NIRMA UNIVERSITY

Institute:	Institute of Technology
Name of Programme:	M.Tech. in Electronics & Instrumentation Engineering (Robotics and Artificial Intelligence)
Semester:	I
Course Code:	6CS801CC25
Course Title:	Artificial Intelligence
Course Type:	Core
Year of Introduction:	2025 - 26

L	T	Practical Component				C
		LPW	PW	W	S	
2	-	2	-	-	-	3

Course Learning Outcomes (CLO):

After successful completion of the course, student will be able to –

1. demonstrate awareness and a fundamental understanding of AI techniques in intelligent agents (BL2)
2. summarise the fundamental concepts and principles of reinforcement learning (BL2)
3. apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation and learning (BL3)
4. make use of tabular methods to solve classical control problems (BL3)
5. choose suitable approximation solutions of reinforcement learning. (BL5)

Unit	Contents	Teaching Hours (Total 30)
Unit-I	**Overview** Foundations, scope, problems, and approaches of AI. Intelligent agents: reactive, deliberative, goal-driven, utility-driven, and learning agents, Artificial Intelligence programming techniques.	04
Unit-II	**Problem-solving through Search** Forward and backward, state-space, blind, heuristic, problem reduction, alpha-beta pruning, minimax, constraint propagation, neural, stochastic, and evolutionary search algorithms, sample applications.	05
Unit-III	**Knowledge Representation and Reasoning** Ontologies, foundations of knowledge representation and reasoning, representing and reasoning about objects, relations, events, actions, time, and space; predicate logic, situation calculus, description logics, reasoning with defaults, reasoning about knowledge, sample applications. Planning: planning as search, partial order planning, construction and use of planning graphs. Representing and Reasoning with Uncertain, Applications of AI(vision/robotics etc.)	05

Unit-IV	**Reinforcement Learning** Introduction and Basics of RL, Defining RL Framework, Markov decision process (MDP), state and action value functions, Bellman equations, optimality of value functions and policies, Bellman optimality equations.	04
Unit-V	**Prediction and Control by Dynamic Programming** Overview of dynamic programming for MDP, definition and formulation of planning in MDPs, principle of optimality, iterative policy evaluation, policy iteration, value iteration.	06
Unit-VI	**Monte Carlo Methods for Model Free Prediction and Control** Overview of Monte Carlo methods for model free RL, Monte Carlo control, On policy and off policy learning, Importance sampling, Incremental Monte Carlo Methods for Model Free Prediction.	06

Self-Study: The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents

Suggested Readings/ References:

1. Russell Stuart, Norvig Peter, Artificial Intelligence: A Modern Approach, Prentice Hall.
2. Aleksander Igor, Burnett Piers ,Thinking Machines, Oxford.
3. Bench-Capon T. J. M., Knowledge Representation: An approach to artificial intelligence, Academic Press.
4. Genesereth Michael R., Nilsson Nils J, Logical Foundations of Artificial Intelligence, Morgan Kaufmann.
5. Michael Negnevitsky, Artificial Intelligence: A Guide to Intelligent Systems, Technical Publication
6. Vinod Chandra S.S., Anand Hareendran S, Artificial Intelligence And Machine Learning, PHI
7. Richard S. Sutton and Andrew G. Barto, Reinforcement learning: An introduction, MIT Press
8. Wiering Marco, Martijn Van Otterlo, Reinforcement Learning-Adaptation, learning, and optimization, Springer
9. Dimitri P. Bertsekas, Reinforcement Learning and Optimal Control, Athena Scientific.

Suggested List of Experiments:

Sr. No.	Title	Hours
1.	Implement Depth-First Search (DFS) and Breadth-First Search (BFS) for problem-solving.	02
2.	Implement A* and Greedy Best-First Search for pathfinding in a grid.	02
3.	Implement the Minimax algorithm with alpha-beta pruning for a simple game (e.g., Tic-Tac-Toe).	02
4.	Solve a Constraint Satisfaction Problem (e.g., Sudoku solver) using backtracking.	02

5.	Write a program to develop an agent that takes random actions in a grid world environment.	02
6.	Write a program that constructs an agent with Q-learning algorithm.	02
7.	Create a program that trains an agent using SARSA and Q-learning.	02
8.	Write a program to create a multi-armed bandit problem with multiple arms or actions, with different exploration strategies as epsilon-greedy and UCB.	02
9.	Write a program to design a Markov Decision Process (MDP) and employ the value iteration algorithm to calculate optimal values.	02
10.	Write a program to design a Markov Decision Process (MDP) and employ the policy iteration algorithm to calculate optimal policy.	02
11.	Write a program to develop an agent that takes random actions in a grid world environment.	02
12.	Write a program that constructs an agent with Q-learning algorithm.	02
13.	Create a program that trains an agent using SARSA and Q-learning.	02
14.	Write a program to create a multi-armed bandit problem with multiple arms or actions, with different exploration strategies as epsilon-greedy and UCB.	02

L = Lecture, T = Tutorial, P = Practical, C = Credit

w.e.f. the academic year 2025 - 26 and onwards