Report: Reinforcement of Learning Algorithms and Environments

1. Introduction

I implemented several Reinforcement Learning algorithms and applied them to different environments to understand their behavior, the effect of parameter changes, and the learning process.

2. Implemented Environments

Two environments were used:

- **GridWorld:** A simple environment used to understand basic reinforcement learning algorithms.
- **MountainCar:** A more challenging environment where the agent learns to reach the goal by building momentum.

3. Implemented Algorithms

GridWorld Algorithms:

- Value Iteration
- Policy Iteration
- Monte Carlo
- Temporal Difference (TD)
- SARSA
- Q-Learning

MountainCar Algorithms:

- Monte Carlo
- Temporal Difference (TD)
- SARSA
- Q-Learning

This mapping ensures that each algorithm is used in a suitable environment.

4. Parameter Adjustment

One of the main features of this project is the ability to **adjust algorithm parameters dynamically**.
Each algorithm has its own set of parameters, such as:

- **Gamma (γ)**: Discount factor that controls the importance of future rewards
- **Alpha (α)**: Learning rate
- **Epsilon (ε)**: Exploration rate for ε-greedy policies
- **Episodes**: Number of training episodes

For example:

- Value Iteration and Policy Iteration use **gamma** only
- Monte Carlo uses **gamma, epsilon, and episodes**
- SARSA and Q-Learning use **alpha, gamma, epsilon, and episodes**

This allows users to experiment and observe how learning behavior changes.

5. Visualization

In the visualization part, I used graphs to show how the reinforcement learning algorithms learn over time:

- **Value Function:**
  This graph shows the state value function $V(s)$after training. Higher values indicate better states that are closer to the goal, which helps in understanding how the algorithm evaluates different states in the environment.
- **Reward per Episode:**
  This graph shows the total reward obtained in each episode during training. An increase in reward overtime indicates that the agent is learning and improving its performance. A moving average is also used to show the overall learning trend more clearly.