coderunlog

October 8, 2025

1 Assignment 2 - Running log

1.1 Problem 1

Implement DQN on a Cart-pole problem. The code can be found on DQN Implementation notebook

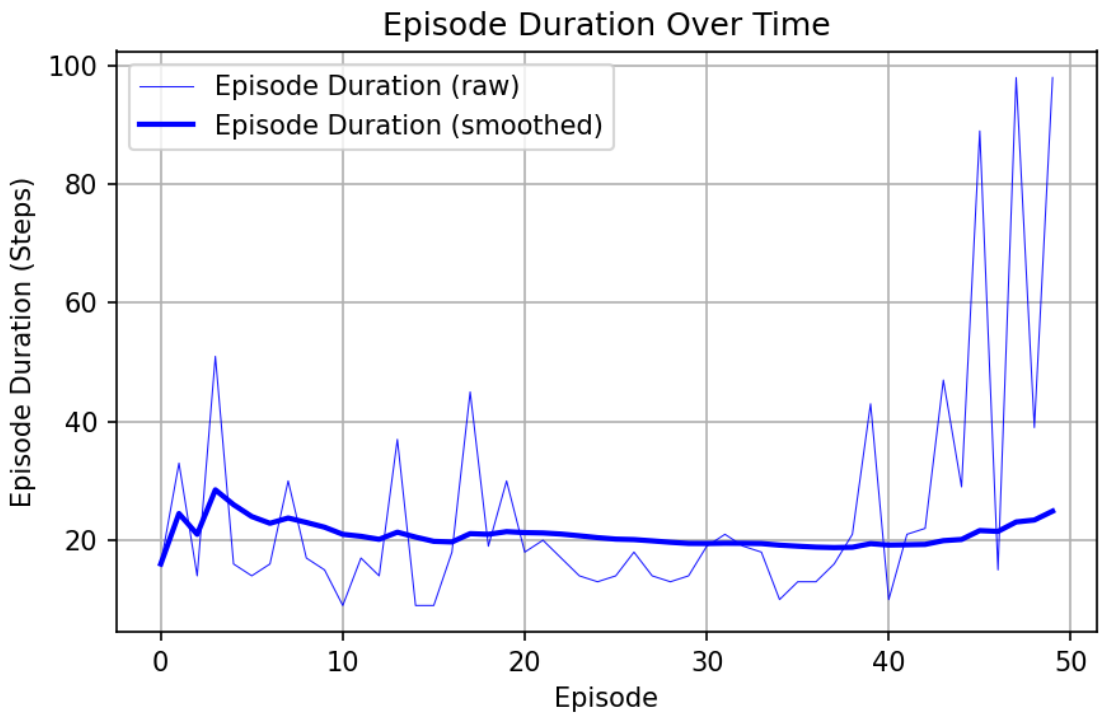
1.2 Prob 1a

Build the tool chain. You can either use the tool chain recommended in the class lecture, i.e., Anaconda + Pytorch +Pycharm, or use your own favorite tool chain. The goal is to implement the given code and obtain a duration-episode plot similar to below.

The code implementation is in `dqn_cartpole.py`. The implementation is using OpenAI's gymnasium, pytorch. To setup the environment with conda. Create a new conda environment. Install pytorch, gymnasium and other necessary packages.

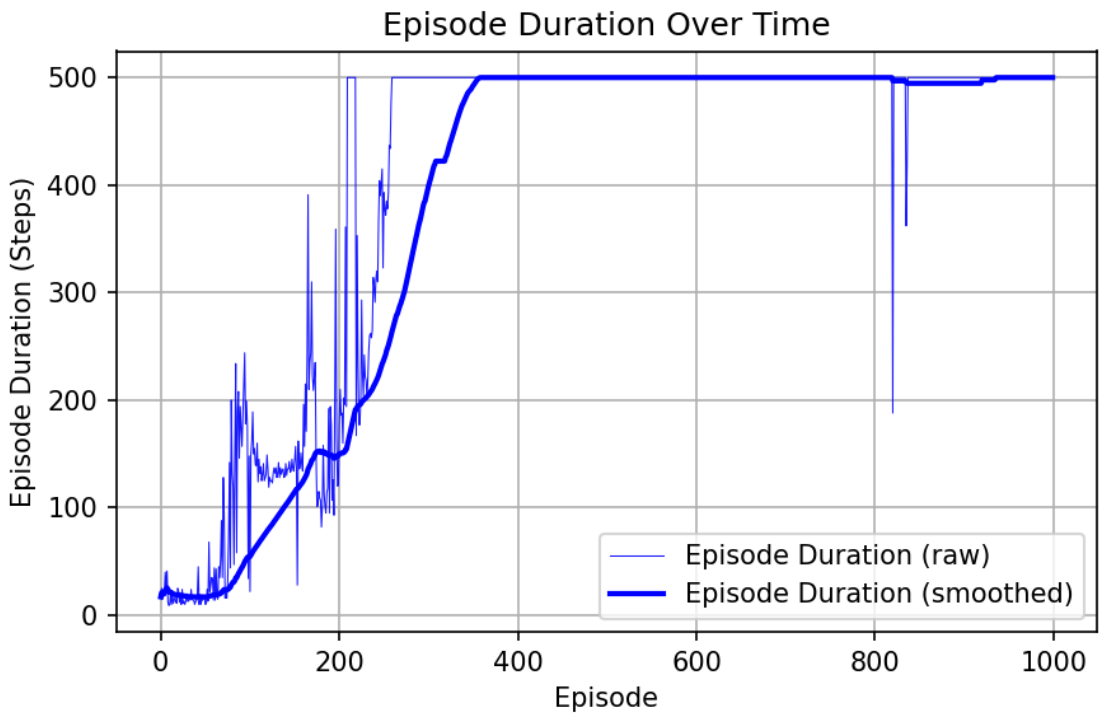
[1]:
```
%matplotlib inline
!python dqn_cartpole.py --problem 1a
#!python dqn_cartpole.py --batch_size 128 --gamma 0.99 --lr 3e-4 --max_episodes␣
↪50

import IPython.display as display
display.display(display.Image(f'images/episode_rewards_1a.png'))
```

```
State space: 4, Action space: 2
INFO:__main__:Training configuration: {'BATCH_SIZE': 128, 'GAMMA': 0.99,
'EPSILON_START': 0.9, 'EPSILON_END': 0.01, 'EPSILON_DECAY': 2500, 'TAU': 0.005,
'TARGET_UPDATE_FREQ': 1, 'LR': 0.0003, 'MEMORY_CAPACITY': 10000, 'MAX_EPISODES':
50, 'logger': <Logger __main__ (INFO)>}
INFO:__main__:Episode 0 reward: 16.0, duration: 16
INFO:__main__:Episode Duration Over Time plot saved to
images/episode_rewards_1a.png
```



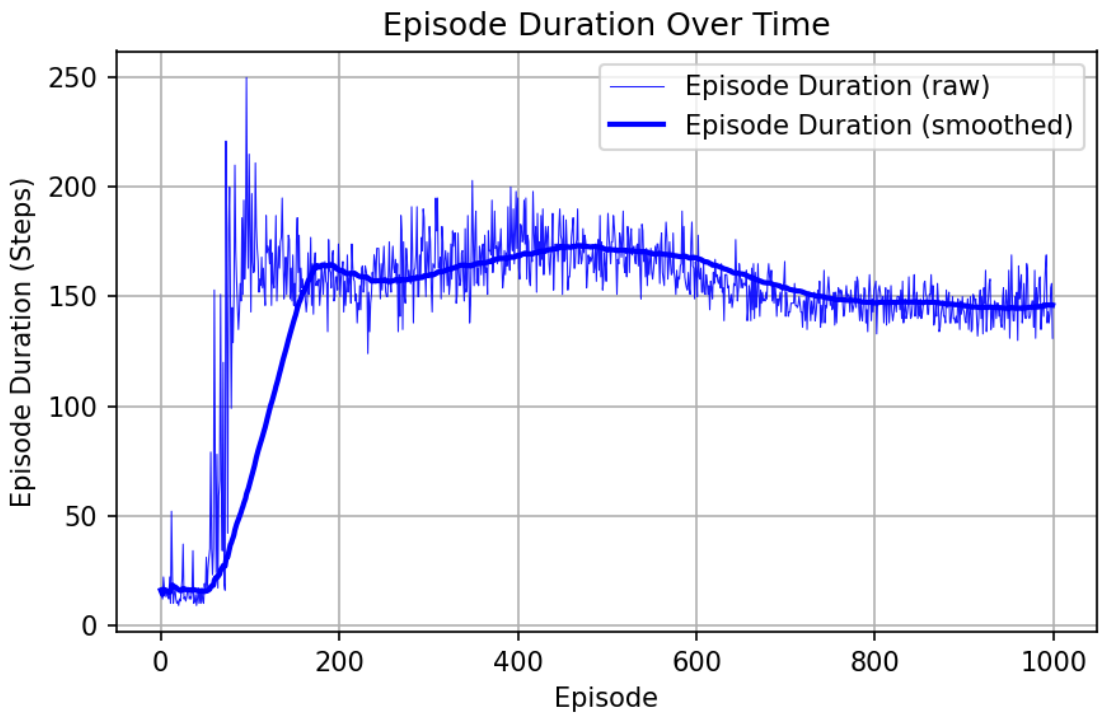
[2]:
```
# Change the episode number from 50 to 1000
!python dqn_cartpole.py --problem 1b
#!python dqn_cartpole.py --batch_size 128 --gamma 0.99 --lr 3e-4 --max_episodes␣
↪1000

display.display(display.Image(f'images/episode_rewards_1b.png'))
```

```
State space: 4, Action space: 2
INFO:__main__:Training configuration: {'BATCH_SIZE': 128, 'GAMMA': 0.99,
'EPSILON_START': 0.9, 'EPSILON_END': 0.01, 'EPSILON_DECAY': 2500, 'TAU': 0.005,
'TARGET_UPDATE_FREQ': 1, 'LR': 0.0003, 'MEMORY_CAPACITY': 10000, 'MAX_EPISODES':
1000, 'logger': <Logger __main__ (INFO)>}
INFO:__main__:Episode 0 reward: 17.0, duration: 17
INFO:__main__:Episode 100 reward: 22.0, duration: 22
INFO:__main__:Episode 200 reward: 180.0, duration: 180
INFO:__main__:Episode 300 reward: 500.0, duration: 500
INFO:__main__:Episode 400 reward: 500.0, duration: 500
INFO:__main__:Episode 500 reward: 500.0, duration: 500
INFO:__main__:Episode 600 reward: 500.0, duration: 500
INFO:__main__:Episode 700 reward: 500.0, duration: 500
INFO:__main__:Episode 800 reward: 500.0, duration: 500
INFO:__main__:Episode 900 reward: 500.0, duration: 500
INFO:__main__:Episode Duration Over Time plot saved to
images/episode_rewards_1b.png
```



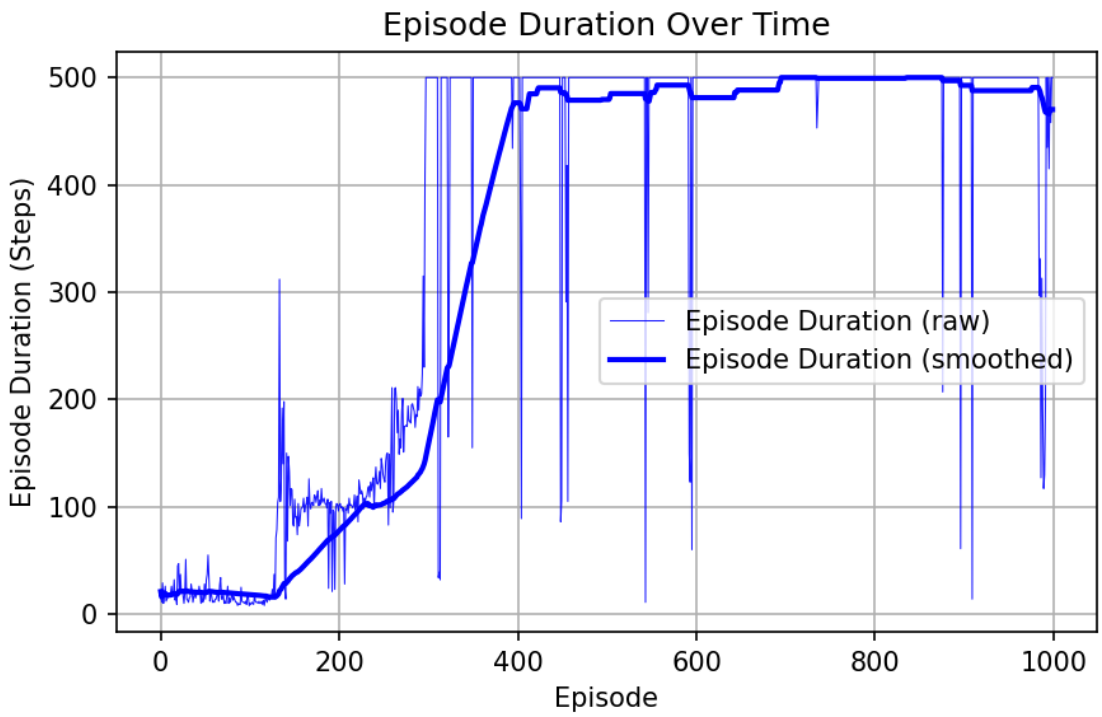
[31]:
```
# Change the gamma from 0.99 to 0.89
!python dqn_cartpole.py --problem 1c
#!python dqn_cartpole.py --batch_size 128 --gamma 0.89 --lr 3e-4 --max_episodes␣
↪1000

display.display(display.Image(f'images/episode_rewards_1c.png'))
```

```
State space: 4, Action space: 2
INFO:__main__:Training configuration: {'BATCH_SIZE': 128, 'GAMMA': 0.89,
'EPSILON_START': 0.9, 'EPSILON_END': 0.01, 'EPSILON_DECAY': 2500, 'TAU': 0.005,
'TARGET_UPDATE_FREQ': 1, 'LR': 0.0003, 'MEMORY_CAPACITY': 10000, 'MAX_EPISODES':
1000, 'logger': <Logger __main__ (INFO)>}
INFO:__main__:Episode 0 reward: 16.0, duration: 16
INFO:__main__:Episode 100 reward: 177.0, duration: 177
INFO:__main__:Episode 200 reward: 163.0, duration: 163
INFO:__main__:Episode 300 reward: 160.0, duration: 160
INFO:__main__:Episode 400 reward: 195.0, duration: 195
INFO:__main__:Episode 500 reward: 174.0, duration: 174
INFO:__main__:Episode 600 reward: 161.0, duration: 161
INFO:__main__:Episode 700 reward: 141.0, duration: 141
INFO:__main__:Episode 800 reward: 155.0, duration: 155
INFO:__main__:Episode 900 reward: 143.0, duration: 143
INFO:__main__:Episode Duration Over Time plot saved to
images/episode_rewards_1c.png
```



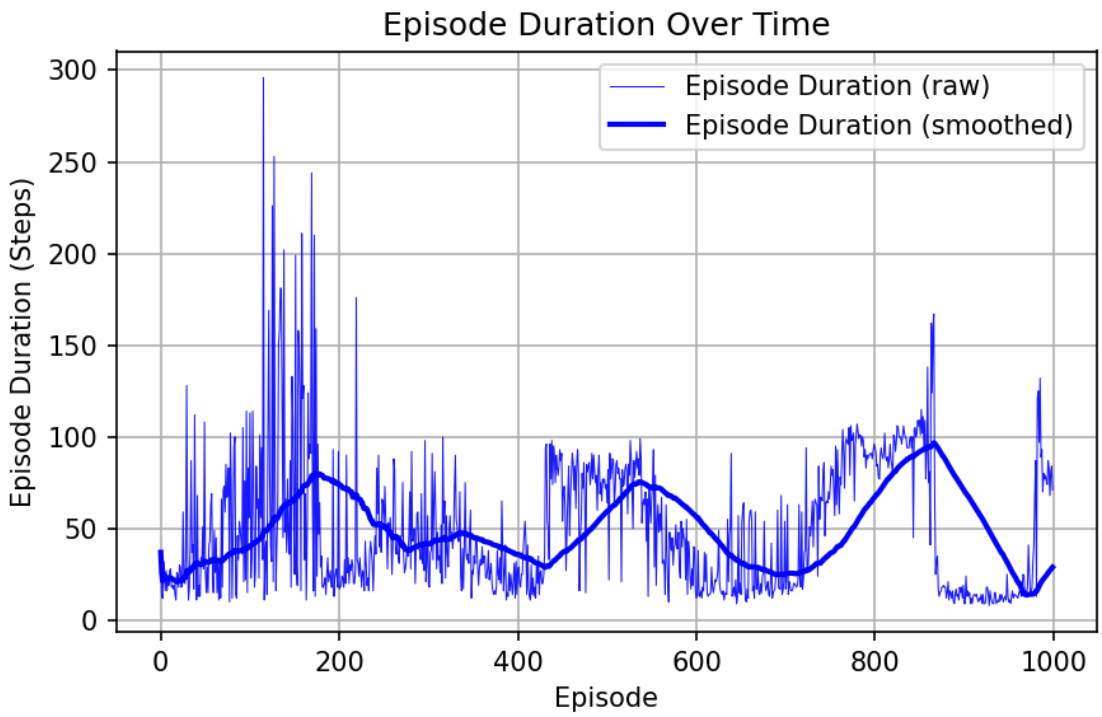
[32]:
```
# Change the batch size from 128 to 1500
!python dqn_cartpole.py --problem 1d
#!python dqn_cartpole.py --batch_size 1500 --gamma 0.99 --lr 3e-4␣
 ↪--max_episodes 1000

display.display(display.Image(f'images/episode_rewards_1d.png'))
```

```
State space: 4, Action space: 2
INFO:__main__:Training configuration: {'BATCH_SIZE': 1500, 'GAMMA': 0.99,
'EPSILON_START': 0.9, 'EPSILON_END': 0.01, 'EPSILON_DECAY': 2500, 'TAU': 0.005,
'TARGET_UPDATE_FREQ': 1, 'LR': 0.0003, 'MEMORY_CAPACITY': 10000, 'MAX_EPISODES':
1000, 'logger': <Logger __main__ (INFO)>}
INFO:__main__:Episode 0 reward: 21.0, duration: 21
INFO:__main__:Episode 100 reward: 10.0, duration: 10
INFO:__main__:Episode 200 reward: 102.0, duration: 102
INFO:__main__:Episode 300 reward: 500.0, duration: 500
INFO:__main__:Episode 400 reward: 500.0, duration: 500
INFO:__main__:Episode 500 reward: 500.0, duration: 500
INFO:__main__:Episode 600 reward: 500.0, duration: 500
INFO:__main__:Episode 700 reward: 500.0, duration: 500
INFO:__main__:Episode 800 reward: 500.0, duration: 500
INFO:__main__:Episode 900 reward: 500.0, duration: 500
INFO:__main__:Episode Duration Over Time plot saved to
images/episode_rewards_1d.png
```



[5]:
```
# Change the learning rate from 3e-4 to 1e-2
!python dqn_cartpole.py --problem 1e
#!python dqn_cartpole.py --batch_size 128 --gamma 0.99 --lr 1e-2 --max_episodes␣
↪1000

display.display(display.Image(f'images/episode_rewards_1e.png'))
```

```
State space: 4, Action space: 2
INFO:__main__:Training configuration: {'BATCH_SIZE': 128, 'GAMMA': 0.99,
'EPSILON_START': 0.9, 'EPSILON_END': 0.01, 'EPSILON_DECAY': 2500, 'TAU': 0.005,
'TARGET_UPDATE_FREQ': 1, 'LR': 0.01, 'MEMORY_CAPACITY': 10000, 'MAX_EPISODES':
1000, 'logger': <Logger __main__ (INFO)>}
INFO:__main__:Episode 0 reward: 37.0, duration: 37
INFO:__main__:Episode 100 reward: 113.0, duration: 113
INFO:__main__:Episode 200 reward: 17.0, duration: 17
INFO:__main__:Episode 300 reward: 19.0, duration: 19
INFO:__main__:Episode 400 reward: 27.0, duration: 27
INFO:__main__:Episode 500 reward: 64.0, duration: 64
INFO:__main__:Episode 600 reward: 12.0, duration: 12
INFO:__main__:Episode 700 reward: 17.0, duration: 17
INFO:__main__:Episode 800 reward: 94.0, duration: 94
INFO:__main__:Episode 900 reward: 9.0, duration: 9
INFO:__main__:Episode Duration Over Time plot saved to
images/episode_rewards_1e.png
```



1.3 Cliff walk example

Grid in which some of the blocks are considered as cliff. The goal is to reach the goal while avoiding the cliff. Write SARSA and Q-Learning code to compare the episodic sum of rewards.

1.4 Prob 2a

Try changing the gamma $\gamma = 0.01, 0.1, 0.5, 0.99, 1$ and plot the episodic sum of rewards.

[6]:
```
# Running Cliff walk example with different gamma values. The plot shows the␣
↪episodic sum of rewards for different gamma values. Using Q-learning and␣
↪SARSA for finding the optimal path.
!python cliff_qlearn_sarsa.py --run_all
```

```
INFO:__main__:Config: Namespace(grid_size=(4, 12), goal_states=[(0, 11)],
start_state=[(0, 0)], cliff_states=[(0, 1), (0, 2), (0, 3), (0, 4), (0, 5), (0,
6), (0, 7), (0, 8), (0, 9), (0, 10)], cliff_reward=-100, step_reward=-1,
goal_reward=100, gamma=0.1, epsilon=0.1, max_episodes=10000,
max_steps_per_episode=500, alpha_qlearning=0.1, alpha_sarsa=0.1, run_all=True)
INFO:__main__:Running Q learning and SARSA with Gamma list: [0.01, 0.1, 0.5,
0.99, 1]
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_0_01_0_1.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
```

```
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0)]
SARSA Path Length: 101 steps
Q-learning Path: [(0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0,
0), (0, 0), (0, 0), (0, 0)]
Q-learning Path Length: 101 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_gamma_0_01.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: -442.69
Q-learning - Final 100 episodes avg: -1670.47
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_0_1_0_1.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3, 1), (3, 0), (3,
1), (3, 0), (3, 1)]
SARSA Path Length: 101 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_gamma_0_1.png

Episode Reward Statistics:
```

```
SARSA - Final 100 episodes avg: -506.93
Q-learning - Final 100 episodes avg: 58.74
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_0_5_0_1.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (3, 0), (3, 1), (3, 2), (3, 3), (3, 4), (3,
5), (3, 6), (3, 7), (3, 8), (3, 9), (3, 10), (3, 11), (2, 11), (1, 11), (0, 11)]
SARSA Path Length: 18 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_gamma_0_5.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: 77.33
Q-learning - Final 100 episodes avg: 54.87
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_0_99_0_1.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (2,
6), (2, 7), (2, 8), (2, 9), (2, 10), (2, 11), (1, 11), (0, 11)]
SARSA Path Length: 16 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_gamma_0_99.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: 75.96
Q-learning - Final 100 episodes avg: 54.71
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_1_0_1.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (2,
6), (2, 7), (2, 8), (2, 9), (2, 10), (2, 11), (1, 11), (0, 11)]
SARSA Path Length: 16 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_gamma_1.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: 79.84
Q-learning - Final 100 episodes avg: 50.23
Optimal reward: 94
INFO:__main__:Running Q learning and SARSA with Epsilon list: [0.01, 0.1, 0.5,
0.99]
INFO:__main__:Saving gridworld with paths to
```

```
images/gridworld_with_paths_0_99_0_01.png
SARSA Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1, 6), (1,
7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
SARSA Path Length: 14 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_epsilon_0_01.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: 85.73
Q-learning - Final 100 episodes avg: 81.42
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/2_gridworld_with_paths_0_99_0_1.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (2,
6), (2, 7), (2, 8), (2, 9), (2, 10), (2, 11), (1, 11), (0, 11)]
SARSA Path Length: 16 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_epsilon_0_1.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: 83.02
Q-learning - Final 100 episodes avg: 52.05
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_0_99_0_5.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (3, 0), (3, 1), (3, 2), (3, 3), (3, 4), (3,
5), (3, 6), (3, 7), (3, 8), (3, 9), (3, 10), (3, 11), (2, 11), (1, 11), (0, 11)]
SARSA Path Length: 18 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_epsilon_0_5.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: 13.42
Q-learning - Final 100 episodes avg: -923.70
Optimal reward: 94
INFO:__main__:Saving gridworld with paths to
images/gridworld_with_paths_0_99_0_99.png
SARSA Path: [(0, 0), (1, 0), (2, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
```

```
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3, 0), (3,
0), (3, 0), (3, 0)]
SARSA Path Length: 101 steps
Q-learning Path: [(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1,
6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (0, 11)]
Q-learning Path Length: 14 steps
INFO:__main__:Saving episode rewards to images/episode_rewards_epsilon_0_99.png

Episode Reward Statistics:
SARSA - Final 100 episodes avg: -4476.97
Q-learning - Final 100 episodes avg: -5078.77
Optimal reward: 94
```

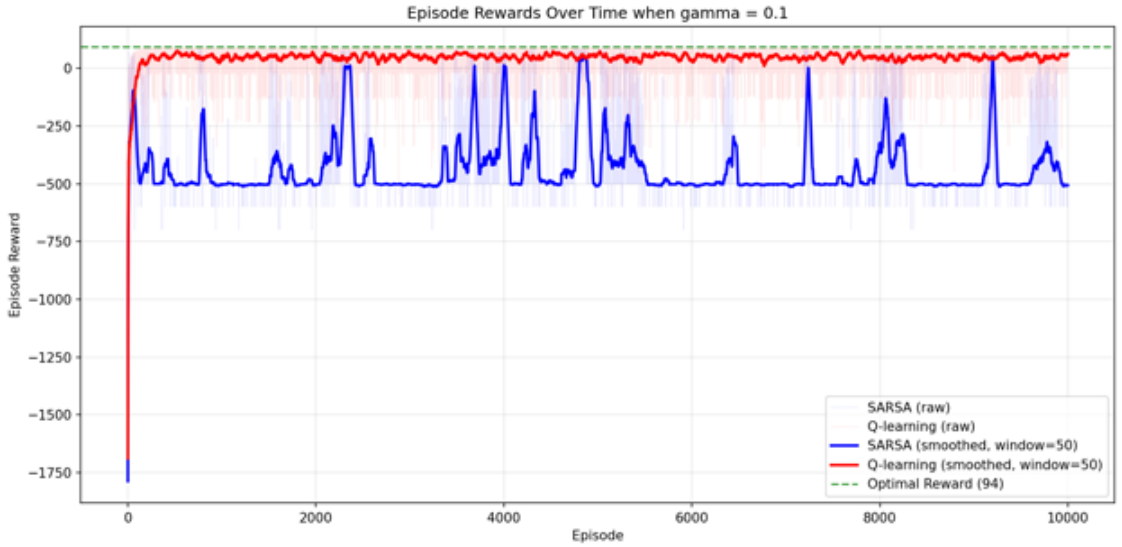
[ ]:

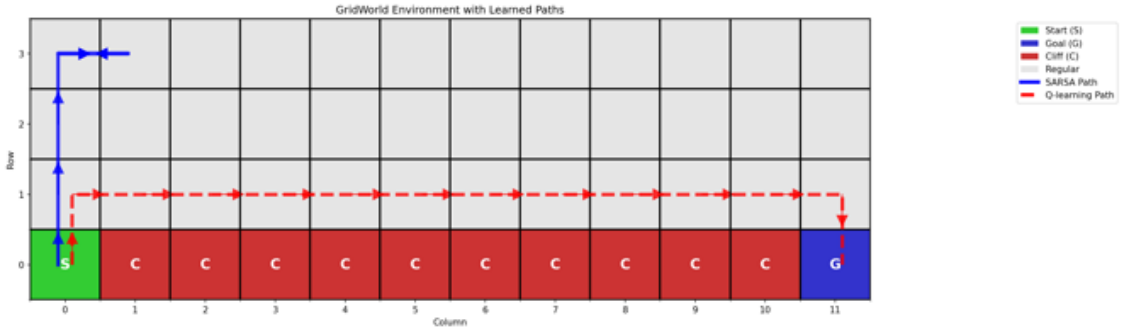
```
!ffmpeg -i images/episode_rewards_gamma_0_01.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_gamma_0_01.png
!ffmpeg -i images/episode_rewards_gamma_0_1.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_gamma_0_1.png
!ffmpeg -i images/episode_rewards_gamma_0_5.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_gamma_0_5.png
!ffmpeg -i images/episode_rewards_gamma_0_99.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_gamma_0_99.png
!ffmpeg -i images/episode_rewards_gamma_1.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_gamma_1.png

!ffmpeg -i images/episode_rewards_epsilon_0_01.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_epsilon_0_01.png
!ffmpeg -i images/episode_rewards_epsilon_0_1.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_epsilon_0_1.png
!ffmpeg -i images/episode_rewards_epsilon_0_5.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_epsilon_0_5.png
!ffmpeg -i images/episode_rewards_epsilon_0_99.png -vf "scale=iw*0.35:ih*0.35"␣
↪images/out/episode_rewards_epsilon_0_99.png


!ffmpeg -i images/gridworld_with_paths_0_01_0_1.png -vf "scale=iw*0.25:ih*0.25"␣
↪images/out/gridworld_with_paths_0_01_0_1.png
!ffmpeg -i images/gridworld_with_paths_0_1_0_1.png -vf "scale=iw*0.25:ih*0.25"␣
↪images/out/gridworld_with_paths_0_1_0_1.png
!ffmpeg -i images/gridworld_with_paths_0_5_0_1.png -vf "scale=iw*0.25:ih*0.25"␣
↪images/out/gridworld_with_paths_0_5_0_1.png
!ffmpeg -i images/gridworld_with_paths_0_99_0_1.png -vf "scale=iw*0.25:ih*0.25"␣
↪images/out/gridworld_with_paths_0_99_0_1.png
```

```
!ffmpeg -i images/gridworld_with_paths_1_0_1.png -vf "scale=iw*0.25:ih*0.25"␣
 ↪images/out/gridworld_with_paths_1_0_1.png

!ffmpeg -i images/gridworld_with_paths_0_99_0_01.png -vf "scale=iw*0.25:ih*0.
 ↪25" images/out/gridworld_with_paths_0_99_0_01.png
!ffmpeg -i images/2_gridworld_with_paths_0_99_0_1.png -vf "scale=iw*0.25:ih*0.
 ↪25" images/out/2_gridworld_with_paths_0_99_0_1.png
!ffmpeg -i images/gridworld_with_paths_0_99_0_5.png -vf "scale=iw*0.25:ih*0.25"␣
 ↪images/out/gridworld_with_paths_0_99_0_5.png
!ffmpeg -i images/gridworld_with_paths_0_99_0_99.png -vf "scale=iw*0.25:ih*0.
 ↪25" images/out/gridworld_with_paths_0_99_0_99.png
```

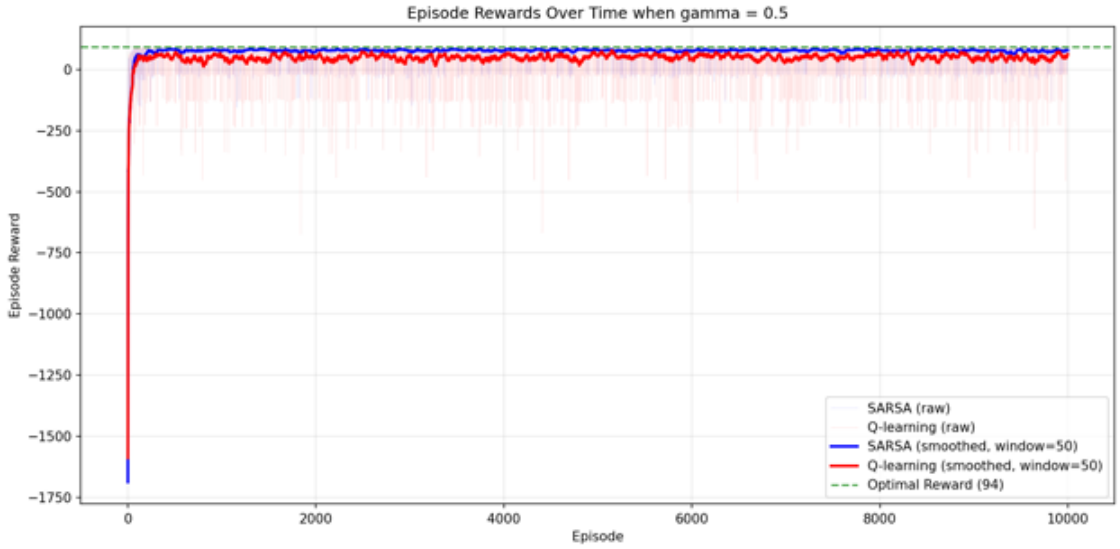
[22]:
```
display.display(display.Image(f'images/out/episode_rewards_gamma_0_01.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_01_0_1.png'))
```

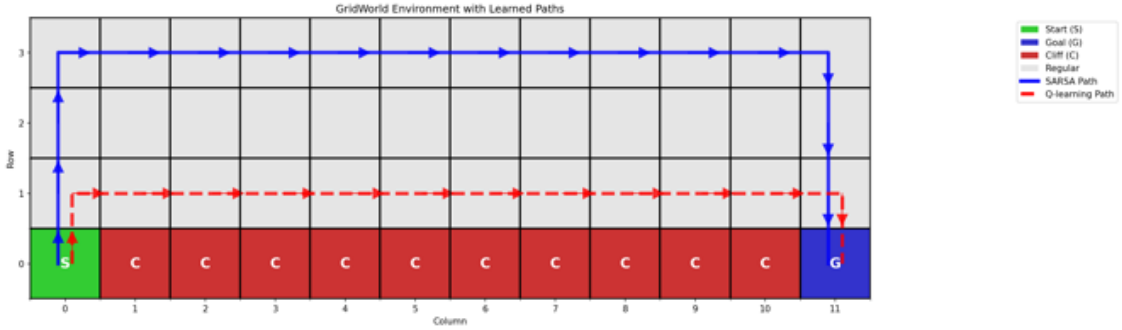
[23]:
```
display.display(display.Image(f'images/out/episode_rewards_gamma_0_1.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_1_0_1.png'))
```



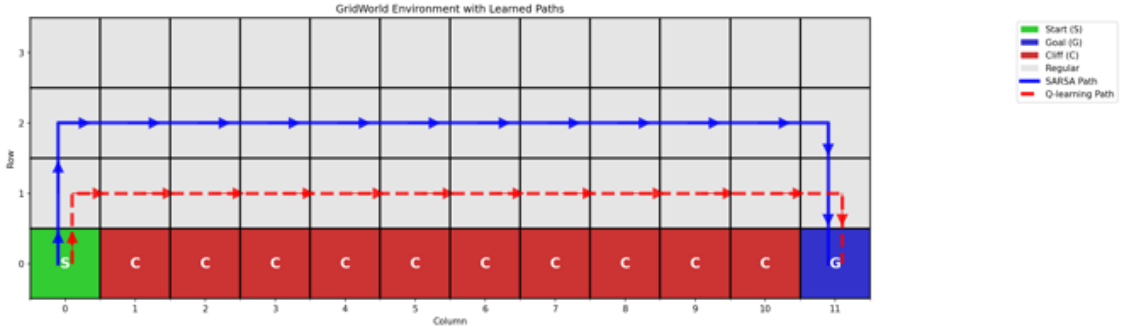


[24]:
```
display.display(display.Image(f'images/out/episode_rewards_gamma_0_5.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_5_0_1.png'))
```

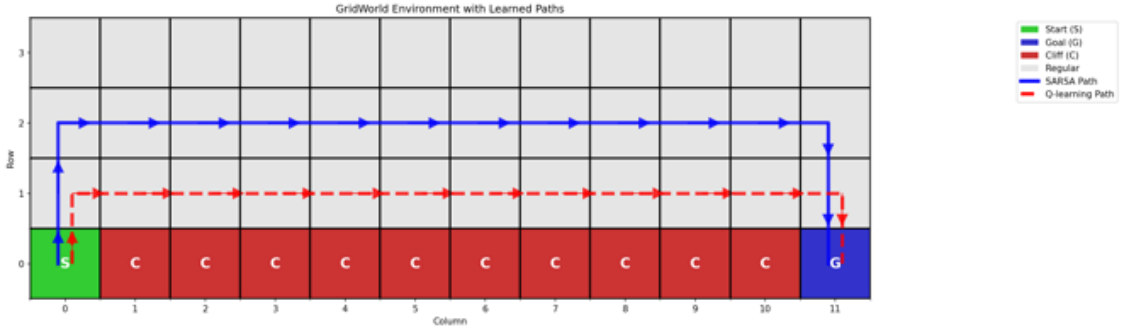




[25]:
```
display.display(display.Image(f'images/out/episode_rewards_gamma_0_99.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_99_0_1.png'))
```

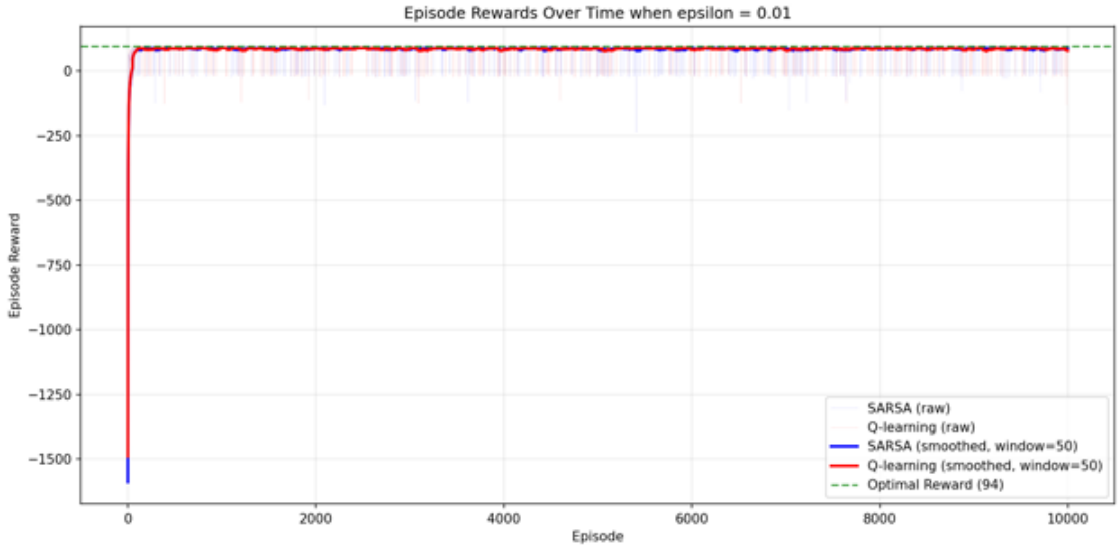


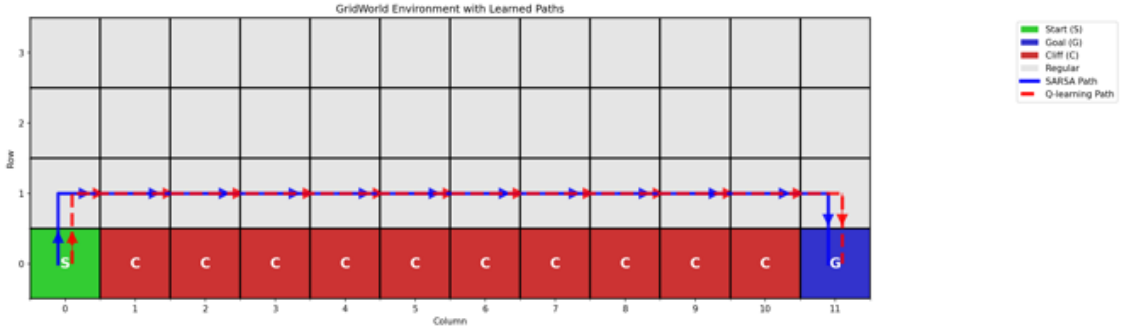
[26]:
```
display.display(display.Image(f'images/out/episode_rewards_gamma_1.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_1_0_1.png'))
```



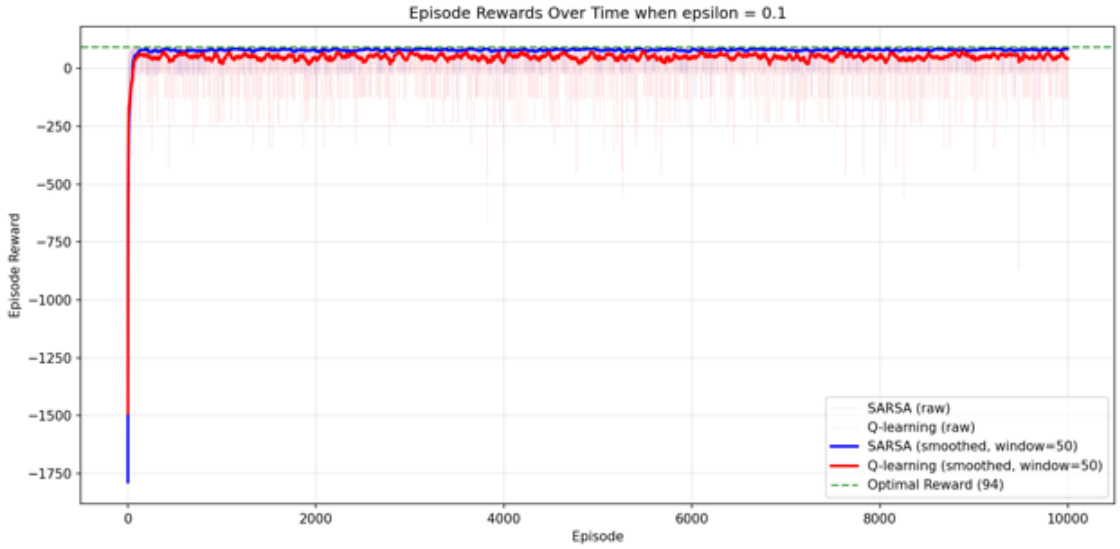
1.5 Keep gamma constant and vary epsilon

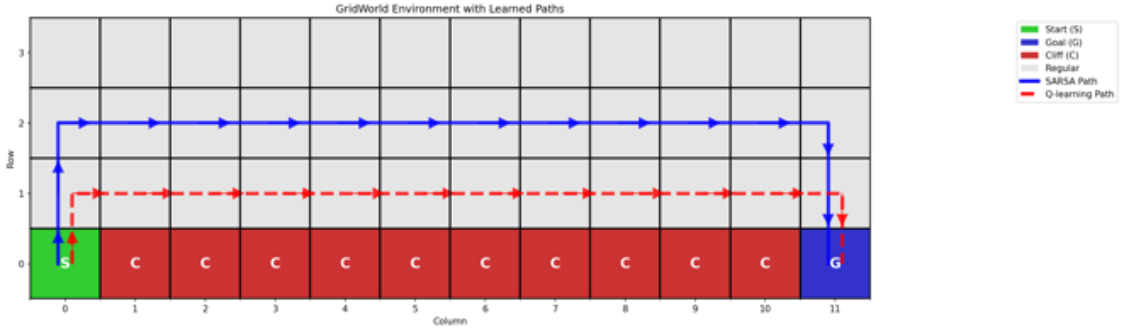
[27]:
```
display.display(display.Image(f'images/out/episode_rewards_epsilon_0_01.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_99_0_01.png'))
```



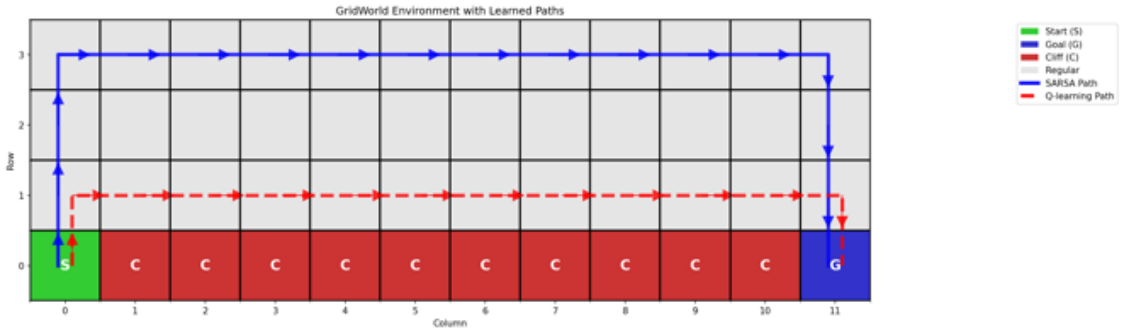


[28]:
```
display.display(display.Image(f'images/out/episode_rewards_epsilon_0_1.png'))
display.display(display.Image(f'images/out/2_gridworld_with_paths_0_99_0_1.
↪png'))
```

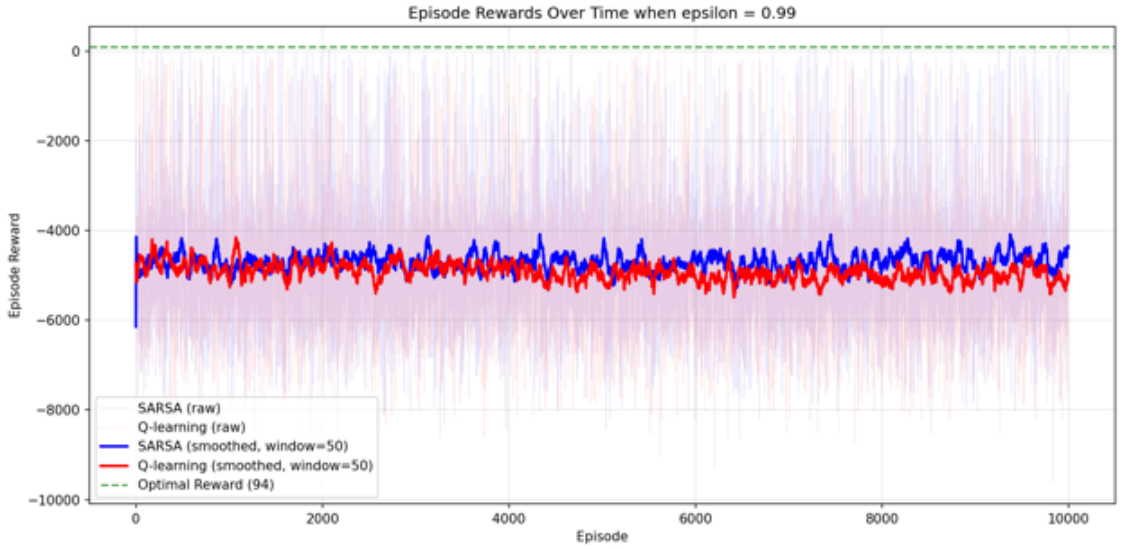


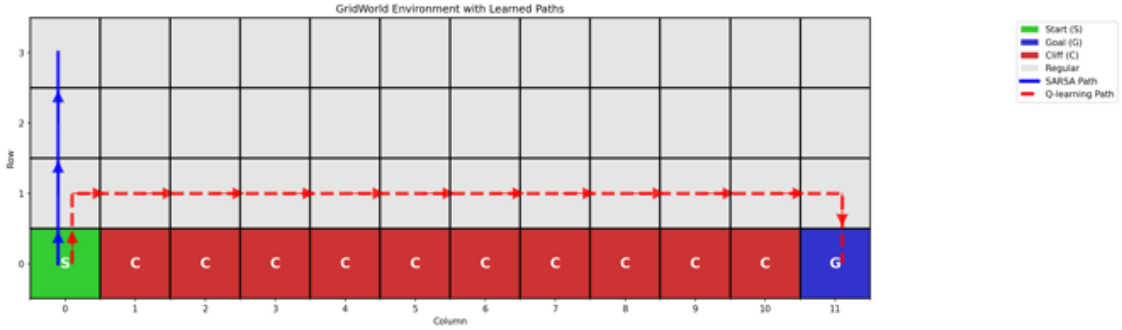


[29]:
```
display.display(display.Image(f'images/out/episode_rewards_epsilon_0_5.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_99_0_5.png'))
```



[30]:
```
display.display(display.Image(f'images/out/episode_rewards_epsilon_0_99.png'))
display.display(display.Image(f'images/out/gridworld_with_paths_0_99_0_99.png'))
```


Episode Rewards Over Time when epsilon = 0.99
Episode Reward
Episode
0
−2000
−4000
−6000
−8000
−10000
0
2000
4000
6000
8000
10000
SARSA (raw)
Q-learning (raw)
SARSA (smoothed, window=50)
Q-learning (smoothed, window=50)
Optimal Reward (94)


GridWorld Environment with Learned Paths
Row
Column
S
C
G
Start (S)
Goal (G)
Cliff (C)
Regular
SARSA Path
Q-learning Path