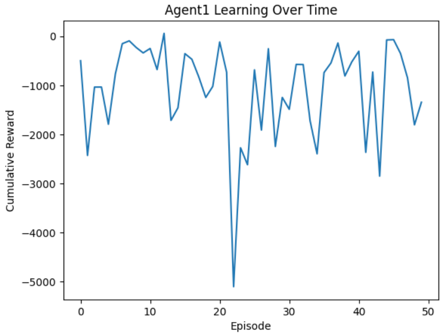
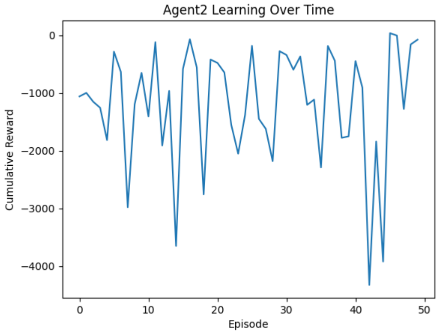
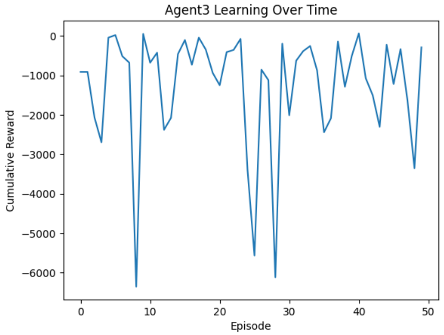
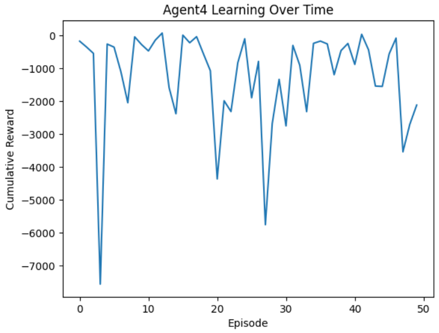
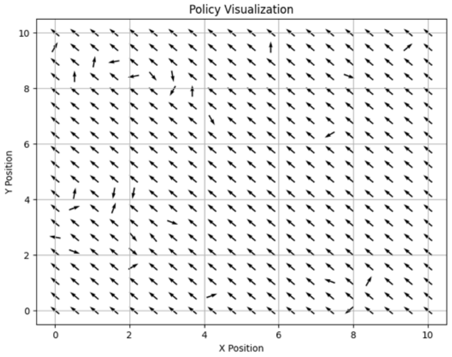
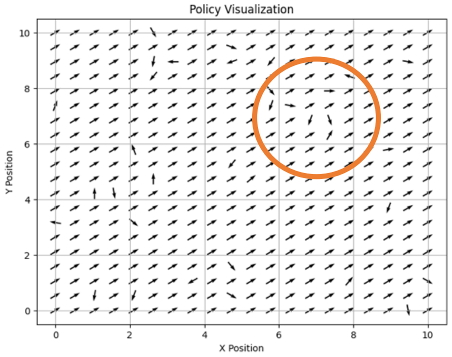
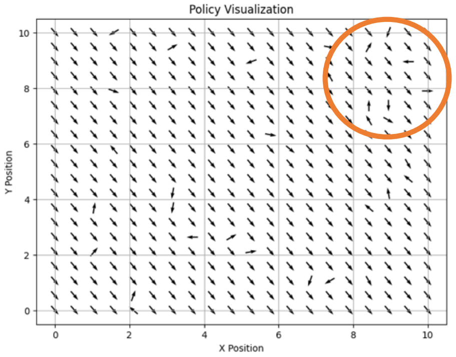
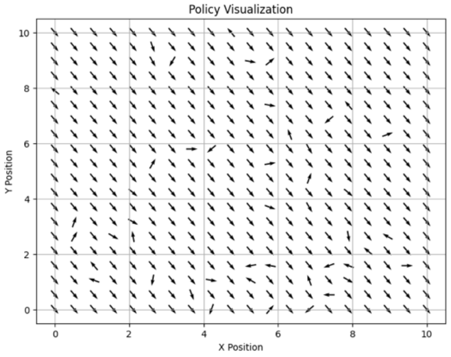
	Monte Carlo	**SARSA**	**Q-Learning**	**General TD(λ)**
Total time for training (second)	4789.87	4144.59	3684.47	5033.94
Training results	 	 	 	
Policy	 	 	 	

(p.s. number of episode = 50)

From the training results, the graphs show:

Agent 1, using Monte Carlo, shows volatility in its learning curve, suggesting that the policy hasn't stabilized. This method's effectiveness can be limited by high variance in returns and slow convergence since it relies on averaging complete returns.

Agent 2, employing SARSA, displays fluctuations in performance, albeit less extreme than Agent 1. There is a minor indication of learning as the curve's lowest points rise slightly towards later episodes. As an on-policy algorithm, SARSA learns a policy considering the exploration; thus, it may exhibit safer learning behaviour that avoids large negative rewards but may not find the optimal policy quickly.

Agent 3, utilizing Q-learning, also has a variable performance with significant dips, implying the exploration of suboptimal actions with high negative rewards. Q-learning is designed to find the optimal policy directly, which might explain the high variance during training. There's a subtle overall improvement, suggesting that while the policy is improving, it might still be far from optimal.

Agent 4, following General TD(λ), presents the most stable learning curve, although it's not devoid of sharp performance drops. This approach can blend the features of Monte Carlo and TD methods, potentially offering a balance between stability and convergence speed depending on the tuning of λ.

After visualize the learned policies:

The provided images show a grid where each cell contains an arrow, which represents the action chosen by a certain policy at each point in the grid. The arrows suggest the direction the agent would move if it were in the corresponding cell.

Agent 1 shows a policy where the arrows mostly point upwards and left across the entire grid. Agent 4 appears to be a general trend of movement towards the right and slightly downwards across the entire grid.

Agent 3 shows a policy where the arrows mostly point downwards and right across the entire grid, with a slight bias towards the upper side of the grid. This suggests a policy that directs the agent to the goal area and possibly to avoid the edges and corners.

Agent 2 has arrows predominantly pointing towards the top right corner, but with a significant number of arrows also pointing directly upwards or to the right in other parts of the grid. This policy might be the most successful to guide the agent to the goal area among the 4 policies.

Across all agents, there's no clear evidence of convergence within the span of 50 episodes. The significant drops in cumulative rewards for all agents indicate substantial exploration. This is a necessary component of learning but should diminish as the agent's policy matures. It's also possible that the agents have not yet found a policy that significantly improves the cumulative reward, or the reward function and environment might be particularly challenging or noisy. These graphs suggest that further training, parameter tuning, and possibly algorithm adjustments are required for better performance. Metrics such as episode length and the variance of returns might offer additional insights into the agents' learning processes. Repetitions of the training with different initial conditions could provide a more robust understanding of each agent's average performance and the stochastic nature of the learning environment.

To summarize, all agents exhibit some degree of learning, with General TD(λ) seeming to provide the most stable but not necessarily the most well performed model. At the same time, it costs much more time among the 4 agents. SARSA and Q-learning cost less time and give better performance, while Monte Carlo's results seem to be less attractive by any standards.