grid_world_1

June 21, 2025

[1]:
```
%load_ext autoreload
%autoreload 2

import numpy as np
import gymnasium as gym

from grid_world_1 import (
    train_agent,
    print_and_visualize_results,
    test_trained_agent,
)
```

```
/opt/anaconda3/envs/csb320-p4/lib/python3.13/site-packages/pygame/pkgdata.py:25:
UserWarning: pkg_resources is deprecated as an API. See
https://setuptools.pypa.io/en/latest/pkg_resources.html. The pkg_resources
package is slated for removal as early as 2025-11-30. Refrain from using this
package or pin to Setuptools<81.
  from pkg_resources import resource_stream, resource_exists
```

Four sets of parameters are selected.

1 is the default parameters passed, for control.

2 has a higher learning rate and also less epsilon decay (higher value). Maybe this allows for a more "creative" model, that while potentially varies more in outcome, may discover more creative approaches. However, it will train for longer.

3 has a higher learning rate but starts with a lower epsilon value. This model might be more "impatient" and might converge more efficiently, but also may have a greater chance of failure.

4 has a higher epsilon min and greater decay (lower value), allowing it to be "creative" more evenly throughout. Additionally, it has a lower gamma value, pressuring it to get to the rewards quickly.

[2]:
```
params_1 = {
    "alpha": 0.1,
    "gamma": 0.99,
    "epsilon": 1.0,
    "epsilon_decay": 0.995,
    "epsilon_min": 0.01,
    "episodes": 10000,
}
```

```
params_2 = {
    "alpha": 0.2,
    "gamma": 0.99,
    "epsilon": 1.0,
    "epsilon_decay": 0.997,
    "epsilon_min": 0.01,
    "episodes": 20000,
}

params_3 = {
    "alpha": 0.3,
    "gamma": 0.99,
    "epsilon": 0.5,
    "epsilon_decay": 0.995,
    "epsilon_min": 0.01,
    "episodes": 10000,
}

params_4 = {
    "alpha": 0.1,
    "gamma": 0.95,
    "epsilon": 1.0,
    "epsilon_decay": 0.990,
    "epsilon_min": 0.5,
    "episodes": 10000,
}
grid = [params_1, params_2, params_3, params_4]
```

[3]:
```
# Create GridWorld environment (FrozenLake)
env = gym.make("FrozenLake-v1", is_slippery=False)  # Ensure deterministic moves
```

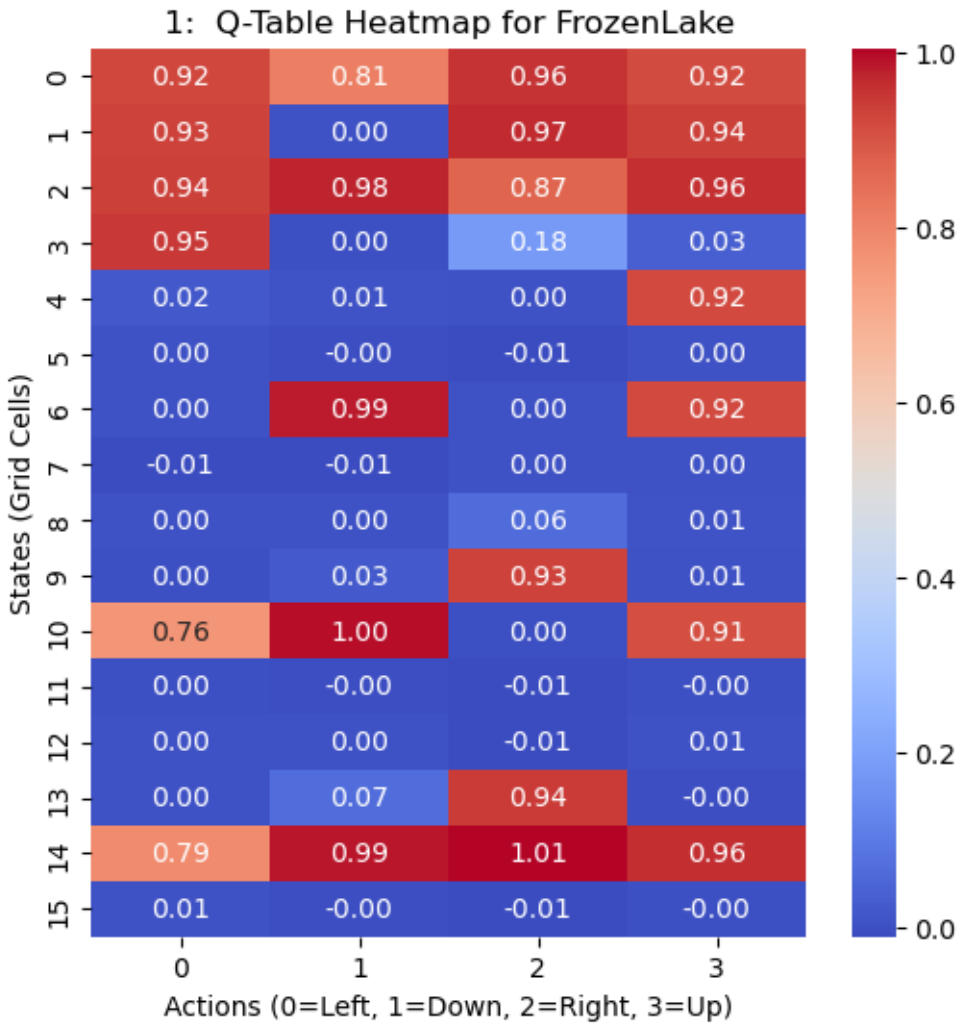
[4]:
```
Qs = []

for i, param in enumerate(grid):
    # Initialize Q-table with small random values to prevent biasing toward 0
    Q = np.random.uniform(
        low=-0.01, high=0.01, size=(
            env.observation_space.n, env.action_space.n
        )
    )
    initial_Q = Q.copy()
    Q, reward_tracking, steps_per_episode, mss = train_agent(initial_Q, Q, env)
    print_and_visualize_results(
        initial_Q, Q, steps_per_episode, mss, text=str(i + 1) + ": "
    )
    Qs.append(Q)
```

```
1:  Initial Q-Table (Before Training):
[[-5.67202499e-03 -3.94572826e-03 -3.20653680e-03  3.40537020e-03]
 [-5.93624417e-03 -4.50906818e-03 -5.46367125e-03  3.69594281e-03]
 [ 6.90598838e-03  8.94128218e-03  4.84198628e-03  6.23575585e-03]
 [-1.53963418e-03 -1.73557972e-03 -1.73505673e-03 -4.99559904e-03]
 [-4.33777554e-03  9.45659852e-03 -3.96447783e-03  1.36521432e-03]
 [ 3.96017958e-03 -3.53575219e-03 -8.66042962e-03  4.64831323e-03]
 [-3.70150771e-03  9.69944058e-03  1.98773334e-03 -1.47902147e-04]
 [-7.63452696e-03 -9.60204051e-03  4.44844294e-03  4.52722568e-03]
 [-7.39029786e-03 -2.24823464e-03 -6.51379936e-03  4.99740667e-03]
 [-9.49178933e-03  9.03736719e-03  1.97921710e-03  6.08385055e-03]
 [-4.88848964e-03 -2.01257928e-03 -2.89367893e-03  7.48812214e-03]
 [ 2.99486758e-05 -2.56333703e-03 -9.09768344e-03 -1.18184920e-03]
 [ 2.53277015e-03  4.86520373e-03 -6.86308838e-03  5.77732544e-03]
 [ 4.40758617e-03 -4.23411911e-03 -5.53357453e-05 -8.41580832e-03]
 [ 9.79434702e-03  5.24453285e-03  7.63119167e-03  8.09456616e-03]
 [ 5.63093305e-03 -3.90395752e-03 -8.13581491e-03 -1.01040969e-03]]

1:  Final Q-Table (After Training):
[[ 9.24234510e-01  8.14000309e-01  9.56291462e-01  9.15654767e-01]
 [ 9.33871838e-01  4.59879626e-03  9.65950971e-01  9.36849780e-01]
 [ 9.36030013e-01  9.75708052e-01  8.67367643e-01  9.60141672e-01]
 [ 9.47269752e-01  2.07315369e-03  1.84082905e-01  3.43735331e-02]
 [ 2.20571867e-02  7.82729445e-03  3.91852618e-03  9.19321494e-01]
 [ 3.96017958e-03 -3.53575219e-03 -8.66042962e-03  4.64831323e-03]
 [ 4.57656094e-03  9.85563689e-01  4.41258208e-03  9.21872945e-01]
 [-7.63452696e-03 -9.60204051e-03  4.44844294e-03  4.52722568e-03]
 [ 4.80772630e-03  4.52363012e-03  6.17909603e-02  7.50632898e-03]
 [ 1.48224836e-03  2.58097938e-02  9.32023054e-01  5.23979130e-03]
 [ 7.62826891e-01  9.95518877e-01  6.68595785e-06  9.13207655e-01]
 [ 2.99486758e-05 -2.56333703e-03 -9.09768344e-03 -1.18184920e-03]
 [ 2.53277015e-03  4.86520373e-03 -6.86308838e-03  5.77732544e-03]
 [ 4.94484937e-03  6.85194238e-02  9.43617161e-01 -4.10219119e-03]
 [ 7.85461349e-01  9.89344582e-01  1.00557462e+00  9.64422930e-01]
 [ 5.63093305e-03 -3.90395752e-03 -8.13581491e-03 -1.01040969e-03]]

1:  Mean Sum of Squares (MSS) between initial and final Q-table: 0.322966
1:  Average Steps to Reach Goal: 6.17
```



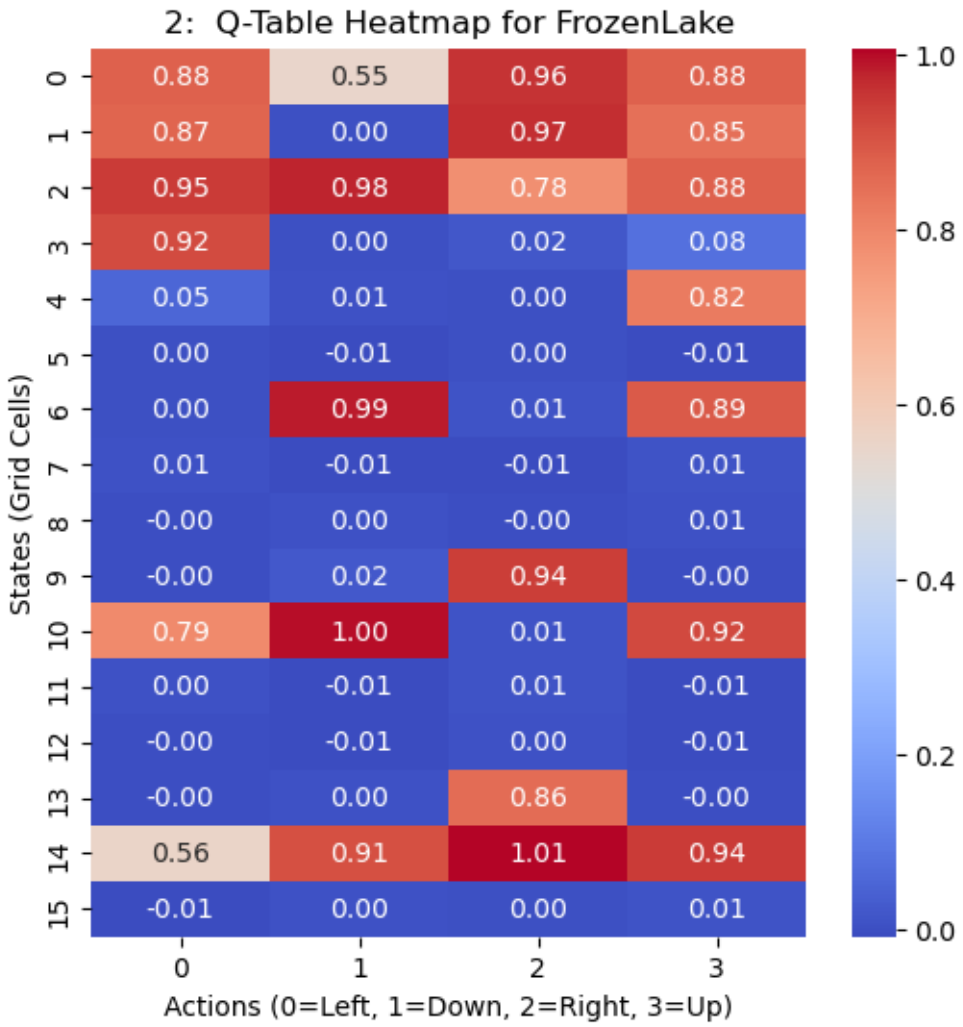
2: Initial Q-Table (Before Training):

```
[[ 0.00295821  0.00876581 -0.00929237 -0.00766454]
 [ 0.00739638  0.00750129 -0.00959037  0.00653542]
 [ 0.00721719  0.00599812  0.00359552 -0.0060092 ]
 [ 0.00631643 -0.00361628 -0.00955712  0.00144412]
 [ 0.00590008  0.00583773 -0.00630731  0.0085792 ]
 [ 0.00087596 -0.00500327  0.00129297 -0.00517895]
 [ 0.00875591 -0.00068847  0.00653321 -0.00440081]
 [ 0.00710722 -0.0079881  -0.00819809  0.00828461]
 [-0.0083391   0.00413089 -0.0001486  -0.00037903]
 [-0.00689094 -0.00168991 -0.00453186 -0.0031989 ]
 [ 0.00086428 -0.00282137 -0.00637036 -0.00810954]
 [ 0.00427282 -0.00559822  0.00995917 -0.00862897]
 [-0.00135075 -0.00861533  0.0043968  -0.00874714]]
```

```
 [-0.00353202 -0.0037433  -0.00884087 -0.00153247]
 [-0.00616533 -0.00737904  0.00645831  0.00039823]
 [-0.00703948  0.00496676  0.00102981  0.00741038]]

2:  Final Q-Table (After Training):
[[ 8.78423642e-01  5.50699577e-01  9.57966777e-01  8.77787198e-01]
 [ 8.71794728e-01  1.28039031e-03  9.67643209e-01  8.46121681e-01]
 [ 9.46441377e-01  9.77417383e-01  7.77478274e-01  8.79545210e-01]
 [ 9.18364635e-01  3.11448472e-03  1.68982492e-02  7.52640532e-02]
 [ 5.11224275e-02  5.31871438e-03  1.26777372e-03  8.21205671e-01]
 [ 8.75955583e-04 -5.00327282e-03  1.29297457e-03 -5.17895416e-03]
 [ 1.51107270e-03  9.87290286e-01  8.19980049e-03  8.89421572e-01]
 [ 7.10721961e-03 -7.98809511e-03 -8.19809445e-03  8.28461389e-03]
 [-1.64592735e-03  4.34833532e-03 -8.17929390e-04  6.34445225e-03]
 [-4.77013198e-03  2.23706400e-02  9.43004524e-01 -1.98510407e-03]
 [ 7.92884903e-01  9.97262915e-01  8.56497677e-03  9.20153630e-01]
 [ 4.27282399e-03 -5.59822324e-03  9.95917362e-03 -8.62897096e-03]
 [-1.35075032e-03 -8.61532830e-03  4.39680335e-03 -8.74714353e-03]
 [-2.74353776e-03  4.26537086e-03  8.55010604e-01 -1.54327516e-03]
 [ 5.55643543e-01  9.08388097e-01  1.00733628e+00  9.44669064e-01]
 [-7.03947679e-03  4.96676250e-03  1.02980967e-03  7.41038176e-03]]

2:  Mean Sum of Squares (MSS) between initial and final Q-table: 0.294581
2:  Average Steps to Reach Goal: 6.25
```

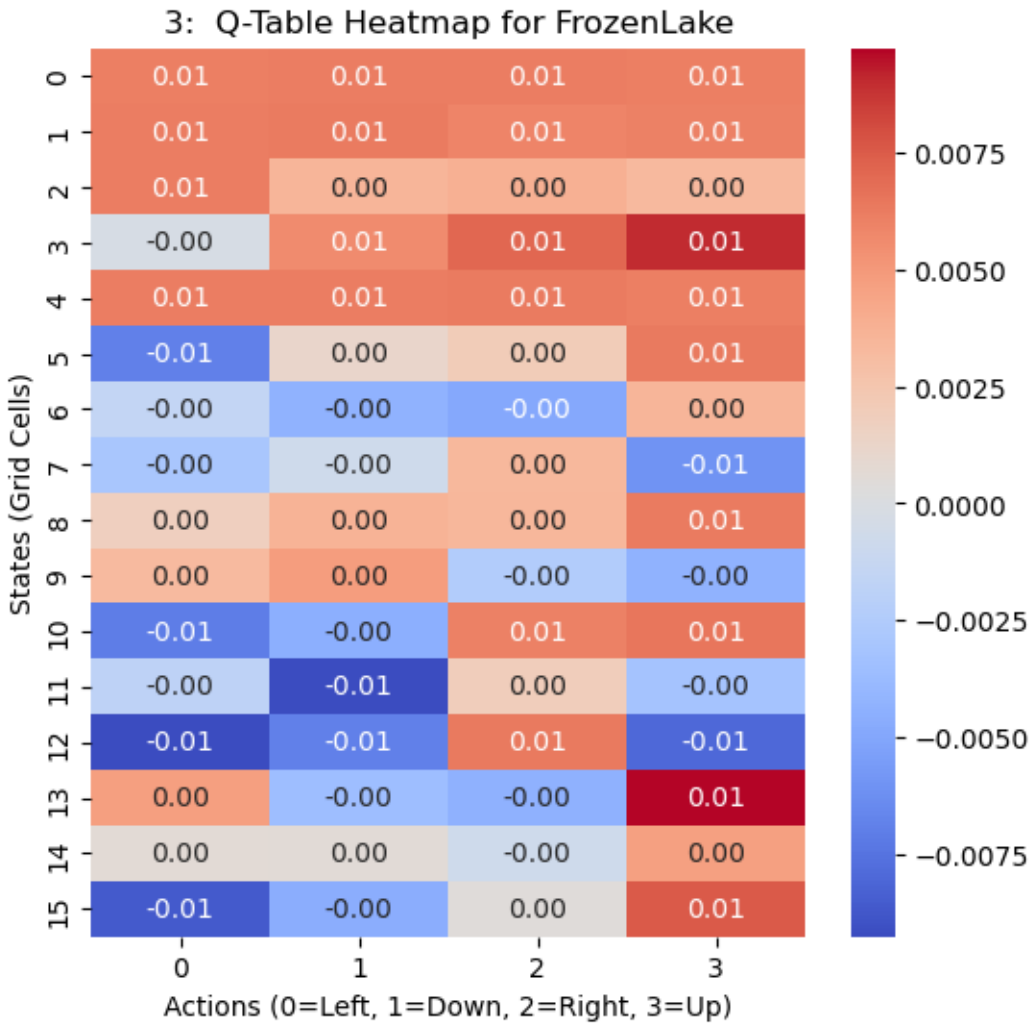



```
3:  Initial Q-Table (Before Training):
[[-0.0062096   0.0071491   0.00561701  0.00179886]
 [ 0.00510335  0.00459257  0.00150779 -0.00935724]
 [ 0.00321045  0.0049334  -0.00758404  0.00042217]
 [-0.00286532  0.00799073  0.00613428  0.00919113]
 [-0.00148257 -0.00436504  0.00945288 -0.00906149]
 [-0.00689567  0.00114654  0.00203732  0.00635089]
 [-0.00438466 -0.00839595 -0.00687427  0.00273436]
 [-0.00297962 -0.00075618  0.00340491 -0.00602926]
 [-0.00921117 -0.00121564  0.00319344  0.00747672]
 [ 0.00325602  0.00361552 -0.00250678 -0.00552022]
 [-0.00706493 -0.00667405  0.0070237   0.0064888 ]
 [-0.00182889 -0.00925796  0.00193535 -0.00320928]
 [-0.00922504 -0.00692847  0.00637141 -0.00799789]
```

```
 [ 0.00411937 -0.00358377 -0.00535821  0.00972664]
 [-0.00142224  0.00057746 -0.00074849  0.00440945]
 [-0.00865343 -0.00464015  0.00039644  0.00754922]]

3:  Final Q-Table (After Training):
[[ 0.00616363  0.00622451  0.00622315  0.00616324]
 [ 0.00621621  0.00628705  0.00593635  0.00608929]
 [ 0.00621207  0.00351853  0.00379941  0.00334722]
 [-0.00025697  0.00558053  0.0071082   0.00902709]
 [ 0.00619768  0.00623322  0.00628738  0.00616345]
 [-0.00689567  0.00114654  0.00203732  0.00635089]
 [-0.00149254 -0.00433234 -0.0049277   0.00356428]
 [-0.00297962 -0.00075618  0.00340491 -0.00602926]
 [ 0.00178397  0.00368447  0.00346473  0.00625183]
 [ 0.00325602  0.00475815 -0.00250678 -0.00433946]
 [-0.00706493 -0.00455638  0.00605323  0.0064888 ]
 [-0.00182889 -0.00925796  0.00193535 -0.00320928]
 [-0.00922504 -0.00692847  0.00637141 -0.00799789]
 [ 0.00471241 -0.00358377 -0.00438586  0.00972664]
 [ 0.00067756  0.00057746 -0.00074849  0.00461328]
 [-0.00865343 -0.00464015  0.00039644  0.00754922]]

3:  Mean Sum of Squares (MSS) between initial and final Q-table: 0.000019
3:  Average Steps to Reach Goal: 2.23
```

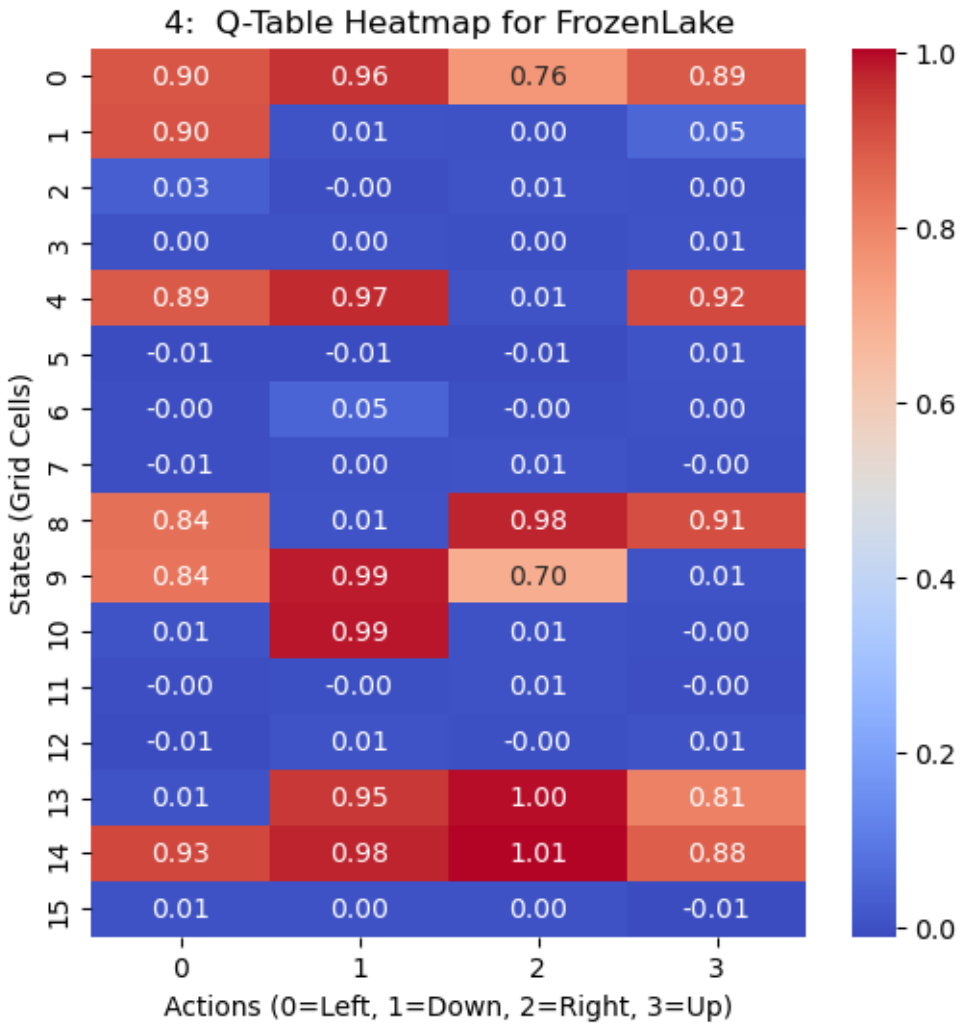



```
4:  Initial Q-Table (Before Training):
[[ 4.89068786e-03  9.12570161e-03 -9.84167223e-03 -9.47564609e-03]
 [ 3.58848642e-03 -5.86658746e-03 -5.82125442e-03  1.18589574e-03]
 [ 6.42907969e-03 -8.78390082e-03  6.79177137e-03 -1.04423864e-03]
 [ 3.13072729e-03 -2.55539733e-03 -5.07583346e-03  6.37097486e-03]
 [ 8.24364587e-04 -9.53991279e-03 -9.75171164e-03  6.52122078e-03]
 [-8.06664408e-03 -7.22290675e-03 -9.86460331e-03  8.98153410e-03]
 [-4.86874367e-03  5.03601350e-03 -1.87414795e-03  1.03129012e-03]
 [-5.46615839e-03  3.68448705e-03  6.20215052e-03 -3.53451464e-06]
 [ 5.27656875e-03  8.29415507e-03 -9.57480509e-04 -1.16306748e-04]
 [ 4.94261723e-04  7.56531933e-03  1.94323520e-03  1.72508639e-03]
 [ 8.75757324e-03 -6.99248932e-03  6.00751876e-03 -3.36922528e-03]
 [-2.39927468e-03 -1.73972412e-03  5.54406824e-03 -2.80191097e-03]
 [-8.13154911e-03  9.15816381e-03 -9.86035936e-04  9.64681394e-03]
```

```
 [-9.26880785e-03  4.71072054e-03  6.62020449e-03 -4.96892039e-03]
 [-3.78951392e-03 -2.36514925e-03 -5.70161260e-03  3.86001171e-03]
 [ 5.65729122e-03  4.39450908e-03  3.67562677e-03 -8.57756636e-03]]

4:  Final Q-Table (After Training):
[[ 8.96743897e-01  9.56316277e-01  7.56939603e-01  8.89875613e-01]
 [ 9.02830333e-01  7.27687981e-03  4.65260827e-03  5.03937240e-02]
 [ 3.29867284e-02 -4.98399640e-03  6.38245682e-03  2.11554699e-03]
 [ 4.48035074e-03  2.77130394e-03  8.32075435e-04  6.25100962e-03]
 [ 8.89570586e-01  9.65976038e-01  8.89171796e-03  9.22127431e-01]
 [-8.06664408e-03 -7.22290675e-03 -9.86460331e-03  8.98153410e-03]
 [-4.86874367e-03  4.80933982e-02 -1.07272025e-03  2.08678900e-03]
 [-5.46615839e-03  3.68448705e-03  6.20215052e-03 -3.53451464e-06]
 [ 8.43730623e-01  9.54982938e-03  9.75733371e-01  9.12657427e-01]
 [ 8.36637741e-01  9.85589264e-01  6.98058158e-01  8.69239437e-03]
 [ 9.03684546e-03  9.87265792e-01  5.90892943e-03 -1.71344310e-03]
 [-2.39927468e-03 -1.73972412e-03  5.54406824e-03 -2.80191097e-03]
 [-8.13154911e-03  9.15816381e-03 -9.86035936e-04  9.64681394e-03]
 [ 7.49117303e-03  9.49851340e-01  9.95544711e-01  8.09300308e-01]
 [ 9.25999223e-01  9.75613600e-01  1.00560072e+00  8.82108544e-01]
 [ 5.65729122e-03  4.39450908e-03  3.67562677e-03 -8.57756636e-03]]

4:  Mean Sum of Squares (MSS) between initial and final Q-table: 0.285188
4:  Average Steps to Reach Goal: 6.02
```



Here, 3 didn't converge, but the other three did. Let's now test the agents for all parameters:

[5]:
```
for Q in Qs:
    test_trained_agent(env, Q)
```

```
Test Episode 1
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 2
```

```
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 3
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 4
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 5
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 1
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 2
Chosen action: 2
Chosen action: 2
Chosen action: 1
```

```
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 3
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 4
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 5
Chosen action: 2
Chosen action: 2
Chosen action: 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 6

Test Episode 1
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 2

Test Episode 2
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 2

Test Episode 3
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 2

Test Episode 4
```

```
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 2

Test Episode 5
Chosen action: 1
Chosen action: 2
Steps taken in this episode: 2

Test Episode 1
Chosen action: 1
Chosen action: 1
Chosen action: 2
Chosen action: 1
Chosen action: 2
Chosen action: 2
Steps taken in this episode: 6

Test Episode 2
Chosen action: 1
Chosen action: 1
Chosen action: 2
Chosen action: 1
Chosen action: 2
Chosen action: 2
Steps taken in this episode: 6

Test Episode 3
Chosen action: 1
Chosen action: 1
Chosen action: 2
Chosen action: 1
Chosen action: 2
Chosen action: 2
Steps taken in this episode: 6

Test Episode 4
Chosen action: 1
Chosen action: 1
Chosen action: 2
Chosen action: 1
Chosen action: 2
Chosen action: 2
Steps taken in this episode: 6

Test Episode 5
Chosen action: 1
Chosen action: 1
```

```
Chosen action: 2
Chosen action: 1
Chosen action: 2
Chosen action: 2
Steps taken in this episode: 6
/opt/anaconda3/envs/csb320-p4/lib/python3.13/site-
packages/gymnasium/envs/toy_text/frozen_lake.py:334: UserWarning: WARN: You
are calling render method without specifying any render mode. You can specify
the render_mode at initialization, e.g. gym.make("FrozenLake-v1",
render_mode="rgb_array")
  gym.logger.warn(
```

[ ]: