AMITY UNIVERSITY
MADHYA PRADESH
Established vide Government of Madhya Pradesh Act No. 27 of 2010

Syllabus

Programme Name: B. Tech. (CSE/IT)							**Session: 2024-28**	
Course Code: CSA501				**Course Name:** REINFORCEMENT LEARNING			**Semester:** V	
Credits (Total)	**L**	**T**	**P**	**Marks (Internal/External)**		**Contact Hours (per week)**	**Independent Study Hour (per week)**	**Section (Group)**
3	2	1	0	30	70	3	3	-
Curriculum level UG						**Basic and applied**	**Student specific course outcome**	**Higher Education Placement Research**

Course Objective:
The objective is to grasp fundamental concepts such as Markov decision processes (MDPs), value functions, policies, and the basic algorithms used in reinforcement learning such as dynamic programming, Monte Carlo methods, and temporal difference learning. Implementation of reinforcement learning algorithms, both model-free (e.g., Q-learning, SARSA) and model-based (e.g., policy iteration, value iteration), and understand their strengths, weaknesses, and applicability.

Course outcomes: After completion of course, the student will be able to:

CO-1	lerstand fundamental concepts and the basic algorithms used in reinforcement learning.
CO-2	Understand various applications of reinforcement learning across different domains such as robotics, game playing, finance, healthcare, and recommendation systems.
CO-3	Understand implementation of reinforcement learning algorithms and understand their strengths, weaknesses, and applicability.
CO-4	Know how to evaluate and compare different reinforcement learning algorithms, understand the importance of exploration-exploitation trade-offs.

CO-5	Critically analyze the societal impact of reinforcement learning technologies and propose responsible development practices.

Teaching Pedagogy:

T1	Classroom teaching (white board), Power Point Presentations, Interactive lectures, Inquiry based teaching
T2	ABL activities, Assignments, Flip Class/ Seminars, Quiz, Oral Viva-voce examination

Assessment Tools

AT1-1	Quiz
AT1-2	Activity Based Learning
AT1-3	Midterm Exams
AT1-4	Flip Class
AT1-5	Seminar Presentation
AT1-6	Assignments
AT1-7	Poster
AT1-8	Oral Viva-voce examination
AT1-9	Industrial Visit Report

Prerequisites: Probability and Statistics, Linear Algebra, Calculus, Algorithms and Data Structures, Machine Learning Basics, Programming Skills

Module wise contents details	**Assessment tools**
Module I: (10 Hours) Introduction to Reinforcement Learning What is RL?, RL algorithm, How RL differs from other ML paradigms Elements of RL: Agent, Policy function,Value function,Model Agent environment interface, Types of RL environment, Deterministic environment Stochastic environment, Fully observable environment, Partially observable environment Discrete environment, Continuous environment, Episodic and non-episodic environment, Single and multi-agent environment, RL platforms, OpenAI Gym and Universe, DeepMind Lab, RL-Glue, Project Malmo, ViZDoom, Applications of RL, Education, Medicine and	Quiz Mid-term Exam Assignment

healthcare, Manufacturing, Inventory management, Finance Natural Language Processing and Computer Vision	
Module II: (9 Hours) Multi-armed Bandits, A k-armed Bandit Problem, Action-value Methods, the 10-armed Testbed, Incremental Implementation, Tracking a Nonstationary Problem, Optimistic Initial Values, Upper-Confidence-Bound Action Selection, Gradient Bandit Algorithms Associative Search (Contextual Bandits) Finite Markov Decision Processes, The Agent–Environment Interface, Goals and Rewards, Returns and Episodes, Unified Notation for Episodic and Continuing Tasks, Policies and Value Functions, Optimal Policies and Optimal Value Functions, Optimality and Approximation, The Bellman equation and optimality, Deriving the Bellman equation for value and Q functions Solving the Bellman equation.	Mid-Term Quiz Assignment
Module III: (9 Hours) Dynamic Programming: Policy Evaluation (Prediction), Policy Improvement, Policy Iteration, Value Iteration, Asynchronous Dynamic Programming, Generalized Policy Iteration, Efficiency of Dynamic Programming, Solving the frozen lake problem Value iteration, Policy iteration Monte Carlo Methods: Monte Carlo Prediction, First visit Monte Carlo, Every visit Monte Carlo Let's play Blackjack with Monte Carlo, Monte Carlo Estimation of Action Values Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Off-policy Monte Carlo Control	Mid-Term Quiz Assignment
Module IV: (8 Hours) Temporal-Difference Learning: TD Prediction, Advantages of TD Prediction Methods, Optimality of TD (0), Sarsa: On-policy TD Control, Q-learning: Off-policy TD Control, Solving the taxi problem using Q learning, Frozen lake solution using Q-learning, Expected Sarsa, Solving the taxi problem using SARSA, Maximization Bias and Double Learning, Games, Afterstates, and Other Special Cases, Deep Q-learning,	Mid-Term Oral Viva-voce examination Seminar Presentation
Module V: (9 Hours) On-policy Prediction with Approximation:Value-function Approximation, The Prediction Objective (VE) , Stochastic-gradient and Semi-gradient Methods , Linear Methods , Feature Construction for Linear Methods, Polynomials , Fourier Basis , Coarse Coding, Tile Coding , Radial Basis Functions , Selecting Step-Size Parameters Manually , Nonlinear Function Approximation: Artificial Neural Networks , Least-Squares TD , Memory-based Function Approximation , Kernel-based Function	Quiz Assignm ent ,Presentat ion

Approximation , Looking Deeper at On-policy Learning: Interest and Emphasis	

Additional Learning:

List of Assignments	**Assignment -1**
	Define reinforcement learning and explain its difference from supervised and unsupervised learning. Explain the key components of a reinforcement learning problem: agent, environment, state, action, reward, policy, and value function. Describe the difference between model-free and model-based reinforcement learning approaches. Discuss the exploration-exploitation trade-off in reinforcement learning and provide examples of exploration strategies. **Assignment -2** Implement a simple reinforcement learning algorithm, such as Q-learning or SARSA, to solve a classic reinforcement learning problem, such as the gridworld environment. Define the environment (e.g., gridworld) with states, actions, rewards, and transition probabilities. 2.Implement the chosen algorithm to learn the optimal policy for navigating the environment. 3. Test the implemented algorithm and evaluate its performance over multiple episodes. 4.Analyze the impact of different hyperparameters (e.g., learning rate, discount factor) on the algorithm's convergence and performance. **Assignment -3** Apply reinforcement learning to a real-world problem or simulated environment of your choice (e.g., a simple game, robotic navigation task, or stock trading simulation). 1. Define the problem and environment, specifying states, actions, rewards, and any additional relevant parameters. 2. Implement a suitable reinforcement learning algorithm to learn a policy for solving the problem. 3.Evaluate the performance of the learned policy and compare it to

	alternative approaches (if applicable). 4.Discuss the limitations and potential improvements of the implemented reinforcement learning solution.
Suggested reading:	**Text:** "Reinforcement Learning: An Introduction", 2nd Edition, Richard S. Sutton and Andrew G. Barto, The MIT Presss,Cambridge, Massachusetts London, England **References:** "Hands-On Reinforcement Learning with Python",1st Edition, Sudharsan Ravichandiran,Pacts Publication "Applied Reinforcement Learning with Python", Ist Edition, Taweh Beysolow II,Apress Publication
Suggested e-resources (Websites /e-books)	https://web.eecs.umich.edu/~baveja/NIPS05RLTutorial/NIPS05RLMainTutorial.pdf

Reinforcement Learning Lab

Course Code: CSA 521 **Credit Unit: 01**

Total Hours: 20

Course Objective:
The main objective is to teach and implement basic methods in Reinforcement Learning including: basics of RL, techniques, algorithms, temporal learning, policy, tools using python, OpenAI and Gym.

SOFTWARE REQUIREMENTS: Python, OpenAI, Gym, Tensorflow, Keras.

List of experiments/demonstrations:

Installation of OpenAI,gym ,Practicals based on theory covered in the lectures using Anaconda Framework, OpenAI Basics,Practicals using OpenAI, gym, keras, tensorflow

Course Outcomes:

Learner will learn

- Basics of python, OpenAI, Gym and deep learning.
- Use of RL tools to implement RL agent and algorithms of RL.

Examination Scheme:

IA			EE			
A	**PR**	**Practical Based Test**	**Major Experiment**	**Minor Experiment**	**LR**	**Viva**
5	10	15	35	15	10	10