Department of Computer science and Engineering, IGDTUW

M. Tech.- CSE (Artificial Intelligence)

First Semester

S. No.	Code	Subject	L-T-P	Credits	Category
1.	MCS- 101	Problem Solving Through AI	3-0-2	4	DCC
2.	MCS- 103	Soft Computing	3-0-2	4	DCC
3.	MCS- 105	Intelligent Data and Information Retrieval	3-0-2	4	DCC
4.	MCS- 107	Data Structures and Algorithm Analysis	3-0-2	4	DCC
5	GEC-101	Generic Open Elective-I #	2-0-0/ 1-1-0/ 0-0-2	2	GEC
6.	ROC -101	Research Methodology	3-0-0	3	ROC
		Total credits		21	

Second Semester

S. No.	Code	Subject	L-T-P	Credits	Category
1.	MIS- 102	Machine Learning	3-0-2	4	DCC
2.	MCS -104	IoT and its Applications in AI	3-1-0	4	DCC
3.	MCS- 106	Social Network Analysis	3-0-2	4	DCC
4.	DEC1 xx	Departmental Elective Course – 1	3-0-2/ 3-1-0	4	DEC
5.	DEC1 xx	Departmental Elective Course - 2	3-0-2/	4	DEC

			3-1-0		
6	ROC- 102	Research Ethics	3-0-0	3	ROC
		Total credits		23	

Third Semester

S. No.	**Code**	**Subject**	**L-T-P**	**Credits**	**Category**
1.	MIS- 203	Neural network and Deep Learning	3-0-2	4	DCC
2.	DEC- 2xx	Departmental Elective-3	3-0-2	4	DEC
3.	DEC- 2xx	Departmental Elective-4	3-10/ 3-0/2	4	DEC
4	GEC- 201	General Open Elective-II #	2-0-0/ 1-1-0/ 0-0-2	2	GEC
5	MCS- 251	Dissertation-I * /Project work	-	8	DCC
6	MCS- 253	Industrial Training/Internship		1	DCC
		Total credits		23	

* Dissertation -I should be preferably done in Industry by taken a real Industry problem .

Fourth Semester

S. No.	Code	Subject	**L-T-P**	**Credits**	Category
1.	MCS- 252	Dissertation -II ***** /Project work	-	20	DCC
		Total credits		20	

* Dissertation -II should be preferably done in Industry by taken a real Industry problem.

List of Departmental Elective Courses

Category	Course Code	Subject	Credits
Departmental	MCS- 108	Knowledge Engineering	3-1-0
Elective Course-1	MCS- 110	AI based Programming Tools	3-0-2
	MCS- 112	Robotics and its Applications	3-0-2
	MCS- 114	Cloud Computing	3-0-2
Departmental	MCS- 116	Big Data Analytics	3-0-2
Elective Course-2	MCS- 118	Parallel Algorithms	3-0-2
	MCS- 120	Knowledge Based System Design	3-0-2
	MCS- 122	Computer vision	3-1-0
Departmental	MCS- 203	Digital Image Processing (DIP)	3-0-2
Elective Course-3	MCS- 205	Natural Language Processing	3-0-2
	MCS- 207	Mobile Application Development	3-0-2
	MCS- 209	Human Computer Interaction	3-0-2
	MCS- 211	Speech Processing and Speech Recognition	3-0-2
Departmental	MCS- 213	Real Time Systems	3-0-2
Elective Course-4	MCS- 215	Agent Based Intelligent Systems	3-1-0
	MCS- 217	Robotics and Applications	3-1-0
	MCS-219	Wireless Sensor Networks	3-0-2

Syllabus of First Year subjects

PROBLEM SOLVING THROUGH Al	
Course Code : MCS - 101 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits: 4 Semester: 1

Introductions: Artificial intelligence aims to understand thinking and intelligence in ways enable the constructions system that are able to reason in uncertain environments. This course will discuss fundamental concepts of Artificial intelligence and learning of how these concepts can be utilized to solve problems pertaining to Al.

Course Objective

- Demonstrative working knowledge in programming language Lips/Prolog in order to write simple programs.
- Know various Al search algorithms.
- Understand the fundamentals of knowledge representation (logic- based, frame-based, semantic nets), inference and theorem proving.
- To teach students , to reformulate any problem from AI perspective

Pre-requisite: basic knowledge of programming language fundamental concepts of mathematics and automation.

Course Outcome: After studying this course students will be able to :

- Develop the Ability to choose knowledge representation method for different problems
- Learn about reasoning and machine learning techniques to real word problems.
- Know how to builds simple knowledge based system.
- Understand the process of Problem solving through AI perspective

Pedagogy: Classroom teaching which focuses on developing understanding of students to digest the concepts of subject with large number of examples.

Contents

Unit -1
Introduction : Al problems Task Domains of Al, Al Techniques: search knowledge, abstraction. Introduction to intelligent program and intelligence agents. Problem solving: basic problem solving Method: state space search, problem characteristics, production systems characteristics, issues in design of intelligent search algorithm.
Unit - II
Heuristic search techniques : Hill climbing techniques, best first search, A* search, problem Reduction: AO* search, constraint satisfaction, means-end Analyst. Programming Languages: Fundamental and concepts of programming languages like prolog or Lips. Relationship of Language with knowledge presentation and inferences.
Unit- III
Knowledge Representation: Knowledge Representation issues. Knowledge Representation using predicate logic: Unification resolution. Rule based systems: Forwarded versus backward reasoning conflict resolution. Structured knowledge Representation: Semantic Nets, Frames, conceptual dependency, scripts.
Unit – Iv
Learning from observation: Inductive learning decision trees, computational learning Theory Explanation based learning Applications: Environmental scince, Robotics, Aerospace, Medical Science etc.
Text Book
1. Stuart j. Russell and peter Norvig, Artificial Intelligence,- A Modern Approach , Pearson 2nd Edition 2009.
2. E. Rich and K. Knight, "Artificial Intelligence", TMH, 2nd ED.,1992.
3. Ela Kumar," Artificial Intelligence ", I.K. International Publishing House, 2011.
Reference Book
1. P.H. Winston, " Artificial Intelligence," Pearson Education, 3rd Edition,2002
2. D.W. Patterson, "Introduction to Al and Expert Systems ", PHI,1992
3. Lugar, Artificial intelligence, PHI publication, 2015 , second edition

Suggestive Lab exercises to be done (More programs can be added in the list).

The language for implementation can be chosen to be any out of C / R / Python / Lisp/ Prolog.

1. To write a program to implement the recursion.
2. Write a program implement forward chaining.
3. Write a program create a dynamic link list.
4. Write a program create a circular link list.
5. Write a program implement A* algorithm.
6. Write a program implement best first search.
7. Write a program to implement Min Max search.
8. Write a program to implement Tic Tac Toe game.

9. Write a program perform the alpha beta pruning.
10. Write a program implement merge sort on two input data list
11. Write a program in python to implement bidirectional search
12. Write a program to implement backward chaining

SOFT COMPUTING	
Course Code: MCS-103 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits: 4 Semester: 1

Introduction: Soft Computing aims to introduce intelligent computing techniques for real world problem solving where the conventional hard computing techniques do not work. The course will solving real world problems when the data is imprecise in nature and the boundaries are non precise. Further the course will introduce to make learning systems and different techniques to optimize the results

Course Objective:

- To appreciate the need of Soft Computing Techniques for solving real world problem which cannot be solved by conventional hard computing techniques.
- To learn different types of sets which can handle imprecise data values.
- To develop systems which have learning capabilities.
- To learn techniques to optimize the results and find the optima.

Pre-requisite: Knowledge of basic Mathematics and Algorithms

Course Outcome: At the end of the course students will be

- Able to realize importance and apply Soft Computing techniques for real world problem solving
- Able to represent the imprecise information using sets and develop inference systems based on these.
- Develop learning systems
- Learn and practice various optimization algorithms for real world problems solving

Pedagogy: Classroom teaching which focuses on developing understanding of students to digest the concepts of subject with large number of examples.

Contents

Unit -1
Introduction of soft computing: Soft computing vs. hard computing, various types of soft computing techniques, applications of soft computing. Concept Of Uncertainty : Presence of uncertainty in real world problems, handling uncertain knowledge, Bayesian Classifiers,

Perplexed bayes classifiers.

(12 Hrs)

Unit - II

Fuzzy sets and fuzzy logic: Introduction to fuzzy logic, classical and fuzzy sets, overview of fuzzy sets, membership function, fuzzy rule generation, operations on fuzzy sets: compliment, intersection, union, combinations on operations, aggregation operation. Fuzzy Extension Principles, Defuzzification. Fuzzy Rule bases, Development of Fuzzy Logic based Expert Systems.CASE STUDIES **(10 Hrs)**

Unit- III

Neural Networks &Rough Sets: Overview of biological neurons, Mathematical model of Neuron, Perceptron and Multi Layer Perceptron, Learning in Artificial Neural Networks; Supervised, Unsupervised and Competitive Learning paradigms; Learning rules and Functions, Back propagation algorithm, Rough Sets. Upper and Lower Approximations, Boundary Region, Decision Tables and Decision Algorithms. Properties of Rough Sets. Rough Membership, Reducts. Discernibility Matrix and Discernibility Functions. Generation of Inference Rules. CASE STUDIES- **(10 Hrs)**

Unit – Iv

Evolutionary Algorithms and Hybrid Algorithms: Introduction, Evolutionary algorithms - Genetic Algorithm: History, terminology, biological background, creation of offspring, working principles of genetic algorithms, fitness function, Roulette wheel selection, Boltzmann selection, cross over mutation, inversion, deletion, and duplication, generation cycle , Swarm Optimization –Part Swarm Optimization and Ant Colony Optimization.Differential Evolutionary Algorithm.

Hybrid Algorithms, Neuro Fuzzy, Fuzzy Evolutionary etc. **(10 Hrs)**

Text Book

1. Principles of Soft Computing - Deepa Shivandan Das, 2012, JohnWiley Publication.

2. Fuzzy Logic: A spectrum of Theoretical and Practical issues, Paul P. Wang, Publication 2004.

3Fuzzy Sets, Fuzzy logic, and Fuzzy Systems: Selected Papers- Lotfi Asker Zadeh, George J. Kilr, Bo yuan,2005.

Reference Book

1.Introduction of Soft Computing- Neuro Fuzzy and Genetic Algorithms, Samir Roy and

Udit Chakraborty, Pearson Education, 2013.
2. D.W. Patterson, "Introduction to Al and Expert Systems ", third edition PHI,2016
3Nature-Inspired Metaheuristic Algorithms: Second Edition , Xin-She Yang, Luniver press, 2010

Suggestive List of experiments (More programs can be added to this list)

1: Write a program To implement Baye's classifier for classification of sample data.

2: Write a program to implement perplexed Baye's classifier for analysis of textual data.

3: Write a program to implement a Fuzzy Set and compute Union, intersection and complementation of members. Your program should take input from the screen.

4: Develop an Expert System in MATLAB Toolbox for Inferencing using Fuzzy Inference Rules.

5: Write a program to develop a multilayered perceptron.

6: Write a program to compute discernibility matrix and discernibility function. Generate Inference Rules and eliminate inconsistent inference Rules.

7: Implement genetic algorithm and solve Travelling salesman Problem for five nodes. Display the number of iterations used to reach the optimal solution.

8: Implement Differential Evolutionary Algorithm to solve forecasting problems.

9: Implement Ant Colony Optimization Algorithm for optimizing results.

10: Implement Particle Swarm Optimization Algorithm On Task Allocation Problem.

INTELLIGENT DATA AND INFORMATION RETRIEVAL	
Course Code: MCS-105 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits: 4 Semester: 1

Introduction: Intelligent Data and Information Retrieval aims to provide application of various concepts of artificial intelligence for organizing& fetching data and information from the internet databases like search Engines . The Subject will introduce various types Intelligent data storage and processing techniques and also how to intelligently retrieve data from web sources so that the results of queries are exact and efficient.

Course Objective:

- To understand the concepts of intelligently organizing data and fetching data from queries.
- To learn the different models for information storage and retrieval.
- To understand indexing and querying in information retrieval systems.
- To learn techniques for intelligently retrieving information from web search

Pre-requisite: Knowledge of basic databases and algorithms

Course Outcome: At the end of the course students will be

- Able to organize data intelligently and fetch using FSQL
- Deduce inferences from stored databases
- Design algorithms for retrieving information effectively.
- Retrieve information efficiently from web

Pedagogy: Classroom teaching which focuses on developing understanding of students to digest the concepts of subject with large number of examples.

Contents

Unit -1
Introduction: Introduction to data and various database Models. Data v/s information. Fuzzy Databases- Type-1 and Type-2 Fuzzy Relational Databases. Fuzzy Functional Dependency and Fuzzy Multivalued Dependency. Intelligent Query Processing using FSQL. Case studies of Fuzzy Databases. **(12 Hrs)**
Unit - II
Deductive Databases- Overview of Deductive databases, datalogue notations , Clausal Forms and Horn clauses, Interpretation of Rules, datalogue programs-safety issues, use of relational

operators, non-recursive queries, Evaluation of Non-recursive datalogue queries. Case studies of deductive databases **(10 Hrs)**
Unit- III
Information Retrieval: Introduction of IR. Comparison between databases and IR Systems. Generic IR pipeline. Retrieval Models- Boolean Model, Vector Space Model, Probabilistic Model, Semantic Model, Fuzzy Model.Wrappers. Relevance feedback, Evaluation Measures- Precision, Recall and F-Score. Fuzzy Queries based development of Question Answering systems, Error detection and correction. **(10 Hrs)**
Unit – Iv
Web Search and Analysis: PageRank Algorithm, HITS algorithm. Webcontent Analysis, ontology based IR. Intelligent Web Agents. Social Search- Collaborative and conversational. Query Expansion using Fuzzy operators. Case studies:-Development of MetaSearch Engine using intelligent operators like OWA, Web crawlers, web spamming , web analytics. **(10 Hrs)**
Text Book
1. Information Retrieval – Algorithms and Heuristics, David A. Grossman, Ophir Frieder, 2nd Edition, 2012, Springer, (Distributed by Universities Press)
2. Modern Information Retrieval Systems, Yates, Pearson Education,2014.
3. Information Storage and Retrieval Systems, Gerald J Kowalski, Mark T Maybury, Springer, 2000.
Reference Book
1. Mining the Web : Discovering Knowledge from Hypertext Data, Soumen Chakrabarti Morgan-Kaufmann Publishers, 2002.
2An Introduction to Information Retrieval, Christopher D. Manning, Prabhakar Raghavan, HinrichSchütze, , Cambridge University Press, Cambridge, England, 2009.
3. Martin, J " Intelligent Information retrieval: PHI publication, 3rd edition, 2013

Suggestive List of experiments (More programs can be added to this list)

1. Write a Program in Fuzzy SQL to create Type-1 Fuzzy Relations
2. Implement Fuzzy Selection and Fuzzy Projection operation in Fuzzy Relations.
3. Create a Program in Prolog to generate a Deductive database
4. Enter atleast 10 queries and draw inferences from the deductive database.
5. Write a program to implement information retrieval via Vector Space Model . Your program should also calculate Precision & Recall.

6. Write a program to implement information retrieval via Fuzzy Model. Your program should also calculate Precision & Recall.
7. Write a program to for query expansion . Your programs should also compare precision and recall values, before and after query expansion.
8. Write a program to extract information from meta search Engines.
9. Write a program to implement error correction using Edit Distance and n-gram methods.
10. Write a program to implement disambiguation in Information Retrieval Query. Calculate Precision & Recall.

DATA STRUCTURES AND ALGORITHM ANALYSIS	
Course Code: MCS- 107 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits: 4 Semester: 1

Introduction: This course is about teaching of various data structure designs & its implementations, analyzing the various algorithm strategies and designing of new algorithms for various classes of problems. It is intended to be a gentle introduction to how we specify data structure, algorithms, some of the design strategies, and many of the fundamental ideas used in algorithm analysis throughout the syllabus.

Course Objective:

- To build an understanding on the basics of core and advance data structure.
- To introduce the various strategies used in the algorithm design and their analysis.
- TO teach the selection of data structure for a particular problem
- To teach students, how to write complex program using dynamic data strcutres

Pre-requisite: Students should have some programming experience. In particular, they should understand recursive procedures and simple data structures such as arrays and linked lists. Students should have some facility with proofs by mathematical induction.

Course Outcome: After studying this course, Students will be able to :

- Successfully design and implements the core and advance data structures
- Successfully analyses the complexity associated with the various data structures
- Analyse, design and implements the various proposed algorithm based on different algorithmic strategies.
- Choose data structures for various complex problems

Pedagogy : Classroom teaching which focuses on developing understanding of students to digest the concepts of subject with large number of examples.

Contents

UNIT-I	10 Hours
Algorithms performance analysis: Time and space complexity, Asymptotic Notations, Complexity Analysis Examples. **Linear Data Structures:** Arrays, Stacks, Queues, Linked lists, **Recursion:** Solving recurrences.	
UNIT-II	10 Hours
Non-linear Data Structure: Trees, Traversals, Binary Search Trees, AVL tree, B-trees, B+ Tree, Red Black Tree. **Graph Algorithms:** DFS, BFS, Minimum Spanning Tree Algorithms, Shortest path Algorithms.	

UNIT-III	12 Hours
Sorting and Searching Algorithms: Quick Sort, Merge Sort, Heap sort; Linear Search and Binary Search. **Hashing:** Hashing Functions, Collision Resolution Techniques	
UNIT-IV	10 Hours
Algorithm Strategies: Greedy paradigm with examples. Divide and conquer paradigm with examples. Dynamic-programming paradigm with examples. **NP Completeness:** P, NP, NP-complete, NP-Hard categories of problems, Cook's theorem.	
Text Books	
1	Y. Langsam et. al., "Data Structures using C and C++", Second Edition, PHI, 2015.
2	E. Horowitz, S. Sahani, Anderson-Freed "Fundamentals of Data Structures in C", Second Edition, University Press, 2008
3	T. H. Cormen, C. E. Leiserson, R. L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Ed., PHI, 2011.
Reference Books	
1	R. L. Kruse, B. P. Leung, C. L. Tondo, "Data Structures and program design in C", PHI, 2010.
2	Ellis Horowitz and Sartaz Sahani, "Fundamental of Computer Algorithms", Galgotia Publications, 2009.
3	A. V. Aho, J. E. Hopcroft, J. D. Ullman, "The Design and Analysis of Computer Algorithms", Addition Wesley, 2009.
4	D. E. Knuth, "The Art of Computer Programming", 2nd Ed., Addison Wesley, 2011.

Suggestive List of experiments (More programs can be added to this list)

1. Write a program that Implement Single Link List with following operations: i) Insertion of a node at first node, at any position and at end of list. ii) Deletion of a node at start, at middle and at end of list. iii) Display the link list. iv) Count the number of nodes in the link list. v) Search a node in the link list. vi) Sort the link list. vii) Reverse the link list.
2. Write a program that Implement Stack with all primitive operations by using Array. Implement Queue with all primitive operations by using Array.
3. Write a program that Implement Stack and Queue with all primitive operations by using link list.
4. Write a program that Implement doubly link list with primitive operations: (i) Create a doubly linked list (ii Insert a new node to the left of the node. (iii) Delete the node of a given data. (iv) Display the contents of the list.

5. Write a program that Implement Circular link list with primitive operations. (i) Creation of the Circular list (ii) Insertion of the node (iii) Deletion an element (iv) Display the list
6. Write a program that Implement Binary Search Technique.
7. Write a program that Implement AVL tree and Red Black Tree.
8. Write a program that Implement Binary Tree and its Traversal.
9. Write a program that Implement BFS & DFS over a graph.
10. Write a program that Implement shortest path algorithms.
11. Write a program that Implement Quick Sort, Merge Sort and Heap Sort.
12. Write a program that implement Disjoint Set Data Structure
13. Write a program that implements Knapsack Problem.
14. Write a program that implements Huffman Coding
15. Write a program to implement Prim’s and Kruskal's Algorithm
16. Write a program to implement Matrix Chain Multiplication Algorithm
17. Write a program to implement Longest Common Subsequence Algorithm.
18. Write a program to implement All Pair Shortest Path Algorithm.