Mandsaur University
Department Of Allied Science

Syllabus for MCA (Computer Applications)
Semester-II

25ALS050T : Mathematical foundation of computer science **L-3,T-1,P-0,C-4**

Course Objective:

- To gain knowledge about Sets, Relations and Functions.
- To study the basics of Proportions & Lattices concepts with their application.
- To provide students with an introduction to the group theory and fields .
- To study the basics of graph theory concepts with their application.
- To give understanding of the theoretical and practical aspects of the use of discrete numeric function and recurrence relation.
- To develop analytical ability to solve real-world problems using these methodologies.

Course Outcomes (COs)

1. Understand the properties of sets, relations, functions and discrete structure
2. Apply the concept of Proportions & lattices for solving many computer applications.
3. Analyze the concepts of set theory group, and field theory for solving computer applications models.
4. Evaluate shortest path using graphical algorithm used in computer models.
5. Evaluate some Recurrence Relation using generating function.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PSO 1	PSO 2
CO1	3	2	1	1	1	-	-	-	-	-	1	1
CO2	1	3	2	1	1	-	-	-	-	-	1	1
CO3	1	2	2	1	1	-	-	-	-	-	1	2
CO4	2	2	3	1	-	-	-	-	-	-	1	2
CO5	1	2	2	1	-	-	-	-	-	-	1	1

High-3 Medium-2 Low-1

Unit-I: Sets theory, Relations and Functions
12 Hours

Sets, Subsets, Power sets, Complement, Union and Intersection, De-Morgan's law Cartesian products. Relations: relational matrices, properties of relations, equivalence relation. Functions: Injection, Surjection and Bijective mapping, Composition of functions, the characteristic functions and Mathematical induction.

Unit-II: Proportions & Lattices
12 Hours

Proposition & prepositional functions, logical connections, truth-values and truth table, the algebra of prepositional functions, the algebra of truth values, applications (switching circuits, basic computer components). Partial order set, Hasse diagrams, upper bounds, lower bounds, maximal and minimal element, first and last element.
Lattices: Lattices, sub lattices, Isotonicity distributive inequality, Lattice homomorphism, lattice isomorphism, complete lattice, Complemented lattice distribution lattice. .

Unit-III: Group theory and Fields
12 Hours

Group axioms, permutation group, sub group, co-sets, normal subgroup, semi group, Lagrange theorem, fields, minimal polynomials, reducible polynomials, primitive polynomial, polynomial roots, applications.

15/5/25 15/05/25 15/05/25 15/05/25

Unit-IV: Graphs **12 Hours**

Finite graphs, incidence and degree, isomorphism, sub graphs and union of graphs, connectedness, walk, paths, and circuits. Eulerian graphs, tree properties of trees, pendant vertices in tree, center of tree, spanning trees and cut vertices, binary tree ,matrix representation of graph, incidence and adjacency matrix and their properties, applications of graphs in computer science.

Unit-V: Discrete Numeric function and Recurrence relation **12 Hours**

Introduction to discrete numeric functions and generating functions, introduction to recurrence relations and recursive algorithms, linear recurrence relations with constant coefficients, homogeneous solutions, particular solutions and total solutions.

Total: 60 Hours

Reference Books:

1. Tremblay, J. P., & Manohar, R. (1975). *Discrete mathematical structures with applications to computer science.* McGraw-Hill, Inc..
2. Ross, K. A., & Wright, C. R. (1985). *Discrete mathematics.* Prentice-Hall, Inc..
3. Deo, N. (2017). *Graph theory with applications to engineering and computer science.* Courier Dover Publications.
4. Khanna, V. K., & Bhamri, S. K. (2016). *A course in abstract algebra.* Vikas Publishing House.

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/
3. https://www.edx.org/course/advanced-algorithmics-and-graph-theory-with-python?index=product&queryID=106bd43f975a7c909005bc27e62f3c98&position=3&v=1&linked_from=autocomplete&c=autocomplete

15/05/25

15/05/25

15/5/25

HOD
Department of Mathematics
Mandsaur University, Mandsaur

15/05/25

Anchal
15/05/25

Sona
15/05/25

MCA
Semester-II

L-2 T-1 P-0 C-3

25MCA300T: Linux Operating System

Course Objectives

- To familiarize students with the Linux environment.
- To learn Command line operations and Linux file structure.
- To learn managing user accounts.
- To Learn Local Security Principles.
- To learn the fundamentals of shell scripting/programming.

Course Outcomes (COs)

1. Understand the components of Linux, including distributions, boot processes.
2. Apply between CLI and GUI interfaces, and proficiently navigate directories.
3. Analyze the usage of package management systems for software installation.
4. Evaluate user accounts, groups, and the root account.
5. Create Bash shell scripts incorporating conditional statements and string manipulation.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	-	3	-	-	2	-	1	2	-	-	-	-
CO2	2	1	3	-	-	1	-	2	1	-	-	-
CO3	-	1	-	3	-	-	2	-	1	-	-	-
CO4	1	-	-	2	3	-	-	1	-	-	-	-
CO5	2	1	2	-	-	3	1	2	-	-	-	-

High-3 Medium-2 Low-1

Unit-I: Introduction to Linux 9 Hours

Linux distributions, more about Linux distributions, Linux structures and installations, installing the Linux system, The Boot Process, BIOS Processes, Master Boot Records (MBR) and Boot Loader, The Linux Kernel, The Initial RAM Disk. Graphical Interface: CLI & GUI, logging in and Out, Locking the Screen, Switching Users, Shutting Down and Restarting, Suspending. **Network Manager and configuration:** Network Configuration, Wired and Wireless Connections and its configurations.

Unit-II: Command line operations 9 Hours

Introduction to the Command Line and its operations, Launching Terminal Windows, The X Window System, Virtual Terminals, sudo, Steps for Setting up and running sudo, Difference between sudo and su. **Basic Operations:** Logging in and Out, Rebooting and Shutting Down, Locating- applications, Accessing Directories using command prompt, Absolute and Relative Paths, Hard and Soft (Symbolic) Links, Navigating the Directory History. Working with files: Standard File Streams, I/O Redirection, Modifying the Command Line Prompt, Pipes, Searching for Files, finding fields in a Directory, Finding Files Based on Time and Size. Viewing files, Removing a file or directory.

Unit-III: File system Architecture **9 Hours**

Package Management Systems on Linux, Working with Different Package Management Systems. **File Operations:** Introduction to filesystem and its hierarchy, Mount points, The Network File system. **The File system Architecture:** Overview of Home Directories, The /bin and /sbin Directories, The /dev Directory, The /var and /etc Directories, The /boot Directory, The /lib and /media-directories, Additional directories under /, Subdirectories under /usr.

Text Editors: Creating Files without Using an Editor, nano and gedit, more advanced editors: vi, emacs, vim, nano etc.

Unit-IV: User Environment **9 Hours**

Use and configure user accounts and user groups-Identifying the Current User, Basics of Users and Groups, Adding and Removing Users, Adding and Removing Groups, The Root Accounts, Elevating to root Account, Startup files. **Environment-Variables:** Setting Environment Variables, the PS1 Variable, the SHELL Variable, File Ownership, File Permission Modes, chmod, chown, chgrp. Working with passwords: How passwords are stored, Password Encryption, Securing the Boot Process and Hardware Resources: Requiring Boot Loader Passwords, Hardware Vulnerability.

Unit-V: Bash Shell Scripting **9 Hours**

Various types of shell, shell programming in bash, conditional and looping statements, case statements, parameter passing and arguments, Shell variables, shell keywords, Creating Shell programs for automate system tasks and report printing, use of grep commands, awk programming. Manipulate strings to perform actions such as comparison and sorting, create temporary files and directories, Create and use random numbers.

Total: 45 Hours

Reference Books:

1. Linux Administration: A Beginner's Guide by Steve Shah,Wale Soyinka, ISBN 0072262591(0-07-226259-1), McGraw-Hill Education
2. Unix for programmers and users (3rd Edition)- Graham Glass & King ables,Pearson Education India.(Low Prices Edition)
3. UNIX Concepts and Applications by Sumitabha Das,Tata Mcgraw hill publications.
4. Unix and Linux System Administration Handbook, by Evi Nemeth,Garth Snyder, Trent R. Hein,Ben Whaley.
5. Linux System Administration,By Tom Adelstein,Bill Lubanovic.

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-2 T-1 P-0 C-3

25MCA310T: Object Oriented Programming using Java

Course Objectives

- To learn basic concepts of Java.
- To understand Concepts of Object-Oriented Programming in Java.
- To learn Multithreading and Exception Handling.
- Learn to develop GUI based applications.
- To learn event handling and database connectivity.

Course Outcomes

1. Understand comprehension of fundamental Java concepts.
2. Apply knowledge of inheritance, polymorphism, packages, and interfaces.
3. Analyze exception handling mechanisms and multi-threading concepts.
4. Evaluate AWT controls and event handling techniques.
5. Create knowledge of Swing components, applet development, and JDBC connectivity.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	-	3	-	-	2	-	1		2	-	-	-
CO2	1	-	3	2	-	1	-	2	-	-	-	-
CO3	2	1	-	3	-	1	-	-	-	1	-	-
CO4	1	-	1	2	3	-	2	1	-	2	-	-
CO5	2	-	2	-	-	3	1	2	1	-	-	-

High-3 Medium-2 Low-1

Unit – I: JAVA BASICS 9 Hours

Review of Object-oriented concepts, History of Java, Java buzzwords, JVM architecture, Data types, Variables, Scope and lifetime of variables, arrays, operators, control statements, type conversion and casting, simple java program, constructors, methods, Static block, Static Data, Static Method String and Stringbuffer Classes, Using Java API Document.

Unit –II: INHERITANCE AND POLYMORPHISM 9 Hours

Basic concepts, Types of inheritance, Member access rules, Usage of this and Super keyword, Method Overloading, Method overriding, Abstract classes, Dynamic method dispatch, Usage of final keyword. PACKAGES AND INTERFACES: Defining package, Access protection, importing packages, Defining and Implementing interfaces, and Extending interfaces. I / O STREAMS: Concepts of streams, Stream classes- Byte and Character stream, Reading console Input and Writing Console output, File Handling.

Unit –III: EXCEPTION HANDLING 9 Hours

Exception types, Usage of Try, Catch, Throw, Throws and Finally keywords, Built-in Exceptions, Creating own Exception classes. MULTI THREADING: Concepts of Thread,

Thread life cycle, creating threads using Thread class and Runnable interface, Synchronization, Thread priorities, Inter Thread communication.

Unit –IV: AWT CONTROLS **9 Hours**

The AWT class hierarchy, user interface components- Labels, Button, Text Components, Check Box, Checkbox Group, Choice, List Box, Panels – Scroll Pane, Menu, Scroll Bar. Working with Frame class, Colour, Fonts and layout managers. EVENT HANDLING: Events, Event sources, Event Listeners, Event Delegation Model (EDM), Handling Mouse and Keyboard Events, Adapter classes, Inner classes.

Unit –V: SWINGS **9 Hours**

Introduction to Swings, Hierarchy of swing components. Containers, Top level containers - JFrame, JWindow, JDialog, JPanel, JButton, JToggleButton, JCheckBox, JRadioButton, JLabel,JTextField, JTextArea, JList, JComboBox, JScrollPane.APPLETS: Life cycle of an Applet, Differences between Applets and Applications, Developing applets, simple applet, **JDBC:** JDBC-ODBC Bridge, The connectivity model, the driver manager, navigating the resultset object contents, java.sql Package.

Total: 45 Hours

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Reference Books:

1. Naughton &Schildt “The Complete Reference Java 2”, Tata McGraw Hill
2. Deitel “Java- How to Program:” Pearson Education, Asia
3. Horstmann & Cornell ``Core Java 2” (Vol I & II), Sun Microsystems
4. Ivan Bayross “Java 2.0”: BPB publications
5. Ivor Horton’s “Beginning Java 2, JDK 5 Ed., Wiley India.

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA
Semester-II

L-2 T-1 P-0 C-3

25MCA320T: Data Structures & Algorithm Design

Course Objectives

- To study basic concepts Data structures.
- To study stack, queue, tree and implementation.
- To study Set and graphs and their implementation.
- To learn memory management and issues.
- To implement searching, sorting and design techniques.

Course Outcomes (COs)

1. Understand the basic concepts of data structures such as arrays, linked lists etc.
2. Apply the underlying principles behind data structures and their applications.
3. Analyze to apply data structures in real-world scenarios, such as sorting, searching, and storing data efficiently.
4. Evaluate be able to analyze algorithms and data structures in terms of their time and space complexity.
5. Create the performance of different data structures and algorithms.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	1	3	-	-	2	-	1	-	2	-	-	-
CO2	2	-	3	2	-	1	-	2	-	-	-	-
CO3	1	2	-	3	-	1	-	-	-	1	-	-
CO4	-	1	-	2	3	-	2	1	-	2	-	-
CO5	2	-	2	1	-	3	-	2	1	-	-	-

High-3 Medium-2 Low-1

UNIT I: Introduction 09Hours

Definition, types of data structures, Arrays, functions, pointers, structures, Algorithm Analysis, Abstract Data type, Big-O notation, **Stack:** Stack, Operations on stack, Recursion, Polish Notation: Infix, Prefix, Postfix, Conversion from one to another using stack, Multiple Stack. **Queue:** Queue, Application of Queue, Circular Queue, Dequeue, Priority Queue.

UNIT II: Searching sorting and design techniques 09 Hours

Searching Techniques, Sorting: Internal Sorting: Bubble Sort, Insertion Sort, Quick Sort, Heap Sort, External Sorting: Merge Sort, Multi-way Merge Sort, Design Techniques: Divide and Conquer Dynamic Programming: Greedy Algorithm, Backtracking.

UNIT III: Linked Lists 09 Hours

Introduction, single linked list, representation of a linked list in memory, Operations on a single linked list, reversing a singly linked list, Stack and queue using linked list, Circular linked list and doubly linked list

UNIT IV: Trees 09 Hours

definitions-height, depth, order, degree, parent and child relationship etc.; Binary Trees-complete binary tree, almost complete binary tree; Tree traversals-preorder, in order and post order traversals, their recursive and non-recursive implementations; expression tree-evaluation; linked representation of binary tree-operations. Threaded binary trees; forests, conversion of forest into tree, Heap.

UNIT V: Graphs **09 Hours**

related definitions: graph representations- adjacency matrix, adjacency lists, adjacency multilist; traversal schemes- depth first search, breadth first search; Minimum spanning tree; shortest path algorithm; kruskals & dijkstras algorithm. Basic idea of AVL tree- definition, insertion & deletion operations; basic idea of B-tree definition, order, degree, insertion & deletion operations

Total: 45 Hours

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Reference Books

1. Data Structures and Algorithms Made Easy byNarasimha Karumanchi, CareerMonk Publications
2. Problem-solving in Data structures and Algorithms Using Java", by Hemant Jain
3. Fundamentals Of Data Structure, By S. Sawhney & E. Horowitz
4. Date Structure: By lipschuists (Schaum's outline Series Mcgraw Hill publication)
5. TennenBaum A.M. & others: Data Structures using C & C++; PHI
6. Yashwant Kanetkar, Understanding Pointers in C, BPB.

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-2 T-1 P-0 C-3

25MCA165T: Cloud Computing Concepts

Course Objectives

- To understand cloud computing fundamentals
- To learn cloud computing applications
- To learn cloud computing architecture and service management
- To learn concept of virtualization
- To learn about cloud security and risk management

Course Outcomes

1. Understand the history, vision, and characteristics of cloud computing.
2. Apply cloud computing architecture principles to design scalable and fault-tolerant cloud solutions.
3. Analyze cloud management and virtualization technologies for effective governance and disaster recovery.
4. Evaluate cloud security fundamentals and implement secure cloud software requirements.
5. Create market-based management strategies and federated cloud systems through case studies of major cloud platforms.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO 8	PO9	PO10	PSO1	PSO 2
CO1		3	-	-	2	-	1		2	-	-	-
CO2	1		3	2	-	1	-	2		-	-	-
CO3	2	1	-	3	-	1		-		1	-	-
CO4	1	-	1	2	3	-	2	1	-	2	-	-
CO5	2		2	-		3	1	2	1	-	-	-

High-3 Medium-2 Low-1

Unit-I : Introduction 9 Hours

Historical development, Vision of Cloud Computing, Characteristics of cloud computing as per NIST, Cloud computing environments, Cloud services requirements, Cloud and dynamic infrastructure. Overview of cloud applications: ECG Analysis in the cloud, Satellite Image Processing, CRM and ERP

Unit-II: Cloud Computing Architecture 9 Hours

Cloud Reference Model, Types of Clouds, Cloud Interoperability & Standards, Scalability and Fault Tolerance, Cloud Solutions: Cloud Business Process Management. Cloud Offerings: Cloud Analytics, Testing Under Cloud, Virtual Desktop Infrastructure.

Unit –III: Cloud Management & Virtualization Technology 9 Hours

Resiliency, Provisioning, Asset management, Concepts of Map reduce, Cloud Governance, High Availability and Disaster Recovery. Virtualization: Fundamental concepts of compute, storage, networking, desktop and application virtualization. Virtualization benefits, server virtualization, Block and file level storage virtualization.

Unit-IV: Cloud Security **9 Hours**

Cloud Information security fundamentals, Cloud security services, Design principles, Secure Cloud Software Requirements, Policy Implementation, Cloud Computing Security Challenges, Virtualization security Management, Cloud Computing Security Architecture.

Unit-V: Cloud Management **9 Hours**

Market Based Management of Clouds, Federated Clouds/Inter Cloud: Characterization & Definition, Cloud Federation Stack, Third Party Cloud Services. Case study: Google App Engine, Microsoft Azure, Hadoop, Amazon.

Total: 45 Hours

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Reference Books:

1. Buyya, Selvi," Mastering Cloud Computing ",TMH Pub
2. Kumar Saurabh, "Cloud Computing", Wiley Pub
3. Krutz , Vines, "Cloud Security " , Wiley Pub
4. Velte, "Cloud Computing- A Practical Approach", TMH Pub
5. Sosinsky, "Cloud Computing", Wiley Pub

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-2 T-1 P-0 C-3

25MCA167T: Mathematics for Machine Learning

Course Objectives:

- To understand basic concepts of linear algebra and matrix theory.
- To learn Matrix decomposition Algorithms.
- To apply the basic concepts of Calculus.
- To apply basic concepts of probability.
- To learn concepts of mean, mode, median etc.

Course Outcomes

1. Demonstrate comprehension of special matrices, linear equations, vector spaces, and linear transformations in linear algebra.
2. Apply concepts of norms, inner products, orthogonal projections, and rotations in analytical geometry to solve geometric problems.
3. Analyze matrix decompositions such as eigenvalues, eigenvectors, Cholesky decomposition, and singular value decomposition to understand the structure of matrices.
4. Evaluate techniques of vector calculus including differentiation, gradients, and multivariate Taylor series to solve optimization and machine learning problems.
5. Synthesize knowledge of probability theory and distributions to construct probability spaces, apply Bayes' theorem, and analyze Gaussian distributions and exponential families.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	-	2	2	3	-	-	-	-	-	-	-	-
CO2	-	-	-	-	-	-	-	-	2	-	-	-
CO3	-	1	-	-	2	-	1	-	-	-	-	-
CO4	2	-	1	-	-	3	-	-	3	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-

High-3 Medium-2 Low-1

Unit-I: Linear Algebra 9 Hours

Special Matrices and their properties, System of linear equations, Rank of a matrix. Vector spaces and subspaces, linear dependent and independent Basis and dimensions, linear transformation, linear operator, kernel and null space of a linear transformation.

Unit-II: Analytical Geometry 9 Hours

Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Orthonormal Basis, Orthogonal Complement, Inner Product of Functions, Orthogonal Projections, and Rotations.

Unit-III: Matrix Decomposition 9 Hours

Determinant and Trace, Eigenvalues and Eigenvectors, Cholesky Decomposition, Eigen decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation, Matrix Phylogeny

Unit-IV: Vector Calculus 9 Hours

Differentiation of Univariate Functions, Partial Differentiation and Gradients, Gradients of Vector-Valued Functions, Gradients of Matrices, Useful Identities for Computing Gradients, Backpropagation and Automatic Differentiation, Higher-Order Derivatives, Linearization and Multivariate Taylor Series

Unit-V: Probability and Distribution 9 Hours

Construction of a Probability Space, Discrete and Continuous Probabilities, Sum Rule, Product Rule, and Bayes' Theorem, Summary Statistics and Independence, Gaussian Distribution, Conjugacy and the Exponential Family, Change of Variables/Inverse Transform

Total 45 Hours

Reference Books:

1. Mathematics for machine learning, A. aldofaisal, Cheng soon ong, and Marc peter deisenroth, Cambridge University Press
2. Matrix Methods in Data Mining and Pattern Recognition, Lars Elden. (2016).
3. Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares, Stephen Boyd and LievenVandenberghe, Cambridge U Press (2018).
4. Probability and Statistics for Engineers and Scientists, Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying E. Ye, (9th Edition), Pearson Education (2015)
5. Pattern Recognition and Machine Learning, Christopher Bishop,Springer,(2010)
6. Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Flach, Cambridge University Press (2015)
7. Elementary Linear Algebra, Enton Howard, Wiley India (2016)
8. Introduction to Linear Algebra, Gilbert Strang, 5th ed., Cengage Learning, 2015

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-2 T-1 P-0 C-3

25MCA168T: Artificial Intelligence & Applications

Course Objectives

- To know about basic concepts Artificial Intelligence.
- To learn about Search Techniques and Knowledge Representations.
- To learn and practice Python programming.
- To know about Natural Language processing.
- To know about Probabilistic Reasoning and Uncertainty.

Course Outcomes

1. Understand the concepts of Artificial Intelligence
2. Apply the useful search techniques
3. Analyze Python Programming to program intelligent systems
4. Evaluate the strengths and weaknesses of various Natural Language Processing (NLP) methods & technologies
5. Create probabilistic reasoning models like Expert Systems, and Learning

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	2	3	1	-	-	-	-	-	-	-	-	-
CO2	-	2	3	1	-	-	-	-	2	-	-	-
CO3	-	1	2	3	-	-	1	-	-	-	-	-
CO4	1	-	2	-	3	-	-	1	-	-	2	1
CO5	-	1	-	2	-	3	-	-	-	1	1	2

High-3 Medium-2 Low-1

Unit-I: Overview of AI 9 Hours

The AI problems, what is an AI technique, Characteristics of AI applications, Problem Solving, Search and Control Strategies General problem solving, production systems, control strategies forward and backward chaining, exhaustive searches depth first breadth-first search, Search in pacemen, multi-Agent pacemen.

Unit-II: Heuristic Search Techniques 9 Hours

Hill climbing, branch and bound technique, best first search & A* algorithm, AND / OR graphs, problem reduction & AO* algorithm, constraint satisfaction problems. **Knowledge Representations:** First-order predicate calculus, Skolemization, resolution principle & unification, interface mechanisms, horn's clauses, semantic networks, frame systems and value inheritance, scripts, conceptual dependency.

Unit-III: Introduction to Python Programming 9 Hours

Data Types and Operators Syntax and numeric functions, control flow, functions, Basic Python manipulation functions, Input-output and local variables, Iteration and recursion, list, tuple, dictionary and set. Decision diagram and value of perfect information Hidden Markov Model, Practical filtering, and implementation of BFS, A* and AO* algorithm.

Unit-IV: Natural Language Processing 9 Hours

Parsing techniques, context-free grammar, case and logic grammars, semantic analysis. Game playing Minimax search procedure, alpha-beta cutoffs, additional refinements. Planning component of planning systems, goal stack planning, non-linear planning. **Reinforcement Learning:** Direct evaluation, Q- learning, model-based: RL grid. Probability- Conditional probability

Unit-V: Probabilistic Reasoning and Uncertainty **9 Hours**

Probability theory, Bayes theorem, and Bayesian networks, certainty factor. Expert Systems Introduction to expert systems and application of expert systems, various expert system shells, **Learning**: Rote learning, learning by induction, explanation-based learning.

Total Hours: 45

Reference Books:

1. Principles of Soft Computing, by S.N. Deepa S.N. Sivanandam
2. Russell, Stuart and Norvig, Peter, “Artificial Intelligence: A Modern Approach”.
3. Spivey, Michael, “An Introduction to Logic Programming”.
4. Weizenbaum, Joseph, “Computer power and human reason”.
5. Elaine Rich and Kevin Knight, “Artificial Intelligence”.
6. Dan W. Patterson, “Introduction to Artificial Intelligence and Expert Systems”.

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA
Semester-II

L-0 T-0 P-2 C-1

25MCA300P: Linux Operating System

Course Objectives

- To familiarize students with the Linux environment.
- To learn Command line operations and Linux file structure.
- To learn managing user accounts.
- To Learn Local Security Principles.
- To learn the fundamentals of shell scripting/programming.

Course Outcomes (COs)

1. Understand the components of Linux, including distributions, boot processes.
2. Apply between CLI and GUI interfaces, and proficiently navigate directories.
3. Analyze the usage of package management systems for software installation.
4. Evaluate user accounts, groups, and the root account.
5. Create Bash shell scripts incorporating conditional statements and string manipulation.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1		3	-	-	2	-	1	2	-	-	-	-
CO2	2	1	3		-	1	-	2	1	-	-	-
CO3	-	1	-	3	-	-	2	-	1		-	-
CO4	1	-		2	3	-	-	1	-	-	-	-
CO5	2	1	2	-		3	1	2		-	-	-

High-3 Medium-2 Low-1

Unit 1: Installation and Basic Configuration of Red Hat Linux

Installation of Red Hat Linux, Partitioning drives, Network configuration, Setting time zones, Creating passwords and user accounts, Shutting down, Basic Linux Commands, Understanding and using basic Linux commands.

Unit 2: User and File Management

User Management, Creating, deleting, and modifying user accounts, Changing passwords, File and Directory Management, Configure file permissions and change ownership.

Unit 3: Network Configuration and Commands

Network Configuration, Understanding network interfaces and configurations, Network Troubleshooting and Tools, Using network tools for diagnostics and troubleshooting.

Unit 4: Shell Scripting Basics

Introduction to Shell Scripting, Writing and executing basic shell scripts, Arithmetic and Logic in Shell Scripts, Performing arithmetic operations in shell scripts, Conditional statements and loops.

Unit 5: Advanced Linux Configuration and Package Management

Environment Variables and System Configuration, Configuring environment variables, Package Management, Installing and managing software packages using yum and RPM,Software selection and installation processes, Advanced Command Usage, Understanding and using the grep command.

List of Experiments

1. Installation of Red HAT Linux operating system.
   a. Partitioning drives
   b. Network configuration
   c. Setting time zones
   d. Creating password and user accounts
   e. Shutting down
2. Write basic commands of Linux with its output.
3. Create new users in the Linux operating system and also run commands to change password and delete users.
4. Study all types of ls commands with syntax.
5. Configure Following network command: ifconfig, netstate, host, arp, ipconfig, hostname, nslookup, route, dig, ping.
6. Study of Grep command in Linux with syntax.
7. Write a shell script that presents a multiple choice question, gets the users answers and reports back whether the answer is right, wrong or not one of the choices Program.
8. Write a shell script to evaluate arithmetic operations.
9. Write a shell script to calculate simple interest.
10. Write a shell script to determine if a given year is a leap year or not.
11. Write a shell script to calculate Fibonacci series.
12. Write a shell Script to determine largest among three integer numbers.
13. Configure environment variable in Linux
14. Configure Change file permission and change ownership
15. Software selection and installation.
16. Installation package yum, RPM.

Total: 30 Hours

Reference Books:

1. Linux Administration: A Beginner's Guide by Steve Shah,Wale Soyinka, ISBN 0072262591(0-07-226259-1), McGraw-Hill Education
2. Unix for programmers and users (3rd Edition)- Graham Glass & King ables,Pearson Education India.(Low Prices Edition)
3. UNIX Concepts and Applications by Sumitabha Das,Tata Mcgraw hill publications.
4. Unix and Linux System Administration Handbook, by Evi Nemeth,Garth Snyder, Trent R. Hein,Ben Whaley.
5. Linux System Administration,By Tom Adelstein,Bill Lubanovic.

List of e-Learning Resources:

1. https://nptel.ac.in/

2. https://www.coursera.org/

Prepared By	Academic Coordinator	HOD	Senior Faculty nominated by DOAA

MCA

Semester-II

L-0 T-0 P-2 C-1

25MCA310P: Object Oriented Programming using Java

Course Objectives

- To learn basic concepts of Java.
- To understand Concepts of Object-Oriented Programming in Java.
- To learn Multithreading and Exception Handling.
- Learn to develop GUI based applications.
- To learn event handling and database connectivity.

Course Outcomes

Students will able to learn

1. Understand writing basic Java programs by understanding and applying Java syntax, data types, and operators.
2. Apply control structures such as conditional statements, loops, and switch-case constructs to solve complex problems in Java.
3. Analyze arrays and data structures, including collections framework classes like ArrayList, LinkedList, and PriorityQueue, to handle and process data efficiently.
4. Evaluate object-oriented programs in Java, including the use of classes, objects, inheritance, polymorphism, and constructors, to create modular and reusable code.
5. Create advanced Java concepts such as interfaces, abstract classes, wrapper classes, and packages to develop robust and maintainable Java applications.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	-	3	-	-	2	-	1		2	-	-	-
CO2	1	-	3	2	-	1	-	2	-	-	-	-
CO3	2	1	-	3	-	1	-	-	-	1	-	-
CO4	1	-	1	2	3	-	2	1	-	2	-	-
CO5	2	-	2	-	-	3	1	2	1	-	-	-

High-3 Medium-2 Low-1

Unit 1: Introduction to Java Programming

Basic Java Syntax and Structure, Writing and running simple Java programs, Variables and Data Types, Understanding different data types and variable declaration.

Unit 2: Control Structures

Conditional Statements, If-else and nested if-else statements, Looping Constructs, For loop, while loop, and do-while loop, Switch-Case and Break/Continue, Using switch-case for multiple conditions, Understanding break and continue.

Unit 3: Arrays and Data Structures

Arrays, Declaring, initializing, and using arrays, Collections Framework, Introduction to Java

Collections.

Unit 4: Object-Oriented Programming

Classes and Objects, Defining and using classes and objects, Constructors
Default, parameterized, and copy constructors, Constructor overloading, Inheritance and Polymorphism, Single-level and multilevel inheritance, Method overriding and using super keyword.

Unit 5: Advanced Java Concepts

Interfaces and Abstract Classes, Defining and implementing interfaces, Wrapper Classes and Type Conversion, Using wrapper classes, Implicit and explicit type conversion, Extending interfaces, Packages and Modules, Creating and using packages, Modular programming in Java.

List of Experiments

1. Program to print "Hello World" on the output screen.
2. Program to perform addition of two numbers.
3. Program to perform swapping using another variable.
4. Program to perform swapping without using another variable.
5. Program to find the greatest number among three numbers using "nested if-else".
6. Program to print table of "2" using for loop.
7. Program to print reverse of a given number.
8. Program to calculate factorial using "while loop".
9. Program to implement calculator using "switch...case".
10. Program to implement the concept of "Break…. continue".
11. Program to implement the concept of "Do…while" loop.
12. Program to check whether the number is palindrome or not.
13. Program to find sum of elements stored in array.
14. Program to sort elements of an array.
15. Program to perform implicit and explicit conversion in java.
16. Program to implement concept of class and object.
17. Program to perform operation on packages.
18. Program to implement constructor and parameterized constructor.
19. Program for copy constructor.
20. Program for parameterized constructor (constructor overloading).
21. Program to implement single level, multilevel inheritance.
22. Program to implement overriding.
23. Program to implement the concept of super used for function and constructor.
24. Program for implementing and applying interfaces in java.
25. Program for variables in interfaces and extending interfaces.
26. Program to implement wrapper classes in java.
27. Program to implement array list in collection framework.
28. Program to implement vectors in the collection framework.
29. Program to implement Linked List in collection framework.
30. Program to implement priority queue in collection framework.

Total: 30 Hours

List of e-Learning Resources:

1. https://nptel.ac.in/

2. https://www.coursera.org/

Reference Books:

1. Naughton &Schildt “The Complete Reference Java 2”, Tata McGraw Hill
2. Deitel “Java- How to Program:” Pearson Education, Asia
3. Horstmann& Cornell ``Core Java 2” (Vol I & II), Sun Microsystems
4. Ivan Bayross “Java 2.0”: BPB publications
5. Ivor Horton’s “Beginning Java 2, JDK 5 Ed., Wiley India.

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-0 T-0 P-2 C-1

25MCA320P: Data Structures & Algorithm Design

Course Objectives

- To study basic concepts Data structures.
- To study stack, queue, tree and implementation.
- To study Set and graphs and their implementation.
- To learn memory management and issues.
- To implement searching, sorting and design techniques.

Course Outcomes (COs)

Students will able to learn

1. Understand basic data structure operations in C++, including array insertion and deletion, stack, and queue operations.
2. Apply algorithms for matrix operations, such as addition, subtraction, multiplication, and transpose, to perform complex data manipulations.
3. Analyze linked lists, including insertion and deletion at various positions.
4. Evaluate sorting and searching algorithms, such as quick sort, bubble sort, and binary search.
5. Create tree and graph traversal techniques, including Kruskal's and Prim's algorithms for minimum spanning trees.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	1	3	-	-	2	-	1	-	2	-	-	-
CO2	2	-	3	2	-	1	-	2	-	-	-	-
CO3	1	2	-	3	-	1	-	-	-	1	-	-
CO4	-	1	-	-	3	-	2	1	-	2	-	-
CO5	2	-	-	1	-	3	-	2	1	-	-	-

High-3 Medium-2 Low-1

Unit 1: Array Operations and Matrix Manipulation

Basic Array Operations, Insertion and deletion in an array, Matrix Operations, Addition, subtraction, multiplication, and transpose of matrices.

Unit 2: Stack and Queue Implementations

Stack Operations, Implementing stack using arrays, Push and Pop operations,Queue Operations, Implementing queue using arrays, Insertion and deletion operations.

Unit 3: Linked Lists

Basic Operations on Singly Linked Lists, Creating nodes and basic operations, Insertion and deletion at various positions.

Unit 4: Searching and Sorting Algorithms

Searching Algorithms, Linear search and binary search,Sorting Algorithms, Various sorting techniques and their implementations.

Unit 5: Advanced Data Structures and Algorithms

Tree and Graph Traversal, Tree traversal techniques (preorder, inorder, postorder), Graph traversal techniques (BFS, DFS).

List of Experiments:

1. Write a program to implement insertion () and deletion () operations in an array.
2. Write a program for addition, subtraction, multiplication and transpose of matrix.
3. Write a program to implement templates in C++.
4. Write a program to implement Push () and Pop () operations in a stack using array.
5. Write a program to convert an expression from infix to postfix.
6. Write a program to implement insertion () and deletion () operations in a queue using array.
7. Write a program to create 5 nodes in a singly linked list.
8. Write a program to insert an element at the end, beginning and at the end position in a single linked list.
9. Write a program to delete an element at the end, beginning and at the end position in a single linked list.
10. Write a program to implement stack using linked list.
11. Write a program to implement Queue using a linked list.
12. Write a program to search an element in an array using linear search and binary search.
13. Write a program to sort an array using insertion sort.
14. Write a program to sort an array using bubble sort.
15. Write a program to sort an array using selection sort.
16. Write a program to sort an array using quick sort.
17. Write a program to implement tree traversal technique.
18. Write a program to implement graph traversal technique.
19. Write a program to implement kruskal’s and prim’s algorithm

Total: 30 Hours

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Reference Books

1. Data Structures and Algorithms Made Easy byNarasimha Karumanchi, CareerMonk Publications
2. Problem-solving in Data structures and Algorithms Using Java", by Hemant Jain
3. Fundamentals Of Data Structure, By S. Sawhney & E. Horowitz
4. Date Structure: By lipschuists (Schaum's outline Series Mcgraw Hill publication)
5. TennenBaum A.M. & others: Data Structures using C & C++; PHI
6. Yashwant Kanetkar, Understanding Pointers in C, BPB.

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-0 T-0 P-2 C-1

25MCA165P: Cloud Computing Concepts

Course Objectives

- To understand cloud computing fundamentals
- To learn cloud computing applications
- To learn cloud computing architecture and service management
- To learn concept of virtualization
- To learn about cloud security and risk management

Course Outcomes

Students will be able to

1. Understand the virtual machines with installation, configuration, and snapshot management across different platforms.
2. Apply AWS services effectively, including EC2 instance management and IAM for security.
3. Analyze Host websites and manage AMIs on AWS, including copying and sharing AMIs between regions.
4. Evaluate cloud storage solutions using AWS S3 for file management and bucket operations.
5. Create containerized applications with Docker, including installation, container management, and custom image creation.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	-	3	-	-	2	-	1		2	-	-	-
CO2	1	-	3	2	-	1	-	2	-	-	-	-
CO3	2	1	-	3	-	1	-	-	-	1	-	-
CO4	1	-	1	2	3	-	2	1	-	2	-	-
CO5	2	-	2	-	-	3	1	2	1	-	-	-

High-3 Medium-2 Low-1

Unit 1:Virtual Machine Management

Virtual Machine Setup, Downloading and installing Virtual Machine software (VirtualBox, VMware, KYM), Basic VM Operations,Starting, restarting, and powering off virtual machines, Editing virtual machine hardware configurations, VM Images and Snapshots, Creating and using image snapshots, Importing and exporting virtual machine images.

Unit 2: AWS Fundamentals

AWS Account Setup, Creating an AWS free tier account, Introduction to Identity and Access Management (IAM), IAM and Security, Creating users and groups, Authorization via policies.

Unit 3: EC2 Instances

Launching and Accessing EC2 Instances, Launching EC2 instances running Linux and Windows SSH and RDP access methods,EC2 Applications and Customization,Hosting websites on EC2 instances, Creating and managing custom AMIs.

Unit 4: AWS S3 and Storage

S3 Bucket Management, Creating and managing S3 buckets, Uploading and downloading files from S3.

Unit 5: Docker and Containerization

Docker Basics,Installing Docker on EC2 instances, Creating and managing Docker containers, Docker Images, Pulling and pushing Docker images from Docker Hub, Creating custom Docker images.

List of Experiments:

1. Download and Install Virtual Machine (Virtual Box, VMware and KYM)
2. Installing Virtual Machine
3. Controlling Virtual Machine (Start, restart, power off)
4. Editing Virtual Machine Hardware
5. Creating and Using Image snapshot
6. Importing and Exporting Virtual Machine images
7. Create AWS free tier account
8. Introduction to IAM
9. Creating a Use rand Group
10. Authorization via Policies
11. Creating and Attaching Policies
12. Launching an EC2running Linux
13. How to ssh into EC2using Linux/Windows
14. Launching an EC2 running Windows
15. Connect Windows Instance using RDP
16. Hosting Website on EC2 Instance
17. Create AWS Custom AMI
18. Copy AMI from one region to another
19. Share AMI with AWS account
20. Create S3 Bucket
21. Upload/Download files from S3Bucket
22. Containerized Application Using Docker container
23. Install docker on EC2 Instance
24. Creating and managing Docker containers
25. Pull and push docker images from docker hub
26. Creating Docker custom Images

Total: 30 Hours

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Reference Books:

1. Buyya, Selvi," Mastering Cloud Computing ",TMH Pub
2. Kumar Saurabh, "Cloud Computing", Wiley Pub
3. Krutz , Vines, "Cloud Security " , Wiley Pub
4. Velte, "Cloud Computing- A Practical Approach", TMH Pub
5. Sosinsky, "Cloud Computing", Wiley Pub

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-0 T-0 P-2 C-1

25MCA167P: Mathematics for Machine Learning

Course Objectives:

- To understand basic concepts of linear algebra and matrix theory.
- To learn Matrix decomposition Algorithms.
- To apply the basic concepts of Calculus.
- To apply basic concepts of probability.
- To learn concepts of mean, mode, median etc.

Course Outcomes

Students will able to

1. Understand proficiency in R programming fundamentals, including syntax, data types, and basic data structures like vectors and matrices.
2. Apply mathematical and statistical operations in R, utilizing functions for numerical computations, prime number generation, and random data sampling.
3. Analyse advanced matrix operations, including creation, manipulation, and transformation, as well as extraction of submatrices and rotation techniques.
4. Evaluate R for data import/export and visualization, effectively handling various data formats and creating customized plots for data representation.
5. Create advanced R programming concepts, such as custom functions, recursion, and data aggregation, for comprehensive data analysis and manipulation.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO 8	PO 9	PO10	PSO1	PSO 2
CO1	-	2	2	3	-	-	-	-	-	-	-	-
CO2	-	-	-	-	-	-	-	-	2	-	-	-
CO3	-	1	-	-	2	-	1	-	-	-	-	-
CO4	2	-	1	-	-	3	-	-	3	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-

High-3 Medium-2 Low-1

Unit 1: Basic R Programming and Data Structures

Introduction to R Programming, Installation and Setup, Basic Syntax and Commands, Data Types and Structures, Vectors: Creation and Manipulation, Matrices: Creation and Operations, Lists and Data Frames.

Unit 2: Mathematical and Statistical Operations

Numerical Computations, Sequences and Arithmetic Operations, Mathematical Functions (sum, mean, product), Prime Numbers and Factorial Calculations, Random Data Generation, Generating Random Numbers, Normal Distribution and Random Sampling.

Unit 3: Matrix Operations and Manipulations

Matrix Operations, Matrix Creation and Labeling, Basic Matrix Operations (Addition, Subtraction, Multiplication, Division), Submatrix Extraction and Matrix Rotation, Advanced Matrix Techniques, Matrix Transposition, Converting Matrices to Arrays.

Unit 4: Data Import, Export, and Visualization

Data Import and Export, Reading and Writing CSV Files, Handling Different Data Formats, Data Visualization, Basic Plotting Functions (bar plots, line plots), Customizing Plots (axes limits, labels).

Unit 5: Advanced R Programming Concepts

Working with Functions, Creating and Using Custom Functions, Recursion and Iteration, Data Analysis and Manipulation, Data Aggregation and Summary (sum, mean), Handling and Analyzing Factors, Data Type Conversion and Manipulation.

List of Experiments:

1. Write a R program to create sequence of numbers from 20 to 50 and find mean of numbers from 20 to 50 and sum of numbers from 51 to 91.
2. Write a R program to create a vector which contains 10 random integer values between -50 and +50.
3. Write a R program to get the first 10 Fibonacci numbers.
4. Write a R program to get all prime numbers up to a given number (based on the sieve of Eratosthenes).
5. Write a R program to read the .csv file and display the content.
6. Write a R program to find the factors of a given number.
7. Write a R program to find the maximum and the minimum value of a given vector.
8. Write a R program to create three vectors a, b, c with 3 integers. Combine the three vectors to become a 3×3 matrix where each column represents a vector. Print the content of the matrix.
9. Write a R program to create a list of random numbers in normal distribution and count occurrences of each value.
10. Write a R program to create three vectors numeric data, character data and logical data. Display the content of the vectors and their type.
11. Write a R program to create a 5 x 4 matrix, 3 x 3 matrix with labels and fill the matrix by rows and 2 × 2 matrix with labels and fill the matrix by columns.
12. Write a R program to draw an empty plot and an empty plot specify the axes limits of the graphic.
13. Write a R program to create a simple bar plot of five subjects' marks.
14. Write a R program to compute sum, mean and product of a given vector elements.
15. Write a R program to create a blank matrix.
16. Write a R program to create a matrix taking a given vector of numbers as input. Display the matrix.
17. Write a R program to create two 2x3 matrix and add, subtract, multiply and divide the matrixes.
18. Write a R program to extract the submatrix whose rows have column value > 7 from a given matrix.
19. Write a R program to rotate a given matrix 90-degree clockwise rotation.
20. Write a R program to convert a matrix to a 1-dimensional array

Total: 30 Hours

Reference Books:

1. Mathematics for machine learning, A. aldofaisal, Cheng soon ong, and Marc peter deisenroth, Cambridge University Press
2. Matrix Methods in Data Mining and Pattern Recognition, Lars Elden. (2016).
3. Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares, Stephen Boyd and LievenVandenberghe, Cambridge U Press (2018).
4. Probability and Statistics for Engineers and Scientists, Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying E. Ye, (9th Edition), Pearson Education (2015)
5. Pattern Recognition and Machine Learning, Christopher Bishop,Springer,(2010)
6. Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Flach, Cambridge University Press (2015)
7. Elementary Linear Algebra, Enton Howard, Wiley India (2016)
8. Introduction to Linear Algebra, Gilbert Strang, 5th ed., Cengage Learning, 2015

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-0 T-0 P-2 C-1

25MCA168P: Artificial Intelligence & Applications

Course Objectives

- To know about basic concepts Artificial Intelligence.
- To learn about Search Techniques and Knowledge Representations.
- To learn and practice Python programming.
- To know about Natural Language processing.
- To know about Probabilistic Reasoning and Uncertainty.

Course Outcomes

Students will able to learn

1. Understand the concepts of Artificial Intelligence
2. Apply the useful search techniques
3. Analyze Python Programming to program intelligent systems
4. Evaluate the strengths and weaknesses of various Natural Language Processing (NLP) methods & technologies
5. Create probabilistic reasoning models like Expert Systems, and Learning

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	2	3	1	-	-	-	-	-	-	-	-	-
CO2	-	2	3	1	-	-	-	-	2	-	-	-
CO3	-	1	2	3	-	-	1	-	-	-	-	-
CO4	1	-	2	-	3	-	-	1	-	-	2	1
CO5	-	1	-	2	-	3	-	-	-	1	1	2

High-3 Medium-2 Low-1

Unit 1: Basic Python Syntax and Functions

Python Syntax and Numeric Functions, Basic syntax rules and numeric operations, Data Structures, List, tuple, and dictionary manipulation.

Unit 2: Control Flow and Recursion

Predicates and Conditionals, Using predicates and conditional statements, Input-Output and Local Variables, Handling input and output, understanding local variables, Iteration and Recursion, Implementing iterative and recursive solutions.

Unit 3: Problem Solving and Algorithms

Algorithmic Problem Solving, Solving problems using algorithms and decision diagrams, Classic Problems, Studying and solving classic problems.

Unit 4: Programming Challenges

String and Number Manipulations, Working with strings and numbers, Calculator and Percentage Calculation, Implementing basic calculators and percentage calculations.

Unit 5: Game Development and Advanced Concepts

Game Development, Implementing simple games, Advanced Theoretical Concepts, Understanding Bayes theorem and Bayesian networks.

List of Experiments

1. Syntax and numeric functions.
2. List, tuple, and dictionary manipulation functions.
3. Predicates and conditionals.
4. Input-output and local variables.
5. Iteration and recursion.
6. Property list and array.
7. Decision diagrams.
8. Study about water jug problem.
9. Program for factorial using recursion.
10. Program for reverse a string.
11. Program for swapping of two numbers.
12. Program for word count.
13. Program to check entered number is palindrome or not.
14. Program to compare two numbers.
15. Program for calculator.
16. Program for calculating the percentage of students and making division accordingly.
17. Algorithm of “Travelling salesman problem” with the solution.
18. Program for “Tic Tac Toe”.
19. Study of 8-puzzle problem.
20. Bayes theorem and Bayesian network.

Total Hours 30

Reference Books:

1. Principles of Soft Computing, by S.N. Deepa S.N. Sivanandam
2. Russell, Stuart and Norvig, Peter, “Artificial Intelligence: A Modern Approach”.
3. Spivey, Michael, “An Introduction to Logic Programming”.
4. Weizenbaum, Joseph, “Computer power and human reason”.
5. Elaine Rich and Kevin Knight, “Artificial Intelligence”.
6. Dan W. Patterson, “Introduction to Artificial Intelligence and Expert Systems”.

List of e-Learning Resources:

1. https://nptel.ac.in/
2. https://www.coursera.org/

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**

MCA

Semester-II

L-0 T-0 P-8 C-4

25MCA390P: Front-end Development Lab

Course Objectives:

- Understand the fundamental concepts of ReactJS and its significance in modern web development.
- Gain proficiency in writing JSX code to create dynamic and interactive user interfaces in ReactJS.
- Implement event handling in React components to respond to user actions appropriately.
- Utilize React features like lists, keys, refs, and fragments to optimize rendering and enhance user experience.
- Integrate React with Bootstrap to leverage pre-built UI components and enhance design aesthetics.

Course Outcomes:

Students will able to learn

1. Understand the significance of ReactJS and its relevance in modern web development.
2. Apply styling techniques to React components.
3. Analyze the architecture of ReactJS and identify its core principles and components.
4. Evaluate the features of ReactJS and compare its capabilities with Native React and AngularJS.
5. Create React applications using appropriate deployment strategies and tools to make them accessible to users.

Articulation Matrix

(Program Articulation Matrix is formed by the strength of the correlation of COs with POs and PSOs. The strength of correlation is indicated as 3 for substantial (high), 2 for moderate (medium) correlation, and 1 for slight (low) correlation)

CO/PO/PSO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2
CO1	2	3	1	-	-	-	-	-	-	-	-	-
CO2	-	2	3	1	-	-	-	-	2	-	-	-
CO3	-	1	2	3	-	-	1	-	-	-	-	-
CO4	1	-	2	-	3	-	-	-	-	-	2	1
CO5	-	1	-	2	-	3	-	-	-	-	1	2

High-3 Medium-2 Low-1

UNIT-I Getting Started with ReactJS 24 Hours

ReactJS introduction, why to learn ReactJS, React Environment Setup- Prerequisite for ReactJS, Ways to install ReactJS, ReactJS – Architecture, ReactJS - Creating a React Application, React create-react-app, Features of ReactJS, ReactJS vs Native React, ReactJS vs AngularJS

UNIT-II ReactJS Fundamentals 24 Hours

ReactJS – JSX, ReactJS – Components: Creating a React component,Creating a class component, Creating a function component, ReactJS – Styling, ReactJS - Properties (props), React Props Validation

UNIT-III Deep Dive into ReactJS 24 Hours

ReactJS state management, ReactJS event Management, React Constructor, React Component API, React Component Life-Cycle, React Forms and user input, Controlled Component, Un-Controlled Component, Form link.

UNIT-IV Exploring Advanced ReactJS Features 24 Hours

ReactJS - Http Client Programming, React Lists, The map() function, React Keys, React Refs,React Fragments, React Router, ReactCSS,React Animation, React Date picker, DOM in React.

UNIT-V Next-Level React Development 24 Hours

React AJAX call- HTTP GET request, HTTP GET Request and Looping through data, React Bootstrap, React Table, React Hooks, React building and deployment.

List of Practical

1. Setting up a basic ReactJS environment using Node.js and npm
2. Installing ReactJS using npm and Yarn
3. Exploring the folder structure of a ReactJS application created with create-react-app
4. Creating a simple React component and rendering it to the DOM
5. Understanding the virtual DOM and its role in ReactJS architecture
6. Contrasting ReactJS with AngularJS in terms of syntax, architecture, and ecosystem
7. Writing JSX code and understanding its similarities to HTML
8. Creating functional components in ReactJS
9. Implementing class components with state and lifecycle methods
10. Styling React components using inline styles, CSS files, and CSS-in-JS libraries like styled-components
11. Passing data between components using props
12. Validating props to ensure data consistency and type safety
13. Managing component state using setState() method and understanding its asynchronous nature
14. Handling events like onClick, onChange, etc., in React components
15. Exploring the constructor method and its usage in class components
16. Accessing and manipulating the React component API methods
17. Understanding the React component lifecycle by logging lifecycle methods
18. Implementing forms in React and handling user input using controlled and uncontrolled components
19. Creating form links to navigate between different parts of the application
20. Making HTTP GET requests to fetch data from an external API
21. Rendering dynamic lists of data using the map() function
22. Managing keys to optimize list rendering performance
23. Utilizing React refs to access and manipulate DOM elements directly
24. Grouping elements using React fragments for cleaner JSX syntax
25. Implementing client-side routing using React Router for navigation

26. Applying CSS styles and animations to React components using ReactCSS and CSS-in-JS libraries
27. Integrating a date picker component into a React application for selecting dates
28. Manipulating the DOM in React using various techniques
29. Design and implement a simple ReactJS program to display “Hello World!”
30. Design and implement a search filter in ReactJS

Total 120 Hours

References Books:

1. Learning React: Functional Web Development with React and Redux 1st Edition by Alex Banks.
2. The Road to React: Your journey to master plain yet pragmatic React.js by Robin Wieruch
3. React.js Essentials: A fast-paced guide to designing and building scalable and maintainable web apps with React.js by Artemij Fedosejev
4. Full-Stack React Projects: Learn MERN stack development by building modern webapps using MongoDB, Express, React, and Node.js, 2nd Edition Paperback by Shama Hoque
5. React.js Book: Learning React JavaScript Library from Scratch by Greg Sidelnikov

List of e-Learning Resources:

1.https://nptel.ac.in/
2.https://www.coursera.org/

Prepared By	**Academic Coordinator**	**HOD**	**Senior Faculty nominated by DOAA**