Detailed Syllabus
Lecture-wise Breakup

Course Code **NBA Code**	24M11CA111 CMAC101	**Semester: ODD** **(specify Odd/Even)**	**Semester: I** **Session: 2024-25** **Month from: July- Dec 2024**
Course Name	Data Structure Using C		
Credits	**L-T-P** **3-0-0**	**Contact Hours**	**3**

Faculty (Names)	**Coordinator(s)**	Dr. Shelendra Pal
	Teacher(s) (Alphabetically)	Dr. Shelendra Pal

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Describe the fundamental concepts and constraints of C programming language.	Understand (Level 2)
CO2	Execute the program based on pointers and user-defined Data types like Structures and Unions	Apply (Level 3)
CO3	Demonstrate the usage of various linear data structures like strings, and arrays and measure the performance of various sorting and searching techniques	Apply (Level 3)
CO4	Experiment with different linear data structures and build solutions for various problems and their respective applications	Analyze (Level 4)
CO5	Select appropriate Lists, Tree, BST, B Tree, and Graphs to solve problems.	Evaluate (Level 5)

Module No.	**Title of Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	**Overview of C**	**Overview of C**: History, data types, Primitive and Non-Primitive data, and its limitations, Input and output function, **Control Statements** – if, else-if, switch, **Control Structures** – while, for, do-while, break and continue, goto, Flow chart. **Arrays**: Definition, Representation, Single dimension, Two dimensional, Limitations of array implementation. **Functions**: definition, Categories of functions, Call by Value, Call by reference, Passing arrays to functions, passing strings to functions, Inline function, and Macros.	6
2.	**Advanced C programming**	**Advanced C programming:** **Pointers:** Pointer Expression, Pointer as function arguments, Functions returning pointers, Pointers to Functions. **Structures**: Declaring and using a structure, Nested Structures, Arrays of Structures, Structures and Functions,	5

		Unions: Declaring a Union, Initializing Unions, Accessing a Member of a Union, Arrays of Union Variables, Unions Inside Structures	
3.	**Strings and Data Structures, Searching and Sorting**	**String**: Definition, Representation, String as ADT, Operations – Insert, Delete, Concatenate, Comparing, Substring Arrays of Strings, Pointers, and Strings **Introduction and Classification of Data Structures:** Linear and Nonlinear; Data structure Operations - Insert, Delete, **Searching and Sorting**: Linear Search, Binary Search, Interpolation Search **Sorting**: Bubble, insertion, Selection, Merge, Quick, Radix, Counting Sort, Bucket sort, Heap sort.	7
4.	**Stacks and Queues**	**Stack**: Definition, Representation, Stack as ADT, Operations and Applications of Stack: Polish and reverse polish expressions, Infix to postfix conversion, evaluation of postfix expression, infix to prefix, postfix to infix conversion; Recursion - Factorial, GCD, Fibonacci Sequence **Queue**: Definition, Representation, Queue as ADT, Operations, Queue Variants: Circular Queue, Priority Queue, Double Ended Queue, Applications of Queues.	8
5.	**Linked Lists**	**Linked List**: Memory Management: Static (Stack) and Dynamic (Heap) memory Allocation, Memory management functions, Definition, Representation, Types: Singly Linked List, Inserting and removing nodes from a list, array implementations of stacks, **Double Linked List**: Inserting and Deleting Nodes, Queue as doubly linked lists, such as insert into position, Delete a specified element, Application of Linked Lists. Stacks, Queues, Double-ended Queues, Priority Queues, Sparse Matrix, and Polynomials using Lists.**Circular Linked List**: Inserting, deleting, and searching elements in lists and their applications.	8
6.	**Trees and Graphs**	**Tree:** introduction of the tree, types of tree: Binary Tree, Binary Search Tree, B Tree, tree traversal- pre-order, post-order, Inorder, AVL Tree, Heap, MinHeap, MaxHeap, Skew Heap, Binomial Heap, and Fibonacci Heap **Graph:** Fundamentals of Graph, Adjacency Matrix, List, GraphTraversal using DFS and BFS, Topology Sort, Basic Algorithms – Shortest Path: Dijkstra algorithm, Minimum Spanning Tree, Prime's algorithm, Krushkal algorithm.	8
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25 (Attendance (5), Tutorial/Quiz/Assignment (10), PBL(10))
Total	**100**

Project-based learning: Create an application either individually or in groups of a maximum of 3-4 students each, to illustrate the concepts of the Data Structure application covered in class/

Structure.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication, etc. (Textbooks, Reference Books, Journals, Reports, Websites, etc. in the IEEE format)	
Text Books:	
1	Luciano Manelli, Introducing Algorithms in C: A Step-by-Step Guide to Algorithms in C, Apress, 2020
2	Behrouz A. Forouzan | Richard F. Gilberg, C Programming and Data Structures, 3rd Edition, CL India, 2019
3	Mike McGrath, C Programming in easy steps, 5th edition, In Easy Steps, 2018
4	Seymour Lipschutz, Data Structures with C, Schaum's Outline Series, McGraw Hill, 2010
5	Robert Kruse, C.L. Tondo, Bruce Leung, Shashi Mogalla, Data Structures and Program Design in C, Pearson Education India
6	Richard F Gilberg, Behrouz A. Forouzan, Thomson, Data structure – A Pseudocode Approach with C
Reference Books:	
1	E Balagurusamy, Data Structures using C,1st Edition, 2013, 2013
2	Reema Thareja, Data structures using C, 2nd, Oxford University Press, 2014

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**1**	**1**	**2**	**1**								**2**	
Write programs in C to implement Basic concepts related to control statements, control Structures, Array, and function	Strongly, associating the knowledge of mathematical & logical concepts used in c programming.	Slightly, understand a problem using mathematical & logical concepts used in the c programming	Slightly, extending mathematical & logical concepts used in the c programming	Moderate, understanding the properties of data while applying them to solving complex problems effectively	There is a slight connection as programming and technology are used to real-life concepts.								Moderately, related as the course helps in better understanding of programming language.	
CO2	**2**	**2**	**1**	**2**	**1**	**1**							**2**	**2**
Write programs in C using a pointer and its levels, the user defines the Data type: Structure, Union	Moderately, apply fundamentals of the c programing.	Moderately, c-programming data types, pre-defined functions & concept and used in the creating program.	Slightly related, programming is used to look into complex problems and develop solutions that take cultural, societal, and environmental factors into account.	Moderately, apply the programming concepts to complex problems	There is a slight connection as programming is used for solving real-world problems.	Slightly understanding the management principles							Moderately related due to understanding programming concepts & their functions and their application in real-world scenarios.	Moderately related due to projects related to programming.
CO3	**2**	**1**	**1**	**2**	**1**	**1**							**2**	**1**
Implement various linear data structures, like string, and string array, Analyse the performance of various	Apply the basic principles of array, string, sorting, and searching techniques in a moderate manner.	Slightly, applications of array, searching, and sorting.	Public health and safety, as well as cultural, societal, and environmental considerations, use slightly related array, string, sorting,	Moderately, implement array, string, sorting, and searching strategies to solve the complex problem.	There is a slight connection as programming is used for solving real-world problems.	Slightly understanding the management principles							moderately related approaches apply for arrays, strings, sorting, and searching in real-world applications.	Slightly related due to projects related to programming

sorting techniques and searching			and searching techniques.											
CO4	**2**	**1**	**1**	**2**	**1**	**1**							**2**	**2**
Applying several linear data structures, like as stacks and queues, to build solutions for various problems and their respective applications.	Moderately, apply and analyze the fundamentals of association rule for linear data structure techniques.	Moderately, linear structure techniques employ association rules.	Slightly, are somewhat connected to the application of association rules for linear data structures for public health and safety, as well as social, cultural, and environmental factors.	Moderately, apply and analyze the association rule for linear data structure techniques to complex problems	Slightly related as there is the usage of linear data structure in frequent real-life patterns.	Slightly understanding the management principles							Moderately related due to understanding association rule for linear or non-linear data structure techniques and it apply in real-world scenarios.	Moderatel Moderately related due to projects related to frequent patterns for linear and non-linear
CO5	**1**	**2**	**1**	**1**										**1**
Experiment with lists, multi-linked lists for Doubly Linked List, and circular List. Tree, BST, B Tree, Graph, Graph traversal	Slightly, apply and analyze the nonlinear data structure for the knowledge exploration	Moderately, applications of graph, tree, and tree traversal algorithms	Slightly, related to nonlinear data structure algorithms and MST algorithms are used for public health and safety, and societal considerations .	Slightly, apply and analyze the nonlinear data structure algorithms and MST algorithms are used for complex problems										Slightly related due to projects related to nonlinear data structure algorithms or MST algorithms.

Detailed Syllabus

Lecture-wise Breakup

Course Code **NBA Code**	24M11CA112 CMAC102	**Semester: 1st** **(Specify:)**	**Semester: 1st** **Session: 2024-2025** **Months: JULY**
Course Name	DBMS		
Credits		**Contact Hours**	**42 hr.**

Faculty (Names)	**Coordinator(s)**	Mr. Sumeshwar Singh
	Teacher(s) (Alphabetically)	

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO 1	Student able to learn &understand the fundamental concepts and architecture of database systems.	Understand (Level 2)
CO 2	Apply data modelling techniques to design relational databases.	Apply (Level 3)
CO 3	Utilize SQL for database creation, manipulation, and querying	Apply (Level 3)
CO 4	Implement transaction management and concurrency control mechanisms.	Analyze (Level 4)
CO 5	Apply security & measures, perform administrative tasks in database management.	Evaluate (Level 5)

Module No.	**Title of the Module**	Topics in the Module	**No. of Lectures for the module**
1.	Fundamental concepts of databases and database management systems.	Database Systems vs. File Systems Database Architecture and Data Models Introduction to Relational Databases Advantages and Applications of DBMS Overview of SQL and NoSQL Databases	10
2.	Techniques and methodologies for data modelling and database design.	Entity-Relationship (ER) Model Enhanced ER (EER) Model Relational Model and Schemas Mapping ER and EER Models to Relational Schemas Normalization: Functional Dependencies, Normal Forms (1NF, 2NF, 3NF, BCNF)	8
3.	Comprehensive study of SQL for database creation, manipulation, and querying.	SQL Basics: DDL, DML, DCL Advanced SQL: Joins, Subqueries, Views, Indexes Stored Procedures and Triggers SQL Performance Optimization	8
4.	Concepts of transactions, concurrency control, and recovery in databases.	Transaction Concepts and Properties (ACID), Transaction Control: COMMIT, ROLLBACK, & SAVEPOINT Concurrency Control Mechanisms Lock-Based Protocols and Timestamp-Based Protocols Deadlock Handling Database Recovery Techniques	8

5.	Database security measures and administrative tasks.	Database Security Issues Access Control and Authorization Data Encryption and Security Policies Backup and Recovery Management Database Tuning and Maintenance	8
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25 (Attendance (10), Tutorial/Quiz/Assignment (10), Mini Project (05))
Total	**100**

Project Based Learning: Students will work in teams of size 3-4 to design, implement, and manage a comprehensive database system for a real world and will apply some DBMS techniques, this project will provide hands-on experience in database management so that students can able to understand & admin the core logic about data base handling. It prepares students for real-world database management challenges, equipping them with the necessary skills to design, implement, and manage robust database systems.

Recommended Reading material:	
1.	A Relational Model of Data for Large Shared Data Banks by E.F. Codd The seminal paper that introduced the relational model, foundational for understanding relational databases
2.	The Transaction Concept: Virtues and Limitations by Jim Gray This paper provides insights into the fundamentals of transaction management in databases.
3.	Concurrency Control and Recovery in Database Systems by Philip A. Bernstein, Vassos Hadzilacos, and Nathan Goodman A comprehensive exploration of concurrency control and recovery techniques in database systems. Bigtable: A Distributed Storage System for Structured Data by Fay Chang et al. An important paper on Google's Bigtable, a distributed storage system for managing structured data at a large scale.

Text Books	
1.	Database System Concepts by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, Edition: 6th Edition Publisher: McGraw-Hill Education Fundamentals of Database Systems by Ramez Elmasri and Shamkant B. Navathe Edition: 7th Edition Publisher: Pearson
2.	Database Management Systems by Raghu Ramakrishnan and Johannes Gehrke Edition: 3rd Edition Publisher: McGraw-Hill Education Edition: 3rd Edition Publisher: McGraw-Hill Education

	An Introduction to Database Systems by C.J. Date Edition: 8th Edition Publisher: Addison-Wesley The Complete Reference by James R. Groff and Paul N. Weinberg Edition: 3rd Edition Publisher: McGraw-Hill Education
3.	Database Systems: The Complete Book by Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom Edition: 2nd Edition Publisher: Pearson

Course Outcomes (COs):

1. **CO1:** Understand the fundamental concepts and architecture of database systems.
2. **CO2:** Apply data modelling techniques to design relational databases.
3. **CO3:** Utilize SQL for database creation, manipulation, and querying.
4. **CO4:** Implement transaction management and concurrency control mechanisms.
5. **CO5:** Apply security measures and perform administrative tasks in database management.

Program Outcomes (POs):

1. **PO1:** Ability to apply knowledge of computing and mathematics to solve complex computing problems.
2. **PO2:** Proficiency in problem analysis and identifying computing requirements for solutions.
3. **PO3:** Design, implement, and evaluate computer-based systems to meet desired needs.
4. **PO4:** Function effectively in teams to accomplish a common goal.
5. **PO5:** Understand professional, ethical, legal, security, and social issues and responsibilities.
6. **PO6:** Communicate effectively with a range of audiences.
7. **PO7:** Engage in lifelong learning and professional development.
8. **PO8:** Use current techniques, skills, and tools necessary for computing practice.
9. **PO9:** Apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of computer-based systems.
10. **PO10:** Understand processes that support the delivery and management of information systems within a specific application environment.

PSO

PSO1: To develop ability to analyze complex system requirements in order to design, develop, and test software applications using appropriate programming languages, frameworks, tools, and methodologies.

PSO2: To develop skills to embark on careers as entrepreneurs, innovators, software developers, consultants or pursue interest in teaching, research and development or higher education.

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO2
CO1	Medium relevance as understanding fundamental concepts requires basic computing and mathematical knowledge	High relevance due to the necessity of analyzing and understanding the computing requirements of databases	Low relevance as understanding concepts is foundational and doesn't directly involve design or implementation.						Medium relevance since applying theoretical principles is necessary for understanding database architecture		Low relevance as this CO is more about understanding than developing or testing software	Medium relevance because a solid understanding of database systems supports further specialized development.	Medium
CO2	High relevance as applying knowledge directly involves computing and mathematical concepts.	High relevance due to the critical analysis required in data modeling	High relevance since designing relational databases is a direct application of this PO					Medium relevance as applying current tools and techniques is necessary for effective data modeling.	High relevance since this involves modeling and design based on computer science theory.		Medium relevance because data modeling is a key part of developing software applications.	High relevance as this skill is essential for careers in software development and higher education.	High
CO3	High relevance as SQL usage requires applying computing knowledge.	High relevance due to the analytical skills needed to write and optimize SQL queries.	High relevance because creating and manipulating databases with SQL directly ties into system design and implementation.					High relevance as proficiency with SQL tools is necessary for computing practice.		Medium relevance because understanding information system processes supports effective SQL usage.	High relevance as SQL is essential for developing and testing database-driven applications	High relevance due to SQL’s importance in software development and related careers.	High
CO4	High relevance because it involves applying complex computing concepts.	High relevance due to the necessity of analyzing transaction requirements and concurrency issues.	High relevance as implementation of these mechanisms is critical in system design.					Medium relevance because current techniques and tools are necessary for implementing these mechanisms.		Medium relevance as understanding the processes supports implementation efforts.	Medium relevance because transaction management and concurrency control are important in software application development	High relevance as these skills are crucial for database management roles.	High
CO5	High relevance due to the application of computing knowledge in security and administration	High relevance because analyzing security requirements is crucial.	High relevance as implementing security measures is a part of system design.		High relevance as understanding security issues and responsibilities is critical.			Medium relevance because current security tools and techniques are necessary.		Medium relevance as understanding management processes aids in security and administration.	Medium relevance as security measures are a key part of developing reliable software.	High relevance because security skills are vital in software development and IT careers.	High

CO-PO-PSO Mapping:

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO2
CO1	Medium	High	Low						Medium		Low	Medium	Medium
CO2	High	High	High					Medium	High		Medium	High	High
CO3	High	High	High					High		Medium	High	High	High
CO4	High	High	High					Medium		Medium	Medium	High	High
CO5	High	High	High		High			Medium		Medium	Medium	High	High

Detailed Syllabus
Lecture-wise Breakup

Subject Code	24M11CA113 **CMAC103**	**Semester:** **(specify Odd/Even):**	**Semester: 1st Session: 2024-2025** **Month: July- December 2024**
Subject Name	Object Oriented Programming using Java		
Credits	**3-0-0**	**Contact Hours**	**3**

Faculty (Names)	**Coordinator(s)**	Preeti Mittal
	Teacher(s) (Alphabetically)	Preeti Mittal

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Explain various programming structures in Java using practical problems	Understand (level 2)
CO2	Discover how to manage multiple tasks concurrently	Understand (level 2)
CO3	Demonstrate how to prevent applications from crashing and give meaningful error messages to users	Apply (level 3)
CO4	Illustrate which built in data structure, feature or design pattern to use to solve a problem based on the given scenario	Analyze (level 4)
CO5	Justify usage of Object Oriented Programming principles to solve practical problems	Evaluate (level 5)

Module No.	**Subtitle of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1	Fundamentals of programming in Java	The Java Virtual Machine (JVM) Naming Conventions in Java The main() method Primitive Data Types Variables Constants Reading Input using scanner Class Displaying Output Command Line Arguments Operators Control Flow	4
2	Classes and Objects in Java	Classes and Objects Class and Instance Variables Constructors Instance Methods Class Methods Passing and returning objects Method Overloading OOPS core principles - Inheritance, Polymorphism, Encapsulation and Abstraction	5

		Garbage collection	
3	Inheritance	Superclass and Subclass protected Members this and super keywords Method Overriding final variables, methods and classes	3
4	Packages and Interfaces	Packages Polymorphic behavior Abstract Classes Creating and using interfaces Default and static interface methods Functional interfaces	3
5	Enumerations and Autoboxing	Enumerations Type Wrappers Autoboxing	2
6	Generic Classes and Methods	Generic Classes Generic Methods Generic Interfaces Using Wildcard Arguments	2
7	Utility Classes	String Handling String Tokenizer Date Calendar	2
8	Exception Handling	When to Use Exception Handling Java Exception Hierarchy Checked vs. Unchecked Exceptions Catching an Exception finally Block Declaring New Exception Types throw and throws clause Stack Unwinding and Obtaining Information from an Exception Object Chained Exceptions try-with-Resources: Automatic Resource Deallocation	4
9	Java Collections API	Arrays Collection Framework Collections Interfaces – Collection, List, Map, Set Concrete Collections – ArrayList, HashMap, HashSet, Iterating through Collections	4
10	Multithreading (Concurrency)	Concepts of Multithreading Difference between process and thread Thread States and Lifecycle Creating threads using Thread class and Runnable interface Synchronization Thread Priorities Inter thread Communication Creating and Executing Threads with the Executor Framework	4
11	File Handling in Java	Explore java.io and java.nio packages Perform basic operations like create, read, update, delete on	3

		files using java.io Basic and advanced file handling using java,nio package	
12	Lambdas	Lambda Expressions	2
13	SOLID Principles	Using SOLID principles with examples	1
14	Design Patterns in Java	1. Creational Design Patterns – Factory, Abstract Factory, Singleton 2. Structural Design Patterns – Adapter, Decorator, Façade 3. Behavioral Design Patterns – Command, Observer, State	3
		Total number of Lectures	42

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25 (Attendance (5), Assignment/Tutorial/ Quiz (15), Mini Project (5))
Total	**100**

Project based learning: Create a Java application in groups of maximum 4 students each, to illustrate the concepts covered in class.

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	Herbert Shildt: *Java: The Complete Reference*, 12th Edition, McGraw-Hill, 2021.
2	Y. Daniel Liang: *Intro to Java Programming, Comprehensive Version*, 10th Edition, Pearson, 2018.
	Reference Books
1	Kathy Sierra, Bert Bares & Trisha Gee: *Head First Java,* 3rd Edition, O’Reilly Media, 2022.
2	Paul Deitel and Harvey Deitel: *Java: How to Program, Late Objects*, 11th Edition, Pearson, 2021.

CO-PO and CO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	2	2	-	2	-	2	-	-	-	1	-	1	1
		Moderately be able to identify and analyze complex computing problems using various programming structures in Java	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent using various programming structures available in Java		Moderately be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions using various programming structures available in Java		Recognize the need for and develop the ability to engage in continuous learning to a moderate extent and be apprised of additional features being introduced in newer version of Java				Slightly engage in team work		Slightly be able to analyze complex system requirements in order to design, develop, and test software applications using Java	Slightly be able to develop skills to embark on various career options
CO2	-	2	2	-	1	-	1	-	-	-	1	-	1	1
		Moderately be able to identify and analyze complex computing problems by learning how to handle multiple tasks concurrently using	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent with the ability to handle multiple tasks concurrently using Java		Slightly be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions with the ability to handle multiple tasks concurrently using Java		Recognize the need for and develop the ability to engage in continuous learning to a slight extent and be apprised of new methodologies being introduced to perform multiple tasks concurrently in Java				Slightly engage in team work		Slightly be able to analyze complex system requirements in order to design, develop, and test software applications using Java	Slightly be able to develop skills to embark on various career options

		Java												
CO3	-	2	2	-	1	-	1	-	-	-	1	-	1	1
		Moderately be able to identify and analyze complex computing problems by learning how to prevent applications from crashing using Java	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent with the ability to prevent applications from crashing using Java		Slightly be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions with the ability to prevent applications from crashing using Java		Recognize the need for and develop the ability to engage in continuous learning to a slight extent and be apprised of new methodologies being introduced to prevent applications from crashing using Java				Slightly engage in team work		Slightly be able to analyze complex system requirements in order to design, develop, and test software applications using Java	Slightly be able to develop skills to embark on various career options
CO4	-	2	2	-	1	-	1	-	-	-	1	-	1	1

		Moderately be able to identify and analyze complex computing problems by learning how to use different built in data structures, features or design patterns for developing Java Applications	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent with the ability to use built in data structures, features or design patterns for developing Java Applications		Slightly be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions with the ability to use various built in data structures, features or design patterns for developing Java Applications		Recognize the need for and develop the ability to engage in continuous learning to a slight extent and be apprised of new methodologies being introduced solve complex problems using built in data structures and their APIs, , features or design patterns				Slightly engage in team work		Slightly be able to analyze complex system requirements in order to design, develop, and test software applications using Java	Slightly be able to develop skills to embark on various career options
CO5	-	2	2	-	2	-	2	-	-	-	1	2	2	2
		Moderately be able to identify and analyze complex computing problems by using various Object Oriented Programming principles	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent by using various Object Oriented Programming principles in Java		Moderately be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions with the ability to use various Object Oriented Programming Principles in Java		Recognize the need for and develop the ability to engage in continuous learning to a moderate extent and build upon existing knowledge o using Object Oriented Programming concepts to solve problems				Slightly engage in team work	Moderately be able to use innovative ideas to create value and wealth for the betterment of the individual and society	Moderately be able to analyze complex system requirements in order to design, develop, and test software applications using Java	Moderately be able to develop skills to embark on various career options

Average	**0**	**2**	**2**	**0**	**1.4**	**0**	**1.4**	**0**	**0**	**0**	**1**	**0.4**	**1.2**	**1.2**

Detailed Syllabus
Lab-wise Breakup

Course Code **NBA Code**	24M15CA114 CMAC154	**Semester:** First **(Specify:)**	**Semester:** ODD 2024 **Session:** 2024-25 **Months:** July-December
Course Name	Machine Learning Lab		
Credits	0-0-1	**Contact Hours**	2 Hr

Faculty (Names)	**Coordinator(s)**	Dr. Tarun Agrawal
	Teacher(s) (Alphabetically)	Dr. Shobhit Tyagi and Dr. Tarun Agrawal

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO 1	Use different mathematical concepts related to machine learning.	Apply (Level 3)
CO 2	Apply various machine learning regression techniques on different problems.	Apply (Level 3)
CO 3	Use different machine learning classifiers on the clustering and classification problems.	Apply (Level 3)
CO 4	Apply different data pre-processing techniques for dimensionality reduction.	Apply (Level 3)
CO 5	Compare different concepts related to training of the neural networks.	Analyze (Level 4)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Labs for each module**
1.	Mathematical concepts	Different mathematical concepts such as operators, matrix multiplication, addition, mean, median, variance, standard deviation etc.	3
2.	Regression Techniques	Different regression techniques such as: linear regression, logistic regression, lasso and ridge regression etc.	3
3.	Machine Learning Classifiers	Different classifiers such as k-nearest neighbor, K-means, random forest, decision tree, support vector machine etc.	3
4.	Data pre-processing	Dimensionality reduction techniques such as principal component analysis, singular value decomposition etc.	3
5.	Neural Network Design and Assessment	Building blocks of CNN such as convolutional layers, Activation layers etc, transfer learning and assessing the training of network using the batch normalization, dropout, early-stopping, confusion matrix using the libraries for data visualization and pre-processing such as Matplotlib, SciPy, Scikit-Learn, pandas etc.	3
		Total number of Lab	**15**

Evaluation Criteria

Components	**Maximum Marks**
Evaluation 1	15
Lab Test1	20
Evaluation 2	15
Lab Test 2	20
Mini Project	10
Attendance	10
TA	10
Total	**100**

Project Based Learning: Students in a group of 3-4 will take some real-world problems and apply machine learning concepts to solve the problem in a meaningful way. Students will be able to understand the core mathematical logic about data processing and concepts related to the training of the machine learning models.

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.	Deep Learning- Ian Goodfellow, Yoshua Benjio, Aaron Courville, The MIT Press
2.	Deep Learning with Python – Francois Chollet- Manning Publication
3.	Machine Learning Using Python- Manaranjan Pradhan and U Dinesh Kumar-Wiley

		PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO 1		3	2	2	2									2	2
		Strongly related to the application of mathematics in ML	Moderately related to be applied for the ML problem analysis	Slightly related to apply for design of the solution of ML problem	Moderately related to apply for the investigation of the data and solution of the ML problem									Moderately related to design and development of the software solution of the ML problems	Moderately related to pursue the career in research
CO 2		2	2	1	1									2	
		Moderately related to apply the	Moderately used in analysis for	Slightly related for	Moderately related to apply for the									Moderately related to design of	

		computing regression technique of ML	regression problem	solution of the regression problem	investigation the regression problem									software for the regression problems	
CO 3		2	2	1	1									2	
		Moderately related to be used in the classification problem	Moderately used in analysis of the classification problem	Slightly related to the solution of the classification problem	Moderately related to be used in the investigation of classification problems.									Moderately related to design of software for the classification of dataset	
CO 4		2	2	2	2									2	2
		Moderately used in the data pre-processing	Moderately used in the analysis of the data-pre-processing	Slightly related to be used in the preparation of the dataset	Moderately related to be used in the preparation of the dataset									Moderately related to pre-process the data before feeding it to the software	Moderately related to pursue the career in data analytics
CO 5		2	2	2	2							1		2	2
		Moderately related to the optimization of the neural network	Moderately related to the analysis of the neural network	Slightly related to the training and assessing the neural network	Moderately related in training of neural network models							Slightly related with the project on the neural network		Moderately related to design the ML software for the various problems	Moderately related to pursue the career in neural network research

Detailed Syllabus
Lecture-wise Breakup

Course Code **NBA Code**	24M11CA115 CMAC105	**Semester: Ist** **(Specify:)**	**Semester: Ist** **Session: JULY -DEC 2024** **Months:** JULY
Course Name	Computer System Architecture		
Credits	**3-0-0**	**Contact Hours**	**3**

Faculty (Names)	**Coordinator(s)**	Jyoti
	Teacher(s) (Alphabetically)	Jyoti

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO 1	Apply data representations and understand components of computer system	Understand (Level 2)
CO 2	Develop logic diagrams which illustrating the execution of micro-operations for each of an Instruction available in ISA	Apply (Level 3)
CO 3	Apply the control unit for design and analysis under the instruction execution of data path of computer system	Apply (Level 3)
CO 4	Analyze computer arithmetic algorithms, memory organization and Instruction level pipelining	Analyze (Level 4)
CO 5	Evaluate the performance of computer systems	Evaluate (Level 5)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	Data Representation and Introduction to architectural components	**Data Representation:** Data types, Complements, Fixed and Floating Point representations, IEEE Standard for Floating Point Numbers ,and Binary codes, **Programmer model of CPU:** Computer components (Block diagram of Digital Computer), Functions. **Basic concept of buses:** Interconnection structures Bus interconnection, types of buses and bus arbitration, Bus structure, and Data transfer	6
2.	Arithmetic and logic unit	**Computer Arithmetic:** Addition and subtraction, Look ahead carries adders. Multiplication: Signed operand multiplication, Booths algorithm and array multiplier. Division and logic operations, Decimal Arithmetic operations.	7
3.	Register Transfer Micro operations and Basic Computer Design	**Register Organization:**Register Transfer Language, Register Transfer, Bus and Memory Transfers, Computer Registers	8

		Micro operations: Arithmetic, Logic and Shift micro operations, Arithmetic Logic Shift Unit. **Basic Computer Organization and Design:** Instruction Codes, Computer Instructions ,types, Instruction formats, Timing and Control, Instruction Cycle, Memory reference instruction, Input-Output Instruction and Interrupt.	
4.	Micro-programmed control with CPU, Computer Arithmetic and Instruction Sets	**Microprogrammed Control:** Control memory, Address Sequencing, micro operations, Micro program example, concept of horizontal and vertical microprogramming. **Design of Control Unit:** General Register Organization, Stack Organization, and Program control,Micro programmed control unit, Hardwired Control **Instruction Sets:** Instruction formats, Addressing modes, Data Transfer and Manipulation,	9
5.	Memory and Input/Output Organization	**Memory Organization:** Basic concept of Memory and Memory Hierarchy, Main Memory, RAM and ROM, Auxiliary memory, Associative memory, CPU-memory interaction , Cache memory and related mapping and replacement policies, Virtual memory, Memory Management hardware. **Introduction to input/output processing:** I/O Modules , Programmed controlled I/O transfer , Interrupt controlled I/O transfer, DMA	6
6.	**RISC,** Pipeline Processing and Computer System Performance	**Reduced Instruction Set Computer (RISC) :** – Introduction to RISC and CISC paradigm, CISC Characteristics, RISC Characteristics, Design issues of a RISC processor **Pipeline and Vector Processing:** Introduction to Pipelining System, Parallel Processing, Pipelining in RISC based Systems (ARM), Pipeline Hazards and its solutions, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing. Performance Measures and Laws For Computer Systems, Benchmark tools	6
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20

End Semester Examination	35
TA	25 (Attendance (5), Tutorial/Quiz/Assignment (15), Mini Project (05))
Total	**100**
Project Based Learning: Students in a group of 3-4 will take some real-world problem and apply real time concepts to solve the problem in a meaningful way on simulation platform. Students can able to understand the core logic about data handling and processing and develop a model prototype of real time hardware application.	

Recommended Reading material: M. Morris Mano, Computer System Architecture, Prentice Hall of India Pvt Ltd, 3rd Edition (updated), 30 June 2017.	
1.	Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, Tata McGraw-Hill Education, 5th Edition, 2011
2.	William Stallings, Computer Organization and Architecture–Designing for Performance, Ninth Edition, Pearson Education, 2013.
3.	David Patterson - John Hennessy**,** Computer Organization and Design: The Hardware/Software Interface: ARM Edition,Morgan Kaufmann,2016

Text Books	
1.	M. Morris Mano, Computer System Architecture, Prentice Hall of India Pvt Ltd, 3rd Edition (updated), 30 June 2017.
2.	David A Patterson, John L Hennessy, Computer Organization and Design, Morgan Kaufmann, 6th Edition, 2017

MCA- PROGRAM SPECIFIC OBJECTIVES (PSO)

PSO
PSO1: To develop ability to analyze complex system requirements in order to design, develop, and test software applications using appropriate programming languages, frameworks, tools, and methodologies.
PSO2: To develop skills to embark on careers as entrepreneurs, innovators, software developers, consultants or pursue interest in teaching, research and development or higher education.

MCA-Programme Outcomes (POs)

PO1. Computational Knowledge: Understand and apply mathematical foundation, computing and domain knowledge for the conceptualization of computing models from defined problems.

PO2. Problem Analysis: Ability to identify, critically analyze and formulate complex computing problems using fundamentals of computer science and application domains.

PO3. Design / Development of Solutions: Ability to transform complex business scenarios and contemporary issues into problems, investigate, understand and propose integrated solutions using emerging technologies

PO4. Conduct Investigations of Complex Computing Problems: Ability to devise and conduct experiments, interpret data and provide well informed conclusions.

PO5. Modern Tool Usage: Ability to select modern computing tools, skills and techniques necessary for innovative software solutions

PO6. Professional Ethics: Ability to apply and commit professional ethics and cyber regulations in a global economic environment.

PO7. **Life-long Learning**: Recognize the need for and develop the ability to engage in continuous learning as a Computing professional.

PO8. Project Management and Finance: Ability to understand, management and computing principles with computing knowledge to manage projects in multidisciplinary environments.

PO9. Communication Efficacy: Communicate effectively with the computing community as well as society by being able to comprehend effective documentations and presentations.

PO10. **Societal & Environmental Concern:** Ability to recognize economical, environmental, social, health, legal, ethical issues involved in the use of computer technology and other consequential responsibilities relevant to professional practice.

PO11. Individual & Team Work: Ability to work as a member or leader in diverse teams in multidisciplinary environment.

PO12. Innovation and Entrepreneurship: Identify opportunities, entrepreneurship vision and use of innovative ideas to create value and wealth for the betterment of the individual and society.

CO-PO-PSO Mapping: justification

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2	-	1	1	1	2	-	1	-	-	-	1	-
	Moderately mapped to data representatio	Moderately mapped to data representation		Slightly mapped to data representation	Slightly mapped to data representation	Slightly mapped to data representation	Slightly mapped to data representation		Slightly mapped to data representation				Slightly mapped to data representation	

	n													
CO2	2	2	1	1	1	2	1	-	1	-	-	-	2	-
	Moderately mapped to execute micro-operations	Moderately mapped to execute micro-operations ed	Slightly mapped to execute micro-operations	Slightly mapped to execute micro-operation	Slightly mapped to execute micro-operation	Moderately mapped to execute micro-operations	Slightly mapped to execute micro-operations		Slightly mapped to execute micro-operations				Moderately mapped to execute micro-operations	
CO3	2	2	2	1	1	2	2	1	1	-	-	-	2	-
	Moderately mapped for control unit design	Moderately mapped for control unit design	Moderately mapped for control unit design	Slightly mapped for control unit design	Slightly mapped for control unit design	Moderately mapped for control unit design	Moderately mapped for control unit design	Moderately mapped for control unit design	Slightly mapped for control unit design				Moderately mapped for control unit design	
CO4	2	2	2	1	2	2	2	1	1	-	-	-	2	-
	Moderately mapped for computer arithmetic operations and memory and I/O organization	Moderately mapped for computer arithmetic operations and memory and I/O organization	Moderately mapped for computer arithmetic operations and memory and I/O organization	Slightly mapped for computer arithmetic operations and memory and I/O organization	Moderately mapped for computer arithmetic operations and memory and I/O organization	Moderately mapped for computer arithmetic operations and memory and I/O organization d	Moderately mapped for computer arithmetic operations and memory and I/O organization	Slightly mapped for computer arithmetic operations and memory and I/O organization	Slightly mapped for computer arithmetic operations and memory and I/O organization	-	-	-	Moderately mapped for computer arithmetic operations and memory and I/O organization	
CO5	2	2	2	1	2	2	2	1	1	-	-	-	2	-
	Moderately mapped for computer	Moderately mapped for computer arithmetic operati	Moderately mapped for computer arithmetic operati	Slightly mapped for computer arithmetic operati	Moderately mapped for computer arithmetic operati	Moderately mapped for computer arithmetic operati	Moderately mapped for computer arithmetic operati	Slightly mapped for computer arithmetic operati	Slightly mapped for computer arithmetic operati	-	-	-	Moderately mapped for computer arithmetic operations and pipelining	

	arithmetic operations and pipelining	ons and pipelining	ons and pipelining	ons and pipelining	ons and pipelining	ons and pipelining	ons and pipelining	ons and pipelining	ons and pipelining					
Average	**2**	**2**	**1.4**	**1**	**.4**	**1.8**	**1.8**	**0.6**	**1**	**-**	**-**	**-**	**1.8**	**-**

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO1**	**PSO2**
CO1	2	2	-	1	1	1	2	-	1	-	-	-	1	-
CO2	2	2	1	1	1	2	1	-	1	-	-	-	2	-
CO3	2	2	2	1	1	2	2	1	1	-	-	-	2	-
CO4	2	2	2	1	2	2	2	1	1	-	-	-	2	-
CO5	2	2	2	1	2	2	2	1	1	-	-	-	2	-
Average	**2**	**2**	**1.4**	**1**	**1.4**	**1.8**	**1.8**	**0.6**	**1**	**-**	**-**	**-**	**1.8**	**-**

Detailed Syllabus
Lab-wise Breakup

Subject Code **NBA Code**	24M15CA111 CMAC151	**Semester: ODD** **(specify Odd/Even)**	**Semester: I** **Session: 2024-25** **Month from: July- Dec 2024**
Subject Name	Data Structure Using C Lab		
Credits	**L-T-P** **0-0-2**	**Contact Hours**	**4**

Faculty (Names)	**Coordinator(s)**	Dr. Shelendra Pal
	Teacher(s) (Alphabetically)	Ms. Mayuri, Shelendra Pal

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	**Identify** the proper data types and data structures for given problems.	Understand (Level 2)
CO2	**Demonstrate** the basic data structure and its operations.	Apply (Level 3)
CO3	**Solve** the various linear data structure problems and check the performance of sorting and searching techniques.	Apply (Level 3)
CO4	**Implement** nonlinear data structures like trees, and graphs.	Apply (Level 3)
CO5	**Experiment** with advanced data structures and their applications.	Analyse (Level 4)

Module No.	**Title of the Module**	**List of Experiments**	**No. of Labs for the module**
1	Control statement, Array, and Function	Output-based C programs to implement the concepts of Input/output function, Control Statements, loops, arrays from basic to two-dimensional, Function: Call by value/reference, inline function.	**6**
2	Pointer and User-defined Data Types	Output-based programs in C to implement concepts of pointers and their level, Structure, and nested structure, and experiment with array and function. Union declared its experiments	**4**
3	String, searching, and Sorting	Searching – Linear Search, Binary Search, Interpolation Search, Bubble, insertion, Selection Merge, Quick, Radix. Time and Space complexity analysis of searching and sorting algorithms	**4**
4	Linear Data Structures	Implementation of Stack: using array, and its applications: Polish and reverse Polish expressions, Recursion - Factorial, GCD, Fibonacci Sequence, matrix multiplication: Implementation of Queue: using array, and its Variants: Circular Queue, Priority Queue,	**5**

		Double Ended Queue, applications	
5	Linear Data Structures - II	Apply and implement various linear data linked list: single, Doubly, circular, and all operations like creation, insertion deletion.	**6**
6	Non-Linear Data Structure – Tree or Graph, MST	Implementation of a tree: B tree, BST and its traversal operation: AVL tree, Adjacency Matrix and List; Traversal using DFS and BFS, Basic Algorithms – Shortest Path, Minimum Spanning Tree	**7**
		Total number of Lab	**32**

Evaluation Criteria

Components	**Maximum Marks**
Evaluation 1	15
Lab Test1	20
Evaluation 2	15
Lab Test 2	20
PBL	20 (Students will submit the mini project in a group of 3-4 members)
Attendance	10
Total	**100**

Project-based learning: Create an application either individually or in groups of a maximum of 3-4 students each, to illustrate the concepts of the Data Structure application covered in class/Structure.

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

Text Books:

1	Luciano Manelli, Introducing Algorithms in C: A Step-by-Step Guide to Algorithms in C, Apress, 2020
2	Behrouz A. Forouzan | Richard F. Gilberg, C Programming and Data Structures, 3rd Edition, CL India, 2019
3	Mike McGrath, C Programming in easy steps, 5th edition, In Easy Steps, 2018
4	Seymour Lipschutz, Data Structures with C, Schaum's Outline Series, McGraw Hill, 2010
5	Robert Kruse, C.L. Tondo, Bruce Leung, Shashi Mogalla, Data Structures and Program Design in C, Pearson Education India
6	Richard F Gilberg, Behrouz A. Forouzan, Thomson, Data structure – A Pseudocode Approach with C

Reference Books:

1	E Balagurusamy, Data Structures using C,1st Edition, 2013, 2013
2	Reema Thareja, Data structures using C, 2nd, Oxford University Press, 2014

COs	CO Attainments	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO 1		3	3		3	3		3						3	3
CO 2		1	1		1	1		1						1	1
CO 3		2	2		2	2		2						2	1
CO 4		2	2		2	2		2						2	1
CO 5		2	2		2	2		2						2	2

Detailed Syllabus
Lab-wise Breakup

Course Code **NBA Code**	24M15CA112 CMAC152	**Semester** Odd	**Semester** I **Session** 2024-25 **Month from July ’24 to Dec’24**
Course Name	Database Management System		
Credits	0-0-1	**Contact Hours**	0-0-2

Faculty (Names)	**Coordinator(s)**	Aarti Goel
	Teacher(s) (Alphabetically)	Aarti Goel and Kirti Jain

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO.1	Contrast various database schemas to represent real-world scenarios, demonstrating entity-relationship modeling, entity integrity, referential integrity, key constraints, and domain constraints.	Understanding (Level II)
CO.2	Execute simple and complex queries for database creation and manipulation.	Apply (Level III)
CO.3	Illustrate commands to retrieve data from database.	Analyse (Level IV)
CO.4	Assess and validate PL/SQL blocks, procedure functions, packages, triggers, and cursors.	Evaluate (Level V)
CO.5	Design a real-time application of database management system.	Create (Level VI)

Module No.	**Title of the Module**	**List of Experiments**	**CO**	**No. of Hours**
1.	ER diagram	1. Creating Entity-Relationship Diagram using case tools such as Draw.io, Microsoft Visio, etc.	CO.1	4
2.	Introduction to MySQL Commands	A case study will be discussed to: 1. Create table, alter table, Drop table 2. Select statement with operations like Where clause, Order by, Logical operators, Scalar functions and Aggregate functions	CO.2	4
3.	Introduction to Joins & Nested Queries	A case study will be discussed to: 1. Create Insert, Update, Delete and Select Statements. 2. Simple Queries, Sorting Results (ORDER BY Clause) 3. SQL Aggregate Functions 4. Grouping Results (GROUP BY Clause) 5. Subqueries, ANY and ALL, Multi-Table Queries, EXISTS and NOT EXISTS 6. Combining Result Tables (UNION, INTERSECT,	CO.3	12

		EXCEPT)		
4.	Procedural Language/ SQL	A case study will be discussed to: 1. Write PL/SQL program for storing data using procedures. 2. Write PL/SQL program for storing data using stored functions. 3. Write PL/SQL program for storing data using cursors and Triggers	CO.4	6
5.	Security	Implement access control, authorization and data encryption techniques.	CO.5	4
		Total Hours		30

Evaluation Criteria

Components	**Maximum Marks**
Lab Test-1	**20**
Lab Test-2	**20**
Day-to-Day	**60 (Project-15, Lab Assessment- 15 + 15, Attendance-15)**
Total	**100**

Project based learning: Each student in a group of 3-4 will have to develop a project based on different real-world problems. Students must study database related Technologies before finalizing the objectives. For handling the multiple records, they will implement cursors and triggers. Student will connect the database to cloud. 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. in the IEEE format)

1.	Henry F Korth, Abraham Silberschatz, S. Sudurshan, Database system concepts, 7th Edition, McGraw-Hill,2019
2.	RamezElmasri ,Shamkant B. Navathe, Fundamentals of Database Systems, 5thEdition, Pearson Education, 2015.
3.	Ramakrishnan, Gehrke, Database Management Systems, Mcgraw-Hill, 3rd Edition, Addison-Wesley,2014.
4.	Thomas Connolly, Carolyn Begg, Database Systems-A Practical Approach to design, Implementation and Management, 6rd Edition, Addison-Wesley,2015.
5.	"PHP and MYSQL Manual" by Simon Stobart and Mike Vassileiou
6.	"PHP and MYSQL Web Development" by Luke Welling and Laura Thomson (Pearson Education), 5th Edition, 2016.

CO-PO-PSO MAPPING

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO.1	1	3	2	2	1				2				2	2

	Moderately mapped	Strongly mapped	Moderately mapped	Moderately mapped	Slightly mapped				Moderately mapped				Moderately mapped	Moderately mapped
CO.2	1	2		1	3				2				2	2
	Moderately mapped	Strongly mapped	Moderately mapped	Moderately mapped	Slightly mapped				Moderately mapped				Slightly mapped	Moderately mapped
CO.3	1	3	2	2	3				2				2	2
	Slightly mapped to make use of relational database and SQL commands for query processing.	Moderately mapped to make use of relational database and SQL commands for query processing.		Slightly Mapped to make use of relational database and SQL commands for query processing.	Strongly mapped to make use of relational database and SQL commands for query processing.				Moderately mapped to make use of relational database and SQL commands for query processing.				Moderately mapped to make use of relational database and SQL commands for query processing.	Moderately mapped to make use of relational database and SQL commands for query processing
CO.4	1	2		1	3				2				2	2
	Slightly mapped to make use of PL/SQL commands for query processing.	Moderately mapped to make use of PL/SQL commands for query processing.	.	Slightly mapped to make use of PL/SQL commands for query processing.	Strongly mapped to make use of PL/SQL commands for query processing.				Moderately mapped to make use of PL/SQL commands for query processing.				Moderately mapped to make use of PL/SQL commands for query processing.	Moderately mapped to make use of PL/SQL commands for query processing.
CO.5	3	3	3	3	3		2		3	3	3	3	2	2
	Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.		Moderately mapped as applying various concepts of database to incorporate real world application.		Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.	Strongly mapped as applying various concepts of database to incorporate real world application.	Moderately mapped as applying various concepts of database to incorporate real world application.	Moderately mapped as applying various concepts of database to incorporate real world application.
NBA Code:	1	3	2	2	3		2		2	3	3	3	2	2

Detailed Syllabus
Lab-wise Breakup

Subject Code **NBA Code**	24M15CA113 **CMAC153**	**Semester: Odd**	**Semester: 1st Session: 2024-2025** **Month: July- December 2024**
Subject Name	Object Oriented Programming using Java Lab		
Credits	**0-0-1**	**Contact Hours**	**2**

Faculty (Names)	**Coordinator(s)**	Varun Srivastava
	Teacher(s) (Alphabetically)	Preeti Mittal and Varun Srivastava

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Interpret various programming structures in Java for solving various engineering problems	Understand (level 2)
CO2	Execute tasks concurrently to achieve better performance	Apply (level 3)
CO3	Examine various methods to prevent program crashing and accordingly generate meaningful messages for users	Analyze (level 4)
CO4	Assess the use of various built-in data structures and design patterns for solving a complex problem	Evaluate (level 5)
CO5	Develop programmable solutions to demonstrate Object Oriented Programming principles for solving real world problems.	Create (level 6)

Module No.	**Subtitle of the Module**	**Topics in the Module**	**No. of Labs for the module**
1	Fundamentals of programming in Java	Data Types Variables Constants Input-Output classes Command Line Arguments Operators and type of operators Control Flow	1
2	Classes and Objects in Java	Classes and Objects Class and Instance Variables Constructors Instance Methods, this keyword, finalize Class Methods Passing and returning objects enum Types Method Overloading Arrays and Strings	1
3	Inheritance	Superclass and Subclass protected Members	1

		this and super keywords Method Overriding final variables, methods and classes	
4	Packages and Interfaces	Packages Polymorphic behavior Abstract Classes Creating and using interfaces	1
5	Enumerations and Autoboxing	Enumerations Type Wrappers Autoboxing	1
6	Generic Classes and Methods And Utility Classes	Generic Classes Generic Methods Generic Interfaces Using Wildcard Arguments String Handling StringTokenizer Date Calendar	3
7	Exception Handling	When to Use Exception Handling Checked vs. Unchecked Exceptions Catching an Exception finally Block Declaring New Exception Types throw and throws clause Stack Unwinding and Obtaining Information from an Exception Object Chained Exceptions try-with-Resources: Automatic Resource Deallocation	1
8	Collections Framework	Collections Interfaces, classes Comparators Legacy classes and Interfaces	1
9	Multithreading (Concurrency)	Creating threads using Thread class and Runnable interface Thread priorities and Synchronization Inter thread Communication Creating and Executing Threads with the Executor Framework	1
10	File Handling in Java	Explore java.io and java.nio packages Perform basic operations like create, read, update, delete on files using java.io Basic and advanced file handling using java,nio package	1
11	Lambdas and Applets	Lambda Expressions Applet architecture and skeletonization	1
12	Design Patterns in Java	1. Creational Design Patterns – Factory, Abstract Factory, Singleton 2. Structural Design Patterns – Adapter, Decorator, Façade	2

		3. Behavioral Design Patterns – Command, Observer, State	
		Total number of Labs	15

Evaluation Criteria **Components** **Maximum Marks** Eval 1 15 Eval 2 15 Lab Test 1 20 Lab Test 2 20 PBL 15 (Students will submit the mini project in a group of 3-4 members) Attendance 15 **Total** **100**	

Project based learning: Create a Java application in groups of maximum 4 students each, to illustrate the concepts covered in class.

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	Herbert Shildt: *Java: The Complete Reference*, 12th Edition, McGraw-Hill, 2021.
2	Y. Daniel Liang: *Intro to Java Programming, Comprehensive Version*, 10th Edition, Pearson, 2018.
	Reference Books
1	Kathy Sierra, Bert Bares & Trisha Gee: *Head First Java,* 3rd Edition, O'Reilly Media, 2022.
2	Paul Deitel and Harvey Deitel: *Java: How to Program, Late Objects*, 11th Edition, Pearson, 2021.

CO-PO and CO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1	2	2	2	-	2	-	2	-	-	-	1	-	1	
	Apply basic data structures in Java for simple engineering problems	Moderately be able to identify and analyze complex computing problems using various programming structures in Java	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent using various programming structures available in Java		Moderately be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions using various programming structures available in Java		Recognize the need for and develop the ability to engage in continuous learning to a moderate extent and be apprised of additional features being introduced in newer version of Java				Participation in team work		Analyze complex system requirements in order to design, develop, and test software applications using Java	
CO2	-	2	2	-	1	-	1	-	-	-	1	-	1	
		Moderately be able to identify and analyze complex computing problems	Design and develop solutions of complex business scenarios and contempo		Slightly be able to select modern computing tools, skills and techniques necessary		Recognize the need for and develop the ability to engage in continuous learning				Participation in team work			

		by learning how to handle multiple tasks concurrently using Java	rary issues and propose integrated solutions to a moderate extent with the ability to handle multiple tasks concurrently using Java		for providing innovative software solutions with the ability to handle multiple tasks concurrently using Java		to a slight extent and be apprised of new methodologies being introduced to perform multiple tasks concurrently in Java							
CO3	-	2	2	-	1	-	1	-	-	-	1	-	1	1
		Moderately be able to identify and analyze complex computing problems by learning how to prevent applications from crashing using Java	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent with the ability to prevent applications from crashing using Java		Slightly be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions with the ability to prevent applications from crashing using Java		Recognize the need for and develop the ability to engage in continuous learning to a slight extent and be apprised of new methodologies being introduced to prevent applications from crashing using Java				Engage in team work		Analysis of complex system requirements in order to design, develop, and test software applications using Java	Develop skills to embark on various career options

CO4	-	2	2	-	1	-	1	-	-	-	1	-	1	
		Moderately be able to identify and analyze complex computing problems by learning how to use different built in data structures for developing Java Applications	Design and develop solutions of complex business scenarios and contemporary issues and propose integrated solutions to a moderate extent with the ability to use built in data structures for developing Java Applications		Slightly be able to select modern computing tools, skills and techniques necessary for providing innovative software solutions with the ability to use various built in data structures for developing Java Applications		Recognize the need for and develop the ability to engage in continuous learning to a slight extent and be apprised of new methodologies being introduced solve complex problems using built in data structures and their APIs				Slightly engage in team work		Slightly be able to analyze complex system requirements in order to design, develop, and test software applications using Java	
CO5	-	2	2	-	2	1	2	-	1	-	2	2	2	2
		Moderately be able to identify and analyze complex computing problems by using various Object Oriented	Design and develop solutions of complex business scenarios and contemporary issues and propose		Moderately be able to select modern computing tools, skills and techniques necessary for providing innovative	Slightly understand the need of following professional ethics while building solutions to problems	Recognize the need for and develop the ability to engage in continuous learning to a moderate extent and build		Slightly be able to communicate solutions effectively with community		Moderately engage in team work as a member or as a leader	Moderately be able to use innovative ideas to create value and wealth for the betterment of the individual and society	Moderately be able to analyze complex system requirements in order to design, develop, and test software applications	Enhancement of skills to embark on various career options

		Programming principles	integrated solutions to a moderate extent by using various Object Oriented Programming principles in Java		software solutions with the ability to use various Object Oriented Programming Principles in Java		upon existing knowledge o using Object Oriented Programming concepts to solve problems						using Java	
Average	**0.13**	**2**	**2**	**0**	**1.4**	**0.2**	**1.4**	**0**	**0.2**	**0**	**1.2**	**0.4**	**1.2**	**0.6**

Detailed Syllabus

Lecture-wise Breakup

Course Code **NBA Code**	24M11CA114 CMAC104	**Semester: 1st**	**Semester: 1st** **Session: ODD 2024** **Months: July-December**
Course Name	Machine Learning		
Credits	**3-0-0**	**Contact Hours**	**3**

Faculty (Names)	**Coordinator(s)**	Dr. Sakshi Gupta
	Teacher(s) (Alphabetically)	Dr. Sakshi Gupta

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO 1	Describe the various mathematical concepts used in machine learning.	Understand (Level 2)
CO 2	Illustrate the different concepts related to machine learning.	Apply (Level 3)
CO 3	Illustrate the various concepts related to the training of the neural network model.	Apply (Level 3)
CO 4	Analyze the various concepts related to the optimization of the deep neural models.	Analyze (Level 4)
CO 5	Assess the different neural network architectures.	Evaluate (Level 5)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	Linear Algebra and Probability	Discrete and continuous probability, sum rule, product rule, mean, median, standard deviation, Linear equations, solving linear equations, matrices, matrix approximation, vector space, inner product, angles and orthogonality, orthogonal complement, inner product, orthogonal projections and rotations, linear independence	8
2.	Introduction to ML	Machine learning introduction, Data Science Vs Machine Learning; mathematics in machine learning, linear regression and logistic regression, lasso and ridge regression, Data analysis and visualization using Matplotlib, NumPy, Pandas, Matplotlib, Scikit-learn, SciPy, PyTorch, and Keras	10
3.	Fundamentals of ML	Fundamentals of machine learning, machine learning problems, types of learning: supervised, unsupervised, semi-supervised learning, reinforcement learning, Different Classifiers: k-nearest neighbor, Decision Tree and Random Forest, Real-World Use Cases of ML: Social Media (Facebook), Cancer Prognosis and Prediction and Fraud Detection.	8
4.	Neural network concepts	Dimensionality Reduction: PCA, Singular Value Decomposition Perceptron's, Multilayer perceptron, Gradient descent, Multilayer networks, Backpropagation Algorithm, Confusion matrix, Loss functions, Bias-variance trade-off, overfitting-underfitting, Activation functions	8
5.	Neural Network	Introduction to Convolutional Neural Network, building blocks of CNN, Transfer Learning, Effective training in Deep Net- early	8

	Models	stopping, Dropout, Batch Normalization, Instance Normalization, Group Normalization, Neural Network architecture: AlexNet, VGG-Net, ResNet, Inception-Net, UNet	
		Total number of Lectures	**42**

Evaluation Criteria

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

Project Based Learning: Students in a group of 3-4 will take some real-world problems and apply machine learning concepts to solve the problem in a meaningful way. Students can able to understand the core logic about data handling and processing using machine learning models.

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.	Chollet, Francois. Deep learning with Python. Simon and Schuster, 2021.
2.	Machine Learning - A Complete Exploration of Highly Advanced Machine Learning Concepts, Best Practices and Techniques by Peter Bradley, Draft2digital, 25 June 2019.
3.	Bengio, Yoshua, Ian Goodfellow, and Aaron Courville. Deep learning. Vol. 1. Cambridge, MA, USA: MIT press, 2017.

Reference Books	
1.	Weidman, Seth. Deep learning from scratch: Building with python from first principles. O'Reilly Media, 2019.
2.	Alpaydin, Ethem. Introduction to machine learning. MIT press, 2020.
3.	Ravichandiran, Sudharsan. Hands-On Deep Learning Algorithms with Python: Master deep learning algorithms with extensive math by implementing them using TensorFlow. Packt Publishing Ltd, 2019.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**1**	**1**	**2**	**1**								**2**	
CO2	**2**	**2**	**1**	**2**	**1**	**1**							**2**	**2**
CO3	**2**	**1**	**1**	**2**	**1**	**1**							**2**	**1**
CO4	**2**	**1**	**1**	**2**	**1**	**1**							**2**	**2**
CO5	**1**	**2**	**1**	**1**										**1**