COURSE CODE	COURSE TITLE	L	T	P	C
10212CS211	ARTIFICIAL INTELLIGENCE TECHNIQUES	3	0	2	4

A. Preamble

The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. In this course students can explore the various Search techniques like basic, advanced and heuristic approach are universal problem-solving methods to solve a specific problem and provide the best result. Evolutionary algorithms are used to optimize the AI applications. Goal and constraint-based approach, Knowledge reasoning and inference helps the students to develop the decisions making strategies in intelligent agents and expert systems.

B. Prerequisite Course

10210CS101 - Problem Solving using C

C. Course Objectives

Learners are exposed to:

- Understand and apply the basic principles of artificial intelligence concepts towards problem solving, knowledge representation, reasoning, inference and learning.
- Explore the artificial intelligence techniques in intelligent agents, expert systems, artificial neural networks, Genetic algorithms, Heuristics approaches and Evolutionary algorithms models.

D. Course Outcomes

Upon the successful completion of the course, students will be able to:

CO No's	Course Outcomes	K - Level
CO1	Identify the problem-solving techniques by using State Space search and heuristic search approach	K3
CO2	Use evolutionary and advanced search techniques to provide optimized solution.	K3
CO3	Solve unified planning approach and Constraint-satisfaction problems.	K3
CO4	Apply Logical knowledge representation for rules and fact-based approach.	K3
CO5	Build the logic of Uncertainty and Reasoning using resolution methods.	K3
Knowledge Level (Based on revised Bloom's Taxonomy) K1-Remember K2-Understand K3-Apply K4-Analyze K5-Evaluate K6-Create		

E. Correlation of Cos with Program Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3		1	1				1	1		1	2		3
CO2	3	3	3	1	1				1	1		1	2		3

CO3	3	3	3	1	1				1	1		1	2		3
CO4	3	3	2	1	2				1	1		1	1		3
CO5	3	3	3	1	2				1	1		1	2		3

High - 3, Medium - 2, Low – 1

F. Course Contents

Unit 1 Basic and Heuristic Search Techniques L-9 Hours

Artificial Intelligence – Introduction - Evolution of AI- State Space Search: Generate and Test- Simple Search- Depth First Search- Breadth First Search- Comparison of BFS and DFS- - Depth Bounded DFS- Depth First Iterative Deepening. Heuristic Search: Heuristic Functions- Best First Search- Hill Climbing- N-Puzzle Problem- Block-world Problem Local Maxima- Variable Neighborhood Descent- Beam Search- Tabu Search- Case Studies – Stock Market prediction, Weather Forecasting.

Unit 2 Evolutionary and Advanced Search Techniques L-9 Hours

Randomized Search and Emergent Systems: stochastic and evolutionary search algorithms- Simulated Annealing- - Emergent Systems- Ant Colony Optimization- Finding Optimal Paths: The Travelling Salesman Problem- Dynamic Programming- Algorithm A*- Iterative Deepening A* - Min-Max-Alpha-Beta Tree search – Monte Carlo Tree Search - Case Studies – Vehicle Routing and Map Navigation.

Unit 3 Planning and Strategies L-9 Hours

Planning: planning as search, partial order planning - A Unified Planning Framework-Forward and Backwards State Space Planning- Goal Stack Planning- Plan Space Planning- The STRIPS Domain-Constraint Satisfaction Problem: N-Queens-Cryptarithmetic- Graph Coloring- Constraint Propagation-Scene Labeling- Higher Order and Directional Consistency- Algorithm Backtracking: Graph Coloring - Case Studies – Robot traversal- Object recognition- Game Playing.

Unit 4 Knowledge Representation L-9 Hours

Ontologies, foundations of knowledge representation- Logical representation -The Scheme- Frames-Semantic Net- Production rules -Scripts - Inheritance in Taxonomies- Description Logics- Formal Concept Analysis- Conceptual Graphs Knowledge reasoning - Reasoning about objects- relations – events- actions- time and space-reasoning with defaults- Reasoning about knowledge Case studies – chatbot system- recommendation system.

Unit 5 Knowledge Facets and Logic Inferences L-9 Hours

Knowledge Facets: Intelligent agents: reactive, deliberative, goal-driven, utility-driven, and learning agents- The Wumpus World. Logic and Inferences: Formal Logic- Resolution Method in Propositional Logic- Resolution method in First Order Logic- Deductive Retrieval – Horn Clauses – Forward Chaining- Backward Chaining- Uncertain Knowledge- Probabilistic Reasoning- connection to logic-independence- Bayes rule- Bayesian networks- probabilistic inference, Case studies – Multi- Agent Decision Making System- Human Computer Interaction.

Total: 45 Hours

G.Laboratory Experiments **Total: 30 Hours**

Part - 1

Task 1 Implementation of Graph search algorithms (Breadth first search and Depth First Search) using following constraints.

BFS: Pick any node, visit the adjacent unvisited vertex, mark it as visited, display it, and insert it in a queue. If there are no remaining adjacent vertices left, remove the first vertex from the queue. Apply recursion concept to follow the above steps until the queue is empty or the desired node is found.

DFS: Pick any node. If it is unvisited, mark it as visited and recur on all its adjacent nodes. Repeat until all the nodes are visited, or the node to be searched is found.

Tools- Prolog

Task 2 Implementation of Hill climbing algorithm for Heuristic search approach using following constraints in python.

i. Create a function generating all neighbours of a solution
ii. Create a function calculating the length of a route
iii. Create a random solution generator
iv. Create a Travelling salesman problem

Tools- Python, Online Simulator - https://graphonline.ru/en/

Task 3 Implementation of A * Algorithm to find the optimal path using Python by following constraints.

- The goal of the A* algorithm is to find the shortest path from the starting point to the goal point as fast as possible.
- The full path cost (f) for each node is calculated as the distance to the starting node (g) plus the distance to the goal node (h).
- Distances is calculated as the manhattan distance (taxicab geometry) between nodes.

Tools- Python, Online Simulator - https://graphonline.ru/en/

Task 4 Implementation of Mini-Max algorithm uses recursion to search through the game-tree using python by applying following constraints.

- In this algorithm two players play the checker's game; one is called MAX and other is called MIN.
- Both the players fight it as the opponent player gets the minimum benefit while they get the maximum benefit.
- Both Players of the game are opponent of each other, where MAX will select the maximized value and MIN will select the minimized value.
- The minimax algorithm performs a depth-first search algorithm for the exploration of the complete game tree.
- The minimax algorithm proceeds all the way down to the terminal node of the tree, then backtrack the tree as the recursion.

Tools : Python

Task 5 Implementation of Ant Colony Optimization to Optimize Ride-Sharing Trip Duration using Python by following constraints.

- To forecast travel times between every pair of pick-up and drop-off locations.
- To find the shortest route that visits a set of locations.
- To implement optimization techniques are required to intelligently search the solution space and find near-optimal solutions.

Tools: Python

Task 6 Solve a Map Coloring problem using constraint satisfaction approach by applying following constraints

- Assign each territory a color such that no two adjacent territories have the same color by considering following parameters: Domains, Variables and Constraints
- Apply Basic Greedy Coloring Algorithm: Color first vertex with first color, do following for remaining V-1 vertices.
- Consider the currently picked vertex and color it with the lowest numbered color that has not been used on any previously colored vertices adjacent to it. If all previously used colors appear on vertices adjacent to v, assign a new color to it.

Tools- Python, Online Simulator - https://graphonline.ru/en/

Task 7 Implementation of Monkey Banana Problem in Goal Stack planning using prolog by applying following constraints.

Imagine a room containing a monkey, chair and some bananas. That have been hanged from the centre of ceiling. If the monkey is clever enough, he can reach the bananas by placing the chair directly below the bananas and climb on the chair. The problem is to prove the monkey can reach the bananas.The monkey wants it, but cannot jump high enough from the floor. At the window of the room there is a box that the monkey can use. The monkey can perform the

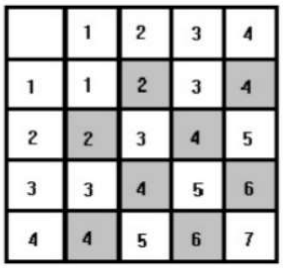
Following actions: -

1) Walk on the floor.
2) Climb the box.
3) Push the box around (if it is beside the box).
4) Grasp the banana if it is standing on the box directly under the banana.

Tools: Prolog

Task 8 Implementation of N-queen problem using backtracking algorithm using prolog

In the 4 Queens problem the object is to place 4 queens on a chessboard in such a way that no queens can capture a piece. This means that no two queens may be placed on the same row, column, or diagonal.



	1	2	3	4
1	1	2	3	4
2	2	3	4	5
3	3	4	5	6
4	4	5	6	7

The n Queens Chessboard.

Tools:Prolog

Task 9 To Build an Intelligent Chatbot system with Python and Dialog-flow using Interactive Text Mining Framework for Exploration of Semantic Flows in Large Corpus of Text.

- To integrate with Google Cloud Speech-to-Text and third-party services such as Google Assistant, Amazon Alexa, and Facebook Messenger.
- Configure Dialogflow to manage your data across GCP services and let you optionally integrate Google Assistant.

Tools- Python, Dialog-flow Framework

Task 10 Implement simple fact for following:

a. Ram likes mango.
b. Seema is a girl.
c. Bill likes Cindy.
d. Rose is red.
e. John owns gold.

Tools-Prolog

Part-2

Use Cases:

Use Case – 1: Implementation of Map navigation using Heuristic search approach in python.
Use Case – 2: Develop a simple neural network for simulating logic gate operations.
Use Case – 3: Solve Multiple criteria production scheduling problem using Genetic algorithm
Use Case – 4: Implementation of Text Lemmatization using NLTK Python Package.
Use Case – 5: Implementation of Intelligent Chat-Bot system using Python and Dialogflow
Use Case – 6: Implementation of Block world problem using STRIPS domain precondition rules in prolog.

Total:75 Hours

G. Learning Resources

i.Text Books:

1. Stuart. J. Russell, et al. “Artificial Intelligence by Pearson: A Modern Approach” 4th Ed, 2020.

ii.Reference Books:

1. Rich and K. Knight,” Artificial Intelligence”, Mc Graw Hill Publishers Inc, 3rd Edition, 2017.
2. Deepak Khemani, “A First Course in Artificial Intelligence”, McGraw Hill Education, 2013.

iii.Online References:

1. “Artificial Intelligence :Search Methods for Problem Solving”, May. 06, 2014 [Online]. Available: https://nptel.ac.in/courses/106/106/106106126/
2. “Artificial Intelligence: Knowledge representation and reasoning” Jan. 19, 2016 [online] Available : https://nptel.ac.in/courses/106/106/106106140/