School of Computer & Information Sciences

M.Tech (Artificial Intelligence)

SCHOOL OF COMPUTER & INFORMATION SCIENCES

Vision Statement:

- To invent, create and bring computing technology solutions to the common man, to the privileged and underprivileged sections of India, to bridge the digital divide and eradication of computer ignorance and digital illiteracy and to build a prosperous and technologically advanced nation.

Mission Statements:

MS-1: To pursue academic and research excellence, nationally and internationally

MS-2: To provide training, advisory, and consultancy to all the stakeholders.

MS-3: To lead the efforts in creative and newer modes of instruction delivery & supervision

School of Computer and Information Sciences

Name of the Academic Program: M.Tech (Artificial Intelligence)

Program Educational Objectives (PEOs)

PEO-1: Produce Post graduates who can contribute to the Research & Development effectively

PEO-2: To provide students a deep insight into cutting edge technologies and tools.

PEO-3: To create globally competent technocrat's with exposure to Scientific & Engineering aspects of development

PEO-4: To work collaboratively on multi-disciplinary projects and exhibit high levels of professional & ethical values

PEO-5: Create awareness of societal problems and its impact

Mapping Program Educational Objectives (PEOs) with Mission Statements (MS)

	MS-1	MS-2	MS-3
PEO-1	3	2	1
PEO-2	2	3	1
PEO-3	3	2	1
PEO-4	2	1	3
PEO-5	1	2	3

Write '3' in the box for 'high-level' mapping, 2 for 'Medium-level' mapping, 1 for 'Low-level' mapping.

School of Computer & Information Sciences

Name of the Academic Program: M.Tech(Artificial Intelligence)

Program Outcomes (POs)

PO-1: To independently carry out research/investigation and development work to solve practical problems

PO-2: To be able to write and present a substantial technical report/document

PO-3: To demonstrate knowledge and understanding of engineering principles and apply the same in solving the problems faced by society.

PO-4: To create, select, learn and apply appropriate techniques, resources, and advanced tools, including modeling and prediction with an understanding of limitations

PO-5: To recognize the opportunities and contribute to collaborative-multidisciplinary scientific research to achieve common goals.

PO-6: To acquire professional and intellectual integrity, professional ethics code of conduct and understand the responsibility to contribute to the society for sustainable development

Program Specific Outcomes (PSOs)

PSO-1: To apply artificial intelligence towards problem solving, inference, perception, knowledge representation and learning.

PSO-2: To investigate applications of AI techniques in intelligent agents, experts systems and other machine learning models.

PSO-3: Experiment learning models to develop intelligent systems that can solve the complex problems.

Mapping of Program Outcomes (POs) and Program Specific Outcomes (PSOs) with Program Educational Objectives (PEOs)

	PEO-1	PEO-2	PEO-3	PEO-4	PEO-5
PO-1	3		1	2	
PO-2	2	1		3	
PO-3	1	3		2	
PO-4	2		1	3	
PO-5			1	3	2
PO-6		1	2		3
PSO-1			1	3	2
PSO-2	1		3		2
PSO-3	3			1	2

Mapping of Program Specific Outcomes (PSOs) where applicable.

Write '3' in the box for 'high-level' mapping, 2 for 'Medium-level' mapping, 1 for 'Low-level' mapping.

School of Computer & Information Sciences
M.Tech (Artificial Intelligence)
Scheme

I-Semester					
Core			Optional Core/Elective		
Code	**Course Title**	**Credits**	**Code**	**Course Title**	**Credits**
CS401	Advanced Operating Systems	4	AI421	Data Mining	3
CS402	Algorithms	4	AI422	Image Processing	3
	Optional core-I	¾	AI423	Knowledge Representation & Reasoning	3
	Optional core-II	¾	AI424	Neural Networks	3
	Elective -I	¾	AI425	Pattern Recognition	3
CS403	*IT Lab –I*	2	AI426	Problem Solving Methods	3
CS404	*DS & Programming Lab*	2	AI427	Simulation & Modelling	3
		21/24			

III-Semester					
Code	**Course Title**	**Credits**			
	Project	6			

School of Computer and Information Sciences

Name of the Academic Program: M.Tech(Artificial Intelligence) (M.Tech-1)

Course Code: CS401 Title of the Course: Advanced Operating Systems
L-T-P : 4-0-0 Credits : 4

Prerequisite Course / Knowledge (If any): -- Basic OS course

Course Outcomes (COs)

After completion of this course successfully, the students will be able to:

- CO1: Discuss the ways system calls work.(Understand)
- CO2: Develop basic process management tasks such as scheduling, deadlock avoidance algorithms. (Create)
- CO3: Develop paging algorithm.(Create)
- CO4: Construct simple device drivers. (Create)
- CO5: Describe different file systems in existence and learn the pros and cons of the various systems. (Understand)
- CO6: Examine real world OS scheduling algorithms such as those used in Linux and Windows. (Analyze)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1				3					
CO2			3	2					
CO3	1			3	2				
CO4			2	3					
CO5		2		3					
CO6	2					3			

Detailed Syllabus:

UNIT - I: Introduction and Operating System Structures

Operating Systems Functionality, Computer Organization and Architecture, OS Operations, Kernel Data Structures, OS Services, User interfaces to OS, Programmer interfaces to OS, OS Structure, System Boot.

UNIT - II: Process and Thread Management

Process Concept, Process operations, Process Scheduling, Extended Process State Diagram, Process Context Switch in detail; Inter process Communication: Pipes, Named Pipes, Shared Memory; Process Synchronization: Signals, Mutexes, Semaphores, Monitors; Thread Management: thread creation, thread scheduling, thread synchronization; Deadlocks: Resource Allocation Graphs, deadlock detection, prevention and avoidance, recovery from deadlock.

UNIT - III: Memory Management

Memory allocation techniques: paging and segmentation, Swapping, structure of the page table; Virtual memory: demand paging, copy-on-write, Page replacement, allocation of frames, kernel memory allocation, thrashing, memory-mapped files, Translation-Lookaside Buffer (TLB).

UNIT - IV: File System Management

Disk management: formatting, boot block, swap-space management, RAID structure; Disk scheduling algorithms: elevator, C-SCAN; File concept, Access methods, Directory structure, file sharing, protection, file system structure; file system implementation: file system metadata storage structures such as inode, allocation methods, free space management, efficiency and performance including disk cache and recovery from failures.

UNIT - V: I/O Management

I/O devices: polling, interrupt-driven, DMA; Application I/O interface: character and block devices, network devices; clocks and timers, nonblocking and asynchronous I/O, vectored I/O; Kernel I/O interface: I/O scheduling, Buffering, Caching.

Reference Books

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne. Operating System Concepts, 9th edition, Wiley.
2. Charles Crowley. Operating Systems: A Design-Oriented Approach, Prentice-Hall India.
3. W. Richard Stevens, . Advanced Programming in Unix Environment, Pearson Education.
4. W. Richard Stevens. Unix Network Programming, vol. 2, Pearson Education.
5. William Stallings. Operating Systems: Internals and Design Principles, Pearson Education.
6. Maurice J. Bach. The Design of the Unix Operating System, Prentice-Hall India.
7. Robert Love. Linux Kernel Development, Pearson Education.
8. Thomas Anderson and Michael Dahlin. Operating Systems: Principles and Practice, 2nd edition, Recursive Books.

School of Computer and Information Sciences

Name of the Academic Program: M.Tech((Artificial Intelligence) (M.Tech-I)

Course Code: CS402 Title of the Course: Algorithms
L-T-P: 4-0-0 Credits : 4

Prerequisite Course / Knowledge (If any): Data Structures in under graduate level, discrete mathematical structures, knowledge of sorting algorithms and basic search strategies

Course Outcomes (COs)

After completion of this course successfully, the students will be able to:

- CO-1: Assess the inherent structure/hardness of a problem (Evaluate)
- CO-2: Select an appropriate strategy to solve a problem (Understand)
- CO-3: Design an algorithm that suits the time complexity requirements of the problem. (Create)
- CO-4: Estimate the time and space complexities of an algorithm along with the necessary mathematical proofs when necessary. (Evaluate)
- CO-5: Devise algorithms by choosing appropriate data structures (Create)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1	2			3					1
CO2				3		1		2	
CO3			3					2	
CO4		3			2				1
CO5		2		3					1

Detailed Syllabus

UNIT-I: Analysis of Algorithms: Asymptotic Notation; Best, worst and average case analysis of algorithms; Solving recurrence relations using substitution method, generating functions, Master's theorem etc. Warm-up to complexity analysis: Heap data structure, priority queue application, Best, worst and average case analysis of a few sorting algorithms like heap sort, insertion, bubble, selection, counting and radix sort algorithms. Strategies for problem solving

UNIT-II: Divide and Conquer strategy: Time complexity analysis for Merge Sort and Quick Sort Algorithms

UNIT-III: Greedy strategy: Theoretical foundation of greedy strategy: Matroids Algorithms for solving problems like Knapsack Problem (Fractional), Minimum Spanning Tree problem; Shortest Paths, Job Scheduling, Huffman's code etc along with proofs of corrections and complexity analysis

UNIT-IV: Dynamic Programming strategy: Identify situations in which greedy and divide and conquer strategies may not work. Understanding of optimality principle. Technique of memorization. Applications to problems like Coin change, 0/1 and 0/n- Knapsack, Shortest Paths, Optimal Binary Search Tree (OBST), Chained Matrix Multiplication, Traveling Salesperson Problem (TSP) etc.

UNIT-V: Backtracking and Branch & Bound strategies: State space tree construction, traversal techniques and solving problems like 0/1 and 0/n knapsack, TSP, Applications of Depth First Search: Topological sorting, Finding strongly connected components and game problems.

UNIT-VI: Theory of NP-Completeness: Complexity classes of P, NP, NP-Hard, NP-Complete, Polynomial reductions, Cook's theorem. Discussion of problems: Satisfiability(SAT), CNF-SAT, Min-Vertex Cover, Max-Clique, Graph Coloring, NP-Completeness proofs.

Reference Books:

1. Introduction to Algorithms-T.Cormen, C.E.Leiserson, R.L.Rivest, PHI, 3rdEdition 2009.
2. Algorithms- R.Johnsonbaugh and M.Schaefer, Pearson, 2004.
3. Fundamentals of Algorithmics - G.Brassard and P.Bratley, PH, 1996
4. The Algorithm Design Manual- Steven S. Skiena, Springer, 2009

School of Computer and Information Sciences

Name of the Academic Program: M.Tech (Artificial Intelligence) (MTech-I)

Course Code: CS404 Title of the Course: Data Structures & Programming Lab

L-T-P: 0-0-3 Credits : 2

Prerequisite Course / Knowledge (If any): Programming and Data Structures at under graduate level

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1: Solve a problem by choosing appropriate data structures in C programming language (Apply)
- CO-2: Select suitable data structure for an idea and propose solution using C Programming Language (Analyze)
- CO-3: Analyze the time taken to solve the problem by using C programming language (Analyze)
- CO-4: Assess the solution in terms of efficiency, modularity and well-documented programs in C under Linux environment (Evaluate)

Mapping of Course Outcomes (COs) with Program Outcomes (POs)and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1	3		1						2
CO2			1	3				2	
CO3		3						2	
CO4			2		1			3	

Detailed Syllabus:

Detailed Syllabus:

UNIT-I: Implementing Stacks and types of Queues as dynamic data structures using linked lists and their applications

UNIT-II: Binary Tree, Binary search trees & Traversals of BST, balanced trees - AVL Trees and their applications

UNIT-III: Sorting Techniques, Basic Searching Techniques, Hashing-Collision Resolution and closed hashing.

UNIT-IV: Graphs: Representations (Matrix and Adjacency List), basic traversal techniques: Depth First Search , Breadth First Search, Implementation of Kruskal Algorithm, Dijkstra Algorithm, Spanning and Minimal Spanning Trees.

UNIT-V: Multi link Structures, B Trees and B+ Trees and their applications.

Reference Books:

1. Horowitz, E., and Sahni.S: Fundamentals of Data structures. Computer Science Press, 1978.
2. Tanenbaum, A.M., and Augenstein, M.J.: Data Structures with Pascal, Prentice - Hall International, 1985.
3. Stubbas, D.: Data Structures with Abstract Data Types and Modula2, Brooks & Cole Pub. Co. 1987.
4. Trembley & Sorenson: An Introduction to Data Structures with Applications; Tata McGraw Hill
5. Kruse, R.L., Leung, B.P., and tondo, C.L.: Data Structures and Program Design in C; Prentice-Hall of India 1999.

School of Computer and Information Sciences

Name of the Academic Program: M.Tech (Artificial Intelligence)(M.Tech-I)

Course Code: AI421 Title of the Course: Data Mining

L-T-P : 3-0-0 Credits : 3

Prerequisite Course / Knowledge (If any):

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1: Explain the need of data mining and its applications (Understand)
- CO-2: Apply preprocessing techniques on raw data (Apply)
- CO-3: Practice the supervised, unsupervised & anomaly detection techniques (Apply)
- CO-4: Apply appropriate algorithms to build the analytical applications (Analyze)
- CO-5: Use the interesting patterns from humongous data to solve Association problems (Apply)
- CO-6: Evaluate the performance of different data-mining algorithms (Evaluate Level)
- CO-7: Create data-mining algorithms/solutions for different applications (Create)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1			1	2					3
CO2			2	1				3	
CO3				1				2	3
CO4	1			2			3		
CO5		2						3	2
CO6		1					3		2
CO7	3				1			2	

Detailed Syllabus:

UNIT- I:
Introduction: Origins of Data Mining, Data Mining Tasks, Major Issues in Data Mining; Data Pre-processing: Data sampling, Data Cleaning, Integration and Transformation, Feature Selection and Dimensionality Reduction Techniques.

UNIT- II:
Classification: Decision Tree Induction, Bayesian Classifier, Rule-Based Classifier, Support Vector Machines, Ensemble methods, Model evaluations.

UNIT- III:
Association Analysis: Apriori algorithm and its extensions, Compact Representation of Frequent Itemsets, Various kinds of Association Rules, Sequential Patterns, Infrequent Patterns.

UNIT- IV:
Clustering: Partitional and Hierarchical clustering methods, Graph and Density based methods, Cluster Evaluation, Self Organizing Maps.

UNIT- V:
Anomaly Detection and Advanced Concepts: Statistical based, and density-based methods. Graph Mining, Social Network Analysis, Web Mining, Spatial Mining and Temporal Mining.

Reference Books:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining, 2nd edition, Dorling Kindersley (India) Pvt. Ltd, Licensees of Pearson Education.
2. Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publishers.
3. Margaret H. Dunham, Data Mining Introductory and Advanced Topics, Pearson Education (Singapore) Pte. Ltd.
4. K.P. Soman, Shyam Diwakar, V. Ajay, Insight into Data Mining Theory and Practice, PHI Learning Private Ltd.,

School of Computer and Information Sciences

Name of the Academic Program: M.Tech(Artificial Intelligence) (M.Tech-I)

Course Code: AI423 Title of the Course: Knowledge Representation & Reasoning

L-T-P: 3-0-0 Credits: 3

Prerequisite Course / Knowledge (If any): Discrete Mathematics

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1: Express knowledge of a domain formally (Understand)
- CO-2: Explain the production systems, frames, inheritance systems and approaches to handle uncertain or incomplete knowledge (Understand).
- CO-3: Examine the principles of reasoning (Analyze)
- CO-4: Describe how knowledge based systems work (Understand)
- CO-5: Illustrate knowledge based approaches to problem solving (Apply)
- CO-6: Design & Develop a knowledge- based system (Create)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1			3				2		
CO2				2				3	
CO3			2				3		
CO4			3						2
CO5			1		2			3	
CO6	2	1							3

Detailed Syllabus:

UNIT-I Introduction**:** The Key Concepts: Knowledge, Representation, and Reasoning, Why Knowledge Representation and Reasoning? Knowledge-Based Systems, Why knowledge Representation? Why Reasoning? The Role of Logic, Propositional Logic basics, Soundness & Completeness, Resolution Proof, Semantic Tableaux, Binary Decision Diagrams

UNIT-II: The Language of First-Order Logic: Introduction, The Syntax, The Semantics, Interpretations, Denotation, Satisfaction and Models, Logical Consequence Why We Care, Explicit and Implicit Belief, Knowledge-Based Systems. Expressing Knowledge. Knowledge Engineering, Vocabulary, Basic Facts, Complex Fact, Terminological Fact, Entailments, Abstract Individuals, Other Sorts of Facts.

UNIT-III: Resolution: The Propositional Case, Resolution Derivations, An Entailment Procedure, Handling Variables and Quantifiers, First-Order Resolution, Answer Extraction., Skolemization, Equality, Dealing with Computational Intractability, The First-Order Case, The Herbrand Theorem, The Propositional Case , The Implications , SAT Solvers, Most General Unifiers, Other Refinements

UNIT-IV: Reasoning with Horn Clauses: Horn Clauses, Resolution Derivations with Horn Clauses, SLD Resolution, Goal Trees, Computing SLD Derivations, Backward Chaining, Forward Chaining, The First-Order Case.

UNIT-V**:** Procedural Control of Reasoning: Facts and Rules, Rule Formation and Search Strategy, Algorithm Design, Specifying Goal Order , Committing to Proof Methods , Controlling Backtracking, Negation as Failure Dynamic Databases, The PLANNER Approach.

UNIT-VI Structured Descriptions: Descriptions, **Noun** Phrases, Concepts, Roles ,and Constants A Description Language, Meaning and Entailment, Interpretations, Truth in an Interpretation, Entailment, Computing Entailments, Simplifying the Knowledge Base, Normalization, Structure Matching, The Correctness of the Subsumption Computation, Computing Satisfaction, Taxonomies and Classification A Taxonomy of Atomic Concepts and Constants, Computing Classification, Taxonomies versus Frame Hierarchies, Beyond the Basics Extensions to the Language Applications of Description Logics.

UNIT-VII: Defaults: Introduction, Generics and Universals , Default Reasoning, Non-monotonicity, Closed-World Reasoning, The Closed-World Assumption Consistency and Completeness of Knowledge Query Evaluation Consistency and a Generalized Assumption Quantifiers and Domain Closure , Circumscription, Minimal Entailment The Circumscription Axiom, Fixed and Variable Predicates, Default Logic, Default Rules Default Extensions Multiple Extensions, Auto epistemic Logic Stable Sets and Expansions, Enumerating Stable Expansions.

UNIT-VIII: Actions & Planning**:** The Situation Calculus, Fluents, Precondition and Effect Axioms, Frame Axioms, Using the Situation Calculus, A Simple Solution to the Frame Problem, Explanation Closure, Successor State Axioms, Complex Actions. **Planning:** Planning in the Situation Calculus, Using Resolution, The STRIPS Representation, Progressive Planning, Regressive Planning Planning as a Reasoning Task, Avoiding Redundant Search, Application-Dependent Control , Beyond the Basics Hierarchical Planning Conditional Planning

Reference Books:

1. Language, Proof and Logic, Jon Barwise & John Etchemendy, CSLI Publications (1999); ch 9-11.
2. Knowledge representation and Reasoning, Ronald J. Brachman & Hector J. Levesque, Elsevier (2004); ch 2-6, 9, 11, 14,15.
3. The Description Logic Handbook: Theory, implementation, and applications, Franz Baader, Deborah L. McGuinness, Daniele Nardi and Peter F. Patel-Schneider, Cambridge University Press (2010); ch 2, 5-6.

School of Computer and Information Sciences

Name of the Academic Program: M.Tech (Artificial Intelligence) (M.Tech-I)

Course Code: AI424 Title of the Course: Neural Networks

L-T-P: 3-0-0 Credits : 3

Prerequisite Course / Knowledge (If any):

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1 Memorize the basic concepts of neural network previously learned and recognize the network models for performance optimization in the present study.
- CO-2 Describe the architecture of networks involving connectionist mechanism, network's working principle, learning connections, network's recognition efficiency.
- CO-3 Categorize the networks in accordance of real life problems to be in supervised and unsupervised manner.
- CO-4 Implement the network structures and demonstrate its performance on real life data
- CO-5. Predict the performance of networks for unknown data and evaluate it through standard metrics.
- CO-6. State the fitability of networks and their additional benefits and disadvantages.
- CO-7. Utilize the concepts and techniques described in the present study to generate innovative technical learning theories in further investigation.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1			3	2				1	
CO2				1				2	3
CO3	1					2		3	
CO4				2	1		3		
CO5	3						1	2	
CO6			1				2	3	
CO7			1			3		2	

Detailed Syllabus:

UNIT –I: Introduction: History of Neuro-computing (The Beginning First Successes, The Quest Years Neuro-computing Takes off). The biological prototype, The neural node (MCP Neuron)

UNIT-II: The Perception, linear separable function, Madaline

UNIT-III: Learning Laws**:** Self Adaption Equations (Training), Coincidence Learning, Performance Learning, Competitive Learning, Filter Learning, Spatiotemporal Learning.

UNIT-IV: Associative Networks: Linear Associator Network, (feedforward) auto-associativity Recurrent Associate Networks (The Hopfield Nets, The Brain state in a Box Network, Associative Network Theorem) BAM's (hetro associativity)

UNIT-V: Multilayer-Network**:** The Backpropagation Networks, Self-organizing Maps, Counter propagation Networks, GMDH (Group Method of Data Handling), Hamming Network.

UNIT-VI: Frontiers of Neuro computing**:** Spatio temporal, Stochastic (Boltzmann) and Hierarchical Networks (Neuro cognitron) – Knowledge-Based Neural Networks. Neurocomputers**:** Machines for Implementing Neural Network. Neuro computing Applications

Reference Books:

1. Artificial Neural Networks, B.Yegnanarayana, Publishers PHI, 1999.
2. Artificial Neural Networks by Robert J Schalkoff, McGraw Hill, 1997.
3. Introduction to Neural Computing, Igor Alek Sander & Helen Morton Chapman & Hall, 1990.
4. Neural Computing: Theory & Practice, P.D.Wasserman, Van Nostrand Reinhold, NY, 1989. Explorations in Microstructure of Cognition.
5. Parallel Distributed Processing: Vol. 1&2, D.E.Rumelhart and J.L.McCelland(Ed) MIT Press, 1986.

School of Computer and Information Sciences

Name of the Academic Program: M.Tech (Artificial Intelligence) (M.Tech-I)

Course Code: AI 425 Title of the Course: Pattern Recognition

L-T-P: 3-0-0 Credits : 3

Prerequisite Course / Knowledge (If any): Students are expected to have knowledge of Mathematics: Calculus, Differential Equations, Linear Algebra, Probability, and Completed Programming Methodology, Programming of Scientific and Mathematical Functions

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1: Describe the approaches to statistical and syntactic pattern recognition (Understand)
- CO-2: Describe the design characteristics of pattern recognitions systems such as curse of dimensionality (Understand)
- CO-3: Apply pattern recognition techniques to solve real world problems (Apply)
- CO-4 : Design simple pattern classifiers, classifier combinations ad structural pattern recognizers (Create)
- CO-5. Develop pattern recognition techniques and the scientific computing environment. (Create)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1									
CO2									
CO3									
CO4									
CO5									
CO6									
CO7									

Detailed Syllabus:

UNIT-I: Introduction to Patterns and Recognition Methods: Concepts of Pattern Recognition, applications in real world; patterns and their representations as random vectors, strings and graph, time series. Feature and data scales. Feature and pattern similarity, dissimilarity measures. Feature selection, dimensionality reduction. PCA and SVD methods. Introduction to pattern classification paradigms such as statistical, syntactical and structural pattern recognition. Nearest Neighbor Classification as a case study.

UNIT-II: Statistical Pattern Recognition: Bayes decision theory, Maximum aposteriori classification, risk and errors; Supervised learning using parametric and nonparametric approaches: Maximum Likelihood estimation, Bayesian parameter estimation approach, Parzen Windows, k-NN estimation;

UNIT-III: Linear Discriminative Classifiers: Decision Boundaries, Separability; Perceptron's, Support Vector Machines, Fisher's Linear Discriminant function, Decision Trees, Random Forests, Projections and Embeddings

UNIT-IV Unsupervised learning and other Methods: Unsupervised classification, clustering: the clustering concept, clustering strategies, a hierarchical clustering, Partitional clustering, c-means algorithm, learning vector quantization, expectation maximization and mean shift.

UNIT-V String and Sequence Recognition Methods: Formal languages; String languages for pattern recognition: selection of pattern primitives, Matching of Strings, Edit distances, pattern grammars, basic formulation of states and sequences, Markov Models, Hidden Markov Models, Viterbi Algorithm, Baum-Welch Learning, Graphs fundamentals of graph theory, basic algorithms for graphs matching

Suggested Reading:

1. Duda, R.O., Hart, P.E., and Stork, D.G. Pattern Classification. Wiley-Interscience. 2nd Edition. 2001. The textbook has a website: http://www.rii.ricoh.com/˜stork/DHS.html.
2. Andrew R. Webb, and Keith D. Copsey, Statistical Pattern Recognition, Third Edition, Wiley-Interscience, 2011

Reference Books

3. Bishop, C. M. Pattern Recognition and Machine Learning. Springer. 2007.
4. Theodoridis, S. and Koutroumbas, K. Pattern Recognition. Edition 4. Academic Press, 2008.
5. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.
6. Hastie, T., Tibshirani, R. and Friedman, J. The Elements of Statistical Learning. Springer. 2001.
7. Koller, D. and Friedman, N. Probabilistic Graphical Models. MIT Press. 2009.

School of Computer and Information Sciences

Name of the Academic Program: M.Tech(Artificial Intelligence) (M.Tech-I)

Course Code:AI426 Title of the Course: Problem Solving Methods

L-T-P: 3-0-0 Credits : 3

Prerequisite Course / Knowledge (If any): Data Structures, Algorithms

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1: Describe various characteristics of Intelligent Agents (Understand)
- CO-2: Choose suitable graph search algorithms for given problems (Apply)
- CO-3: Analyse time and space complexities of the various problem solving methods(Analyze)
- CO-4: Examine AI problems as Constraint Satisfaction Problems (Analyze)
- CO-5: Justify the scope of Uncertainty in AI problem Solving and its applications (Evaluate)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1		2					1	3	
CO2			1	2			3		
CO3							3		
CO4				1			2		3
CO5	2			1				3	

Detailed Syllabus:

UNIT- I: INTRODUCTION

Introduction–Definition – Applications of Artificial Intelligence in Modern Society– Characteristics of Intelligent Agents–Typical Intelligent Agents – Characteristics of Agents and Environments, Rational Agent Behavior, Nature of Environments, Structure of Agents, Physical symbol system hypothesis, Weak and Strong AI

UNIT- II: SEARCH AND PROBLEM SOLVING

Problem solving Methods – Search Strategies- Uninformed – Informed – Heuristics – Local Search Algorithms and Optimization Problems -Searching with Partial Observations – Solving Problems by Searching, Beyond Classical Search, Local Search Algorithms and Optimization Problems, Local Search in Continuous Spaces, Searching with Non-deterministic Actions, Searching with Partial Observations, Online Search Agents and Unknown Environments

UNIT-III: ADVERSARIAL SEARCH

Games, Optimal Decisions in Games, Alpha–Beta Pruning, Imperfect Real-Time Decisions, Stochastic Games, Partially Observable Games, State-of-the-Art Game Programs, Alternative Approaches

UNIT-IV: CONSTRAINT SATISFACTION PROBLEMS

Defining Constraint Satisfaction Problems, Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs, Local Search for CSPs, Structure of Problem. Introduction to GA and Solution of CSP by GA

UNIT-V UNCERTAIN ENVIRONMENTS

Quantifying Uncertainty, Acting under Uncertainty, Probability, Conditional Probability and other related background for Bayes Theory, Representing Knowledge in an Uncertain Domain, Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Exact and Approximate Inference in Bayesian Networks, Introduction to other approaches to Uncertain Reasoning

School of Computer & Information Sciences

Name of the Academic Program: M.Tech (Artificial Intelligence) (M.Tech-I)

Course Code : AI427 Title of the Course : Simulation & Modeling

L-T-P :. 4-0-0 Credits : 4

Prerequisite Course / Knowledge (If any): Sound knowledge of Mathematics, Probability, and Statistics at UG level. Aptitude and exposure to the subject Operation Research will help further.

Course Overview: This is a first course of 3rd Paradigm of research provides philosophy of modelling and designing of simulations experiments. The Students will be able generate random deviates for a given uncertainty/probabilistic model. The students will acquire verification & validation methods, Monte Carlo simulations and basic mathematical/Statistical modelling and simulation of the same. Expected to develop the skill of optimization by hands-on demonstration in matlab and/or R.
The course provides the students with a good understanding of the discrete-event simulation concepts and help them to utilize the concepts to analyse the performance of various computing or real-world systems.

Course Outcomes (COs)

After completion of this course successfully, the students will be able to……

CO1: Discuss the role and limitations that simulation and modelling can play in the real world problem-solving process (Understand)
CO2: Discuss how to generate generation of pseudo random numbers and test for random. (Understand)
CO3: Apply transformation methods to generate random deviates for a given probability model. (Discrete as well as Continuous) (Apply)
CO4: Apply Monte Carlo methods for generating random deviates for a given probability model. (Apply)
CO5: Develop apt mathematical model for a given Real world problem and designing simulation strategies (Create)
CO6: Analyse the given variance reduction method. (Analyze)
CO7: Analyse the simulated data by carrying out discretise statistics and inferential statistics . (Analyze)
CO8: Design an apt Monte Carlo optimization method for recommending an near optima solution for some of the optimization problems of the real world. (Create)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
CO7												
CO8												

Detailed Syllabus:

UNIT- I

Introduction to Simulation: Simulation, advantages, Disadvantages, Areas of application, System environment, components of a system. Overview of Mathematical modelling: Model of a system, types of models, steps in a simulation study.- Chemical reactor, Population dynamic.
Random Numbers: Properties, Generations methods, Tests for Random numbers – Frequency test, Runs test, Autocorrelation test, Gap test, Poker test.

UNIT- II
Introduction to Probability distributions: Weibull, Triangular, Erlang and Gamma distributions and their applications (No analytical treatment)

Random Variate Generation:
Inverse Transform Technique- Exponential, Uniform, Weibull, Triangular distributions, Direct transformation for Normal and lognormal Distributions Convolution Method – Erlang distribution Acceptance and Rejection technique – Poisson and Gamma distributions. Generalised lambda system and its use in simulation. Generation of random variate of multivariate normal distribution.

UNIT- III

Simulation Examples: Simulation of Queuing systems, Simulation of Inventory System, Monte Carlo simulation, General Principles, Concepts in discrete – events simulation, event scheduling/Time advance algorithm.

UNIT- IV
Input Modeling, Verification and Validation of Simulation Model. Evaluation of simulation experiments.

UNIT- V
Monte Carlo methods. Optimization via simulation. Variance reduction methods. Basic GA, PSO and SGD.

Text books:

1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol, Discrete-Event System Simulation, Pearson Prentice Hall, 2010, 5th Edition (ISBN: 0136062121).

2. Narsingh Deo -System Simulation with Digital Computer; PHI Publication (EEE), ISBN – 0-87692-028-8

3. Byron J. T. Morgan (auth.)-Elements of Simulation-Springer US (1984) ISBN 978-0-412-24580-0 ISBN 978-1-4899-3282-2 (eBook) DOI 10.1007/978-1-4899-3282-2

4. Geoffery Gordon – System Simulation (2005) Prentice-Hall of India Private Limited, New Delhi ISBN-81-203-0140-4

Reference:

1. Averill M Law, W David Kelton -Simulation Modeling & Analysis, McGraw Hill International Editions – Industrial Engineering series, ISBN – 0-07-100803-9.

2. Reuven Y Rubinstein and Dirk P Kroese - **Simulation and the Monte Carlo Method, Third Edition** Print ISBN:9781118632161 |Online ISBN:9781118631980, 2017 John Wiley & Sons,

School of Computer & Information Sciences

M.Tech (Artificial Intelligence)

Scheme

II-Semester					
Core			Optional Core/Elective		
Code	**Course Title**	**Credits**	**Code**	**Course Title**	**Credits**
CS451	Software Engineering	4	AI471	Colour Image Processing	
	Optional core-III	4	AI472	Machine Learning	
	Elective –II	¾	AI473	Deep Learning	
	Elective –III	¾	AI474	Natural Language Processing	
	Elective –IV	¾	AI474	Rough Computing	
CS452	*IT Lab –II*	2			
CS453	*SE Lab*	2			
CS454	*Communication Skills*	3			
		21/24			
IV-Semester					
Code	**Course Title**	**Credits**			
	Project	6			

School of Computer and Information Sciences

Name of the Academic Program: **M.Tech (Artificial Intelligence) (M.Tech-II)**

Course Code: CS 451 Title of the Course : **Software Engineering**

L-T-P : 3-0-0 Credits : 3

Prerequisite Course / Knowledge (If any): It is expected that the students must have done at least one programming course at undergraduate/postgraduate level

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO1: Explain the models of software development process (Understand)
- CO2: Evaluate the appropriateness of different models of software development for their application in various domains (Evaluate).
- CO3: Apply the requirements engineering to software systems. (Apply)
- CO4: Describe Software Architectures (understand).
- CO5: Assess the applicability of software architectures for various combinations of non-functional requirements (Evaluate level).
- CO6: Apply object oriented and structured and structured paradigms to design software systems (Apply).
- CO7: Apply testing strategy to test software applications (Apply).

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1			2	3	1				
CO2			3	2	1				
CO3			1	3	2				
CO4			3	2	1				
CO5	3		2		1				
CO6		1		3	2				
CO7	2	1		3					

Detailed Syllabus:

UNIT-I: Introduction to Software Engineering

Need of software engineering, systems engineering, challenges in software engineering, Software process models, quality characteristics of software systems, Ethics in Software Engineering.

UNIT-II: Requirements Engineering

Requirements engineering process, requirements specification, structured and object oriented analysis

UNIT-III: Software Design

Architectural design, detailed design, Structured and object oriented design, user interface design

UNIT-IV: Software Testing

Verification, Validation, testing techniques, Testing Process

UNIT-V: Tools and Evolution

CASE Tools, Reverse engineering, Reengineering and Configuration management.

Reference Books:

1. Ian Sommerville (2016), "*Software Engineering*",10th Edition, Pearson Education Limited, Global Edition
2. Roger S Pressman, Bruce R Maxim(2015), "*Software Engineering, A Practitioner's Approach*", 8th Edition, TataMcGraw Hill, Indian Edition
3. Grady Booch, James Rumbaugh, Ivor Jacobson(2005), "*The Unified Modeling Language User Guide*", 2nd Edition, Addison Wesley Professional.US

School of Computer and Information Sciences

Name of the Academic Program: **M.Tech (Artificial Intelligence) (M.Tech-II)**

Course Code CS 453 Title of the Course : **Software Engineering Lab**

L-T-P : 0-0-3 Credits : 2

Prerequisite Course / Knowledge (If any): It is expected that the students must have done at least one programming course at undergraduate/postgraduate level

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO1: Create user stories (Create).
- CO2: Develop test plans for test first development (Create).
- CO3: Design & develop the stories (Create).
- CO4: Create the documentation (Create).
- CO5: Develop Software requirements specification document (Create).
- CO6: Apply object oriented and structured paradigm (Apply).
- CO7: Generate test reports (Create).

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1	2			3	1				
CO2		2	3	1					
CO3			2	3	1				
CO4	2	3			1				
CO5		1	3	2					
CO6		1		2	3				
CO7		2	3	1					

Detailed Syllabus:

For a given case study/problem statement, the following deliverables are to be realized

- Define stories
- Identify tasks and develop test plan for stories/task (with the help of specifications)
- Design and develop increments
- Test the increments and release the increment
- Apply object oriented and structured modelling
- Implement the case study for plan driven approach by writing use case specification, designing the system and implementing the same.

Reference Books:

1. Ian Sommerville (2016), "*Software Engineering*",10th Edition, Pearson Education Limited, Global Edition
2. Roger S Pressman, Bruce R Maxim(2015), "*Software Engineering, A Practitioner's Approach*", 8th Edition, TataMcGraw Hill, Indian Edition

School of Computer and Information Sciences

Name of the Academic Program: **M.Tech (Artificial Intelligence) (M.Tech-II)**

Course Code : AI475 Title of the Course : **Rough Computing**

L-T-P : 3-0-0 Credits : 3

Prerequisite Course / Knowledge (If any): Discrete Mathematics, Knowledge in one programming language

Course Outcomes (COs)

After completion of this course successfully, the students will be able to

- CO-1: Explain the fundamental principles of Rough sets and its applications to Uncertainty Management (Understand)
- CO-2: Practice Rough set algorithms on Data Mining and Knowledge Discovery in Databases (Apply)
- CO-3: Use principles of Boolean Reasoning to solve real world applications (Apply)
- CO-4: Discuss the Principles of Feature Subset Selection (Reduct Computation) and different ways of Reduct Computation (Understand)
- CO-5: Design the extensions and adaptations of Rough sets for different vague/uncertain scenarios and will be equipped to deal with new situations unseen before (Create)
- CO-6: Apply Rough sets and its extensions for Hybrid Soft Computing Model (Classification and Clustering) construction (Apply)

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

	PO1	PO2	PO3	PO4	PO5	PO6	PSO1	PSO2	PSO3
CO1			3	2				1	
CO2						1	3		2
CO3	2				1		3		
CO4									
CO5			1	3				2	
CO6				1			2		3

Detailed Syllabus:

Unit I:

- Fundamentals of Rough sets:
    1. Rough set principles
    2. Indiscernibility Relation and Properties of Rough Approximations
    3. Information system, Decision system

Unit II:

- Boolean Reasoning
    1. Discernibility Relation and Matrix
    2. All Reduct Computation
    3. Discretization

Unit III:

- Feature selection
    1. Reduct and Core Computation in Information/Decision Systems
    2. Based on Relative Dependency between Attributes
    3. Alternative approaches to Reduct Computation using Genetic Algorithms and Ant Colony Optimization
    4. A Scalable Rough Set Knowledge Reduction Algorithm

Unit IV:

- Rule Discovery
    1. Decision Tree formation & Rule generation
    2. LEM1, LEM2 Algorithms
- Hybrid Systems
    1. Rough sets and Neural Networks for Classification
    2. Rough K-Means Algorithm for Clustering

Unit V:

- Extensions of Rough Sets
    1. Extensions for Incomplete Decision/Information Systems (Tolerance Rough Sets)
    2. Extensions for Hybrid Decision/Information Systems (Fuzzy Rough Sets)

Reference Books:

1. Aboul Ella Hassanien, Zbigniew Suraj, Dominik Slezak & Pawan Lingras (2008) "Rough Commuting. Theories, Technologies and Applications" Information Science Reference, Hershey, New York
2. Zdzislaw Pawlak (1992) "Rough Sets: Theoretical Aspects of Reasoning about Data", Kluwer Academic Publishers.
3. Rafael Bello, Rafael Falcón, Witold Pedrycz, Janusz Kacprzyk (Eds.) (2008) Granular Computing: At the Junction of Rough Sets and Fuzzy Sets. Studies in Fuzziness and Soft Computing, Volume 224, Springer
4. Sankar Kumar Pal, Lech Polkowski and Andrzej Skowron (Eds) (2004) "Rough-Neural Computing - Techniques for Computing with Words" Springer.