cambridge University Press, 2016

Reference Books

1.	Kamna Malik, Praveen Choudhary, Software Quality- A practitioner's approach, Tata Mc Graw Hill, 2009
2.	Jeff Tian, Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement, Wiley India Pvt Ltd, 2005
3.	Gillies, Alan C., Software quality : theory and management, Cengage Learning, 2014
4.	Software Quality Journal-https://www.springer.com/journal/11219
5.	International Conference on Software Engineering- https://dl.acm.org/conference/icse

Detailed Syllabus
Lecture-wise Breakup

Course Code	23M12CS114	**Semester Even (specify Odd/Even)**	**Semester** 1st **Session** 2024-25 **Month from** July 2024 to Dec 2024
Course Name	Computer Vision		
Credits	3	**Contact Hours**	3-0-0

Faculty (Names)	**Coordinator(s)**	Dr. Ankit Vidyarthi
	Teacher(s) (Alphabetically)	Dr. Ankit Vidyarthi

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Understand the fundamental concepts of Computer Vision	Understand Level (Level 2)
CO2	Understand basic concepts, terminology, theories, models and methods in the field of computer vision	Understand Level (Level 2)
CO3	Determine known principles of human visual system	Apply Level (Level 3)
CO4	Illustrate methods related to multi-scale representation, edge detectionand detection of other primitives, stereo, motion and object recognition	Analyzing Level (Level 4)
CO5	Predicting a design of a computer vision system for a specific problem	Evaluate Level (Level 5)

COs	**PO1:** An ability to independently carry out research/ investigation and development work to solve practical problems	**PO2:** An ability to write and present a substantial technical report/document	**PO3:** Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program	**PSO 1:** Students should be able to develop and implement the solution of real-life computing problems using contemporary technologies	**PSO2:** Students should be able to apply ethical principles and commit to professional and social responsibilities
CO1	**1** Covers definitions, vision components introduction, and their use		**2** Covers general terminologies used to understand the vision systems	**3** Basic fundamentals help to build the vision pipeline for problem representation	
CO2	**2** Vision problems require an understanding of models and their working to solve practical problems	**2** Suitable model prediction for specific projects and its demonstration to society	**3** To solve a specific problem having multiple methods and identification of the best among all	**3** Solving the problems using the hybridization of the vision systems with existing algorithms	
CO3			**1** Human visual perspective to solve specific problems		
CO4	**2** Covers image representation using the frequency bands	**2** Representation of the images to understand the hidden pattern		**3** Covers a wide range of algorithms for object representation and template matching	**2** various problems of the society handled using the multi-scale representation
CO5	**3** Design of a vision system for a specific problem		**2** Building new algorithms and procedures for vision problems	**3** New design and algorithms for specific problems	
AVG.	**2**	**2**	**2**	**2**	**2**

Module No.	Title of the Module	Topics in the Module	No. of Lectures for the module
1.	Introduction	Introduction of Computer Vision, Monocular imaging system, Orthographic& Perspective Projection, Cameramodel and Camera calibration, Binocular imaging systems	4
2.	Image Processing and Feature representation	Image representations (continuous and discrete), Edge detection, Image filtering, Thinking in frequency, Image pyramids and applications	6
3.	Feature Detection and Matching	Edge detection, Interest points and corners, Local image features, Feature matching and hough transform, Model fitting and RANSAC	8
4.	Motion Estimation	Regularization theory, Opticalcomputation, StereoVision, Motionestimation,Structure from motion, Feature Tracking and Optical Flow	10
5.	Shape Representation and Segmentation	Deformable curves and surfaces, Snakes and active contours, Level set representations, Fourier and wavelet descriptors, Medialrepresentations,Multi-resolution analysis	8
6.	Object recognition	Hough transforms and other simple object recognition methods, Shapecorrespondence and shape matching, Principal Component analysis, Shape priors for recognition, Mixture of Gaussians and advanced feature encoding	6
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1 Examination	20
T2 Examination	20
End Semester Examination	35
TA	25 (Attendance (10), Tutorial/Quiz/Class-Test/ (5), Mini Project(10))
Total	**100**

Project Based Learning: Students in a group of 2 will take some real world problem and apply AI logics to solve the healthcare problem in a meaning way. Students can able to understand the core logic about data handling and processing.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

	Text Books
1.	Forsyth DA, Ponce J. Computer vision: a modern approach. prentice hall professional technical reference; 2002
2	Lakshmanan V, Görner M, Gillard R. Practical machine learning for computer vision. " O'Reilly Media, Inc."; 2021
	Reference Books
3.	Szeliski, R..*Computer vision: algorithms and applications*. Springer Nature, (2022)
4.	Chen K, Schönlieb CB, Tai XC, Younes L, editors. Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging: Mathematical Imaging and Vision. Springer; 2023
5.	Chowdhary CL, Reddy GT, Parameshachari BD. Computer Vision and Recognition Systems: Research Innovations and Trends. CRC Press; 2022

Detailed Syllabus
Lab-wise Breakup

Course Code	**17M15CS111**	**Semester** ODD	**Semester I Session** 2024 -2025 **Month from** July to Dec 2024
Course Name	Advanced Algorithms Lab		
Credits	1	**Contact Hours**	2

Faculty (Names)	**Coordinator(s)**	Manish Kumar Thakur
	Teacher(s) (Alphabetically)	Manish Kumar Thakur

COURSE OUTCOMES		**COGNITIVE LEVELS**
C170.1	Implement algorithms and use appropriate advanced data structures for solving computing problems.	Level 3: Apply
C170.2	Design algorithms using divide-and-conquer, greedy and dynamic programming strategies, and further recite algorithms that employ these strategies.	Level 3: Apply Level 5: Evaluate
C170.3	Illustrate the mathematical foundation of network flows and some important flow algorithms.	Level 2: Understand Level 3: Apply
C170.4	Implement randomized algorithms to solve various problems, and validate their correctness and complexity.	Level 3: Apply Level 4: Analyze
C170.5	Understand P, NP, polynomial reduction, NP-hardness, and NP-Completeness.	Level 2: Understand Level 4: Analyze
C170.6	Comprehend and select algorithm design approaches in a problem specific manner.	Level 6: Create

Module No.	**Title of the Module**	**List of Experiments**	**CO**
1.	Fundamentals of data structures and algorithmic problem solving	Searching, Sorting, time complexity, Heaps, Arrays, Linked List, Trees, Fibonacci heaps, splay trees, dynamic trees.	CO1
2.	Divide and Conquer Technique	Solving Matrix multiplication problem and subset- sum problem using divide-and-conquer approach	CO2
3.	Greedy Algorithms	Greedy Approximation algorithms- Set Cover Problem, K Centers Problem, Fractional and 0/1 Knapsack, Coinage problem; Bin packing; Job scheduling, Graph coloring; and Text compression using Huffman coding and Shannon-Fanon coding.	CO2
4.	Dynamic Programming Technique	Fundamentals of Dynamic programming based solution approach, Printing Shortest Common Super sequence, Dynamic Programming on Trees, Maximum sum rectangle in a 2D matrix.	CO2
5.	Graph Algorithms	Solve and analyze Graph problems, Algorithms. All Pair Shortest Problem, Subset-sum problem. Minimum Spanning Trees (Prim's and Kruskal algorithms); Shortest Path using Dijkstra's algorithm, K-clique problem, Graph Coloring problem.	CO1, CO2
6.	Flows in Network	Network flows - max flow and min-cost flow/circulation, Edmonds-Karp algorithm	CO3

7.	Tractable and Non- Tractable Problems	One Way of Coping with NP-Hardness. Randomized Rounding. Vertex Cover and Travelling Salesman Problem.	CO4, CO5
8.	Mini-Project	Mini-Project	CO6

Evaluation Criteria

Components	**Maximum Marks**
Lab Test# 1	20
Lab Test# 2	20
D2D work	60
Total	**100**

Project based learning: Students in group of 3 to 4 students are required to develop mini-project based on the concepts taught in this course like Greedy algorithms, dynamic programming, network flow, etc. The solution approach for the identified problem statements should include the usages of advanced data structures including string data structures. The problem statements may be a puzzle-based games, graph-based problems, string-based problems, etc. The developed mini project will enhance the algorithmic thinking and problem-solving approaches of students which are highly desirable to excel in software industries.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

1.	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein , Introduction to Algorithms, MIT Press, 3rd Edition, 2009
2.	Hochbaum “Approximation Algorithms for NP-Hard Problems”, 1996.
3.	Ahuja, Magnanti and Orlin, “Network Flows: Theory, Algorithms and Applications”, 1993.
4.	Horowitz and Sahni, Fundamentals of Computer Algorithms, Computer Science Press, 1978
5.	Study material on //fileserver2