UMC-203 Assignment 3

Sehaj Ganjoo
sehajganjoo@iisc.ac.in
Sr No: 23651

April 8, 2025

Files included:

- `23651_disabled_1.pkl` (Scenario 1, Configuration 1)
- `23651_disabled_2.pkl` (Scenario 1, Configuration 2)
- `23651_enabled_1.pkl` (Scenario 2, Configuration 1)
- `23651_enabled_2.pkl` (Scenario 2, Configuration 2)
- 23651_Assignment3_report.pdf (This report)
- 23651_Assignment3.ipynb (Jupyter Notebook)

Training the Q-Learning Agent

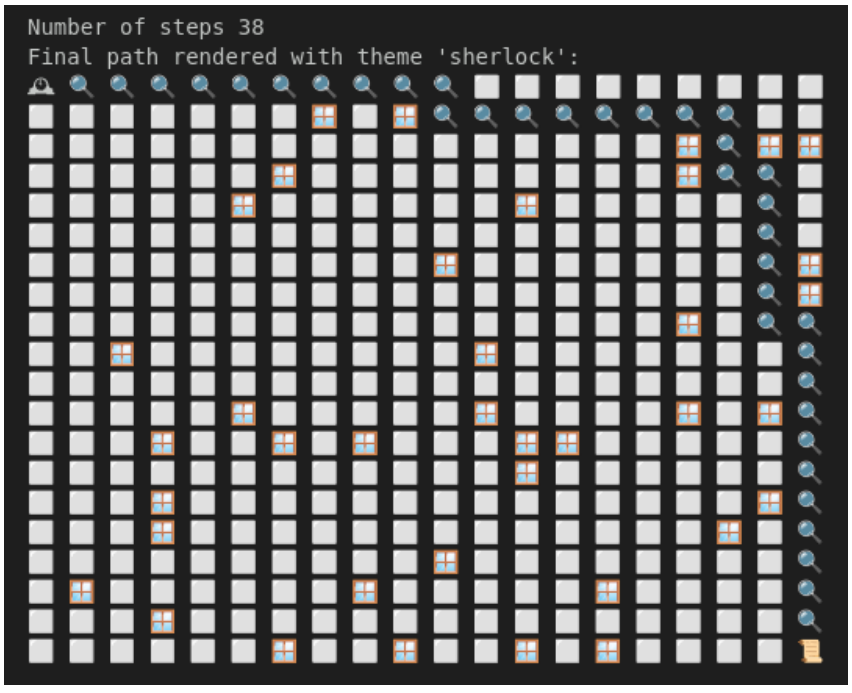

Figure 1: Actual Path

Following are the Q-Learning parameters used for training the agent:

```
# Q-learning parameters
###################################
num_actions = 4
gamma = 0.8             # between 0 - 1
alpha = 0.1             # between 0 - 1
epsilon = 0.9           # between 0 - 1
epsilon_decay = 0.99    # between 0.1 - 1
min_epsilon = 0.1
num_episodes = 10000
max_steps = 1000
###################################
```

Scenario 1: Traps and Boosts disabled

Reward Configuratin 1:

The pickle file for this scenario is `23651_disabled_1.pkl`.

The agent is trained with the following reward configuration:

```
# ========== REWARD CONFIG 1 ==========
REWARD_GOAL = 1000 # Reward for reaching goal .
REWARD_TRAP = -50 # Trap cell .
REWARD_OBSTACLE = -200 # Obstacle cell .
REWARD_REVISIT = -100 # Revisiting same cell .
REWARD_ENEMY = -100 # Getting caught by enemy .
REWARD_STEP = -2 # Per - step time penalty .
REWARD_BOOST = 50 # Boost cell .
# ###################################
```

Training:

```
New best at episode 0: 1000 steps and Reward -186072.00
Episode 0/10000 - Epsilon: 0.9900 - Total Steps: 1000 - Episode
   Reward: -186072.00 - Best Reward: -186072.00
New best at episode 2: 1000 steps and Reward -185532.00
New best at episode 3: 1000 steps and Reward -176542.00
New best at episode 4: 1000 steps and Reward -169544.00
New best at episode 6: 589 steps and Reward -98090.00
New best at episode 8: 479 steps and Reward -78518.00
New best at episode 10: 271 steps and Reward -41754.00
New best at episode 16: 290 steps and Reward -37510.00
New best at episode 24: 150 steps and Reward -22026.00
New best at episode 28: 133 steps and Reward -13046.00
New best at episode 41: 146 steps and Reward -12074.00
New best at episode 46: 114 steps and Reward -8514.00
New best at episode 47: 74 steps and Reward -5034.00
New best at episode 81: 60 steps and Reward -1514.00
New best at episode 112: 56 steps and Reward -1506.00
New best at episode 123: 57 steps and Reward -1210.00
New best at episode 124: 49 steps and Reward -994.00
New best at episode 139: 53 steps and Reward -498.00
New best at episode 161: 49 steps and Reward 6.00
New best at episode 163: 44 steps and Reward 18.00
New best at episode 164: 44 steps and Reward 118.00
New best at episode 165: 45 steps and Reward 614.00
New best at episode 197: 40 steps and Reward 922.00
New best at episode 228: 38 steps and Reward 926.00
Episode 1000/10000 - Epsilon: 0.1000 - Total Steps: 59 - Episode
   Reward: -3794.00 - Best Reward: 926.00
Episode 2000/10000 - Epsilon: 0.1000 - Total Steps: 46 - Episode
   Reward: -290.00 - Best Reward: 926.00
Episode 3000/10000 - Epsilon: 0.1000 - Total Steps: 126 - Episode
   Reward: -15846.00 - Best Reward: 926.00
Episode 4000/10000 - Epsilon: 0.1000 - Total Steps: 173 - Episode
   Reward: -35834.00 - Best Reward: 926.00
Episode 5000/10000 - Epsilon: 0.1000 - Total Steps: 211 - Episode
   Reward: -37806.00 - Best Reward: 926.00
Episode 6000/10000 - Epsilon: 0.1000 - Total Steps: 75 - Episode
   Reward: -7606.00 - Best Reward: 926.00
Episode 7000/10000 - Epsilon: 0.1000 - Total Steps: 104 - Episode
   Reward: -15506.00 - Best Reward: 926.00
Episode 8000/10000 - Epsilon: 0.1000 - Total Steps: 80 - Episode
   Reward: -9898.00 - Best Reward: 926.00
Episode 9000/10000 - Epsilon: 0.1000 - Total Steps: 46 - Episode
   Reward: -286.00 - Best Reward: 926.00

Training completed. Total episodes: 9999
Hence, Number of Steps taken by the agent: 38
```

Path Learnt:

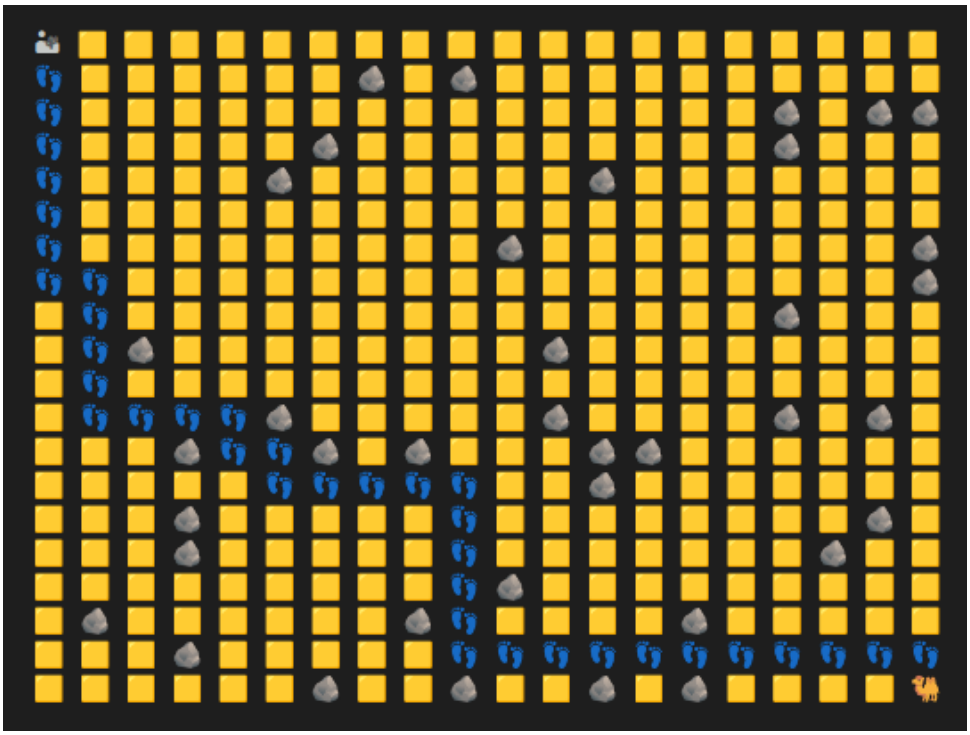

Figure 2: Path Learnt by the agent: Config 1 Scenario 1

```
Oasis reached! You survived the sands!
        Count  Reward
Goal        1  1000.0
Trap        0     0.0
Boost       0     0.0
Obstacle    0     0.0
Step       38   -76.0
Revisit     0     0.0
Total           924.0

Total Allowed Ateps: 1000
```

Q-Value Updates

- actions = [(-1, 0), (1, 0), (0, -1), (0, 1)] # Up, Down, Left, Right
- gamma = 0.8 = γ
- alpha = 0.1 = α
- REWARD_STEP = -2 = r
- Choosing action on basis of best Q-value (Exploitation)

Q-Learning update rule:

```
Q_table[state][action] = Q_table[state][action] + alpha * (reward +
    gamma * max(Q_table[new_state]) - Q_table[state][action])
```

Step 1:

```
starting_state = (0, 0)
Q_table[0][0] = [-442.93245878 -388.35970988 -444.08170308
   -420.81603719]
max(Q_table[0][0]) = -388.35970988
action = (1, 0) # Down
new_state = (1, 0)
Q_table[1][0] = [-483.7452502  -389.40455482 -536.48693291
   -430.73132237]
max(Q_table[1][0]) = -389.40455482
#Update Q_table[0][0][1]
Q_new = -338.35970988 + 0.1 * (-2 + 0.8 * (-389.40455482) -
   (-388.35970988))
Q_new = -380.87610328
so, Q_table[0][0][1] = -380.87610328
Q_table[0][0] = [-442.93245878 -380.87610328 -444.08170308
   -420.81603719]
```

Step 2:

```
current_state = (1, 0)
Q_table[1][0]= [-483.7452502  -389.40455482 -536.48693291
   -430.73132237]
max(Q_table[1][0]) = -389.40455482
action = (1, 0) # Down
new_state = (2, 0)
Q_table[2][0] = [-436.55832973 -383.91626555 -427.891149
   -407.89213864]
max(Q_table[2][0]) = -383.91626555
#Update Q_table[1][0][1]
Q_new = -389.40455482 + 0.1 * (-2 + 0.8 * (-383.91626555) -
   (-389.40455482))
Q_new = -381.37740058
so, Q_table[1][0][1] = -381.37740058
Q_table[1][0] = [-483.7452502  -381.37740058 -536.48693291
   -430.73132237]
#update current_state
current_state = new_state
```

Step 3:

```
current_state = (2, 0)
Q_table[2][0] = [-436.55832973 -383.91626555 -427.891149
    -407.89213864]
max(Q_table[2][0]) = -383.91626555
action = (1, 0) # Down
new_state = (3, 0)
Q_table[3][0] = [-427.99453489 -373.6439321  -408.88406082
    -401.61353476]
max(Q_table[3][0]) = -373.6439321
#Update Q_table[2][0][1]
Q_new = -383.91626555 + 0.1 * (-2 + 0.8 * (-373.6439321) -
    (-383.91626555))
Q_new = -375.61615356
so, Q_table[2][0][1] = -375.61615356
Q_table[2][0] = [-436.55832973 -375.61615356 -427.891149
    -407.89213864]
#update current_state
current_state = new_state
```

Step 4:

```
current_state = (3, 0)
Q_table[3][0] = [-427.99453489 -373.6439321  -408.88406082
    -401.61353476]
max(Q_table[3][0]) = -373.6439321
action = (1, 0) # Down
new_state = (4, 0)
Q_table[4][0] = [-458.42953297 -345.5577566  -394.53101162
    -421.81610624]
max(Q_table[4][0]) = -345.5577566
#Update Q_table[3][0][1]
Q_new = -373.6439321 + 0.1 * (-2 + 0.8 * (-345.5577566) -
    (-373.6439321))
Q_new = -364.12415942
so, Q_table[3][0][1] = -364.12415942
Q_table[3][0] = [-427.99453489 -364.12415942 -408.88406082
    -401.61353476]
#update current_state
current_state = new_state
```

Step 5:

```
current_state = (4, 0)
Q_table[4][0] = [-458.42953297 -345.5577566  -394.53101162
    -421.81610624]
max(Q_table[4][0]) = -345.5577566
action = (1, 0) # Down
new_state = (5, 0)
Q_table[5][0] = [-374.62766851 -332.60988983 -405.16088045
    -346.56963618]
max(Q_table[5][0]) = -332.60988983
#Update Q_table[4][0][1]
Q_new = -345.5577566 + 0.1 * (-2 + 0.8 * (-332.60988983) -
    (-345.5577566))
Q_new = -337.81077213
so, Q_table[4][0][1] = -337.81077213
Q_table[4][0] = [-458.42953297 -337.81077213 -394.53101162
    -421.81610624]
#update current_state
current_state = new_state
```

Therefore, the Q-table is updated as follows:

Q-Table

Q_table[0][0] = [-442.93245878 -380.87610328 -444.08170308 -420.81603719]
Q_table[1][0] = [-483.7452502 -381.37740058 -536.48693291 -430.73132237]
Q_table[2][0] = [-436.55832973 -375.61615356 -427.891149 -407.89213864]
Q_table[3][0] = [-427.99453489 -364.12415942 -408.88406082 -401.61353476]
Q_table[4][0] = [-458.42953297 -337.81077213 -394.53101162 -421.81610624]
Q_table[5][0] = [-374.62766851 -332.60988983 -405.16088045 -346.56963618]

Reward Configuratin 2:

The pickle file for this scenario is `23651_disabled_2.pkl`.

The agent is trained with the following reward configuration:

```
# ========== REWARDS ==========
REWARD_GOAL     = 2000  # Reward for reaching goal.
REWARD_TRAP     = -25   # Trap cell.
REWARD_OBSTACLE = -50 # Obstacle cell.
REWARD_REVISIT  = -250 # Revisiting same cell.
REWARD_ENEMY    = -50 # Getting caught by enemy.
REWARD_STEP     = -5   # Per-step time penalty.
REWARD_BOOST    =  50    # Boost cell.
#####################################
```

Training:

```
New best at episode 0: 1000 steps and Reward -391380.00
Episode 0/10000 - Epsilon: 0.9900 - Total Steps: 1000 - Episode
   Reward: -391380.00 - Best Reward: -391380.00
New best at episode 1: 1000 steps and Reward -369525.00
New best at episode 4: 711 steps and Reward -256755.00
New best at episode 9: 607 steps and Reward -216275.00
New best at episode 15: 502 steps and Reward -172825.00
New best at episode 17: 471 steps and Reward -139575.00
New best at episode 18: 269 steps and Reward -77125.00
New best at episode 27: 118 steps and Reward -23535.00
New best at episode 58: 98 steps and Reward -17525.00
New best at episode 72: 85 steps and Reward -13395.00
New best at episode 91: 90 steps and Reward -11575.00
New best at episode 113: 62 steps and Reward -6485.00
New best at episode 128: 51 steps and Reward -2725.00
New best at episode 139: 47 steps and Reward -1275.00
New best at episode 149: 44 steps and Reward 1285.00
New best at episode 289: 38 steps and Reward 1815.00
Episode 1000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -302525.00 - Best Reward: 1815.00
Episode 2000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -308240.00 - Best Reward: 1815.00
Episode 3000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -305690.00 - Best Reward: 1815.00
Episode 4000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -307100.00 - Best Reward: 1815.00
Episode 5000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -309830.00 - Best Reward: 1815.00
Episode 6000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -307755.00 - Best Reward: 1815.00
Episode 7000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -307870.00 - Best Reward: 1815.00
Episode 8000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -308780.00 - Best Reward: 1815.00
Episode 9000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -308805.00 - Best Reward: 1815.00

Training completed. Total episodes: 9999
Hence, Number of Steps taken by the agent: 38
```

Path Learnt:

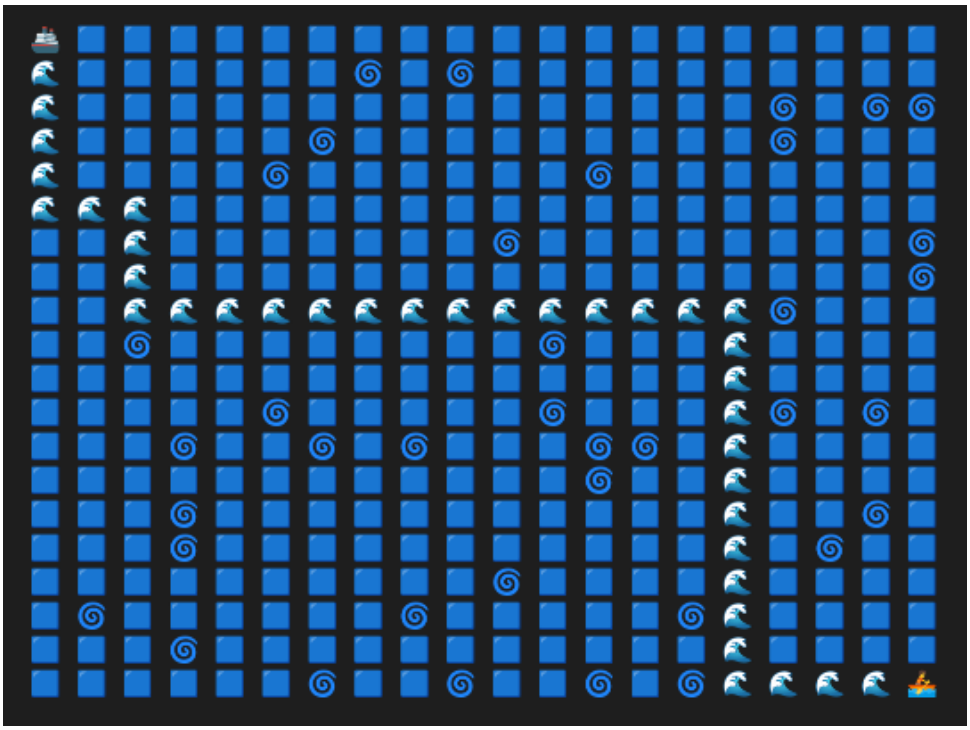

Figure 3: Path Learnt by the agent: Config 2 Scenario 1

```
Aye aye! You discovered the hidden island!
         Count  Reward
Goal         1  2000.0
Trap         0     0.0
Boost        0     0.0
Obstacle     0     0.0
Step        38  -190.0
Revisit      0     0.0
Total            1810.0

Total Allowed Ateps: 1000
```

Q-Value Updates

- actions = [(-1, 0), (1, 0), (0, -1), (0, 1)] # Up, Down, Left, Right
- gamma = 0.8 = γ
- alpha = 0.1 = α
- REWARD_STEP = -5 = r
- Choosing action on basis of best Q-value (Exploitation)

Q-Learning update rule:

```
Q_table[state][action] = Q_table[state][action] + alpha * (reward +
    gamma * max(Q_table[new_state]) - Q_table[state][action])
```

Step 1:

```
starting_state = (0, 0)
Q_table[0][0] = [-1162.59607199  -999.16919911 -1116.68895745
    -1112.47523598]
max(Q_table[0][0]) = -999.16919911
action = (1, 0) # Down
new_state = (1, 0)
Q_table[1][0] = [-1232.84628853 -1001.68925836 -1249.8218805
    -1062.67430421]
max(Q_table[1][0]) = -1001.68925836
#Update Q_table[0][0][1]
Q_new = -999.16919911 + 0.1 * (-5 + 0.8 * (-1001.68925836) -
    (-999.16919911))
Q_new = -979.8874198678
so, Q_table[0][0][1] = -979.8874198678
Q_table[0][0] = [-1162.59607199  -979.88741987 -1116.68895745
    -1112.47523598]
#update current_state
current_state = new_state
```

Step 2:

```
current_state = (1, 0)
Q_table[1][0]= [-1232.84628853 -1001.68925836 -1249.8218805
    -1062.67430421]
max(Q_table[1][0]) = -1001.68925836
action = (1, 0) # Down
new_state = (2, 0)
Q_table[2][0] = [-1175.10217682  -940.68156692 -1109.11904392
    -1006.50525216]
max(Q_table[2][0]) = -940.68156692
#Update Q_table[1][0][1]
Q_new = -1001.68925836 + 0.1 * (-5 + 0.8 * (-940.68156692) -
    (-1001.68925836))
Q_new = -977.2748578776
so, Q_table[1][0][1] = -977.2748578776
Q_table[1][0] = [-1232.84628853  -977.27485788 -1249.8218805
    -1062.67430421]
#update current_state
current_state = new_state
```

Step 3:

```
current_state = (2, 0)
Q_table[2][0] = [-1175.10217682  -940.68156692 -1109.11904392
    -1006.50525216]
max(Q_table[2][0]) = -940.68156692
action = (1, 0) # Down
new_state = (3, 0)
Q_table[3][0] = [-1064.40972236  -877.08092426 -1072.99732908
    -965.94248017]
max(Q_table[3][0]) = -877.08092426
#Update Q_table[2][0][1]
Q_new = -940.68156692 + 0.1 * (-5 + 0.8 * (-877.08092426) -
    (-940.68156692))
Q_new = -917.2798841688
so, Q_table[2][0][1] = -917.2798841688
Q_table[2][0] = [-1175.10217682  -917.27988417 -1109.11904392
    -1006.50525216]
#update current_state
current_state = new_state
```

Step 4:

```
current_state = (3, 0)
Q_table[3][0] = [-1064.40972236  -877.08092426 -1072.99732908
    -965.94248017]
max(Q_table[3][0]) = -877.08092426
action = (1, 0) # Down
new_state = (4, 0)
Q_table[4][0] = [-1008.65380405  -887.64209517 -1017.13123774
    -971.33905383]
max(Q_table[4][0]) = -887.64209517
#Update Q_table[3][0][1]
Q_new = -877.08092426 + 0.1 * (-5 + 0.8 * (-887.64209517) -
    (-877.08092426))
Q_new = -860.8841994476
so, Q_table[3][0][1] = -860.8841994476
Q_table[3][0] = [-1064.40972236  -860.88419945 -1072.99732908
    -965.94248017]
#update current_state
current_state = new_state
```

Step 5:

```
current_state = (4, 0)
Q_table[4][0] = [-1008.65380405  -887.64209517 -1017.13123774
    -971.33905383]
max(Q_table[4][0]) = -887.64209517
action = (1, 0) # Down
new_state = (5, 0)
Q_table[5][0] = [ -921.09960632  -846.9238057  -1016.18694669
    -801.58858701]
max(Q_table[5][0]) = -801.58858701
#Update Q_table[4][0][1]
Q_new = -887.64209517 + 0.1 * (-5 + 0.8 * (-801.58858701) -
    (-887.64209517))
Q_new = -863.5049726138
so, Q_table[4][0][1] = -863.5049726138
Q_table[4][0] = [-1008.65380405  -863.50497261 -1017.13123774
    -971.33905383]
#update current_state
current_state = new_state
```

Therefore, the Q-table is updated as follows:

Q-Table

Q_table[0][0] = [-1162.59607199 -979.88741987 -1116.68895745 -1112.47523598]
Q_table[1][0] = [-1232.84628853 -977.27485788 -1249.8218805 -1062.67430421]
Q_table[2][0] = [-1175.10217682 -917.27988417 -1109.11904392 -1006.50525216]
Q_table[3][0] = [-1064.40972236 -860.88419945 -1072.99732908 -965.94248017]
Q_table[4][0] = [-1008.65380405 -863.50497261 -1017.13123774 -971.33905383]
Q_table[5][0] = [-921.09960632 -846.9238057 -1016.18694669 -801.58858701]

Scenario 2: Traps and Boosts enabled

Reward Configuratin 1:

The pickle file for this scenario is `23651_enabled_1.pkl`.

The agent is trained with the following reward configuration:

```
# ========== REWARDS ==========
REWARD_GOAL = 2000 # Reward for reaching goal .
REWARD_TRAP = -25 # Trap cell .
REWARD_OBSTACLE = -2000 # Obstacle cell .
REWARD_REVISIT = -500 # Revisiting same cell .
REWARD_ENEMY = -50 # Getting caught by enemy .
REWARD_STEP = -5 # Per - step time penalty .
REWARD_BOOST = 100 # Boost cell .
#####################################
```

Training:

```
New best at episode 0: 1000 steps and Reward -1081805.00
Episode 0/10000 - Epsilon: 0.9900 - Total Steps: 1000 - Episode
   Reward: -1081805.00 - Best Reward: -1081805.00
New best at episode 2: 1000 steps and Reward -1040195.00
New best at episode 3: 1000 steps and Reward -940235.00
New best at episode 12: 1000 steps and Reward -925750.00
New best at episode 13: 413 steps and Reward -364600.00
New best at episode 14: 344 steps and Reward -276370.00
New best at episode 17: 330 steps and Reward -271115.00
New best at episode 21: 165 steps and Reward -110880.00
New best at episode 29: 105 steps and Reward -48240.00
New best at episode 75: 93 steps and Reward -32645.00
New best at episode 77: 86 steps and Reward -31725.00
New best at episode 110: 79 steps and Reward -27870.00
New best at episode 115: 67 steps and Reward -11230.00
New best at episode 123: 50 steps and Reward -9800.00
New best at episode 134: 48 steps and Reward -5955.00
New best at episode 150: 44 steps and Reward -1455.00
New best at episode 181: 38 steps and Reward 2220.00
New best at episode 282: 44 steps and Reward 2420.00
New best at episode 287: 46 steps and Reward 2430.00
New best at episode 982: 62 steps and Reward 2690.00
Episode 1000/10000 - Epsilon: 0.1000 - Total Steps: 156 - Episode
   Reward: -249745.00 - Best Reward: 2690.00
Episode 2000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2219510.00 - Best Reward: 2690.00
Episode 3000/10000 - Epsilon: 0.1000 - Total Steps: 158 - Episode
   Reward: -246635.00 - Best Reward: 2690.00
Episode 4000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2239685.00 - Best Reward: 2690.00
Episode 5000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2280765.00 - Best Reward: 2690.00
Episode 6000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2262605.00 - Best Reward: 2690.00
Episode 7000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2233925.00 - Best Reward: 2690.00
Episode 8000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2232390.00 - Best Reward: 2690.00
Episode 9000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -2227005.00 - Best Reward: 2690.00

Training completed. Total episodes: 9999
Hence, Number of Steps taken by the agent: 62
```

Path Learnt:

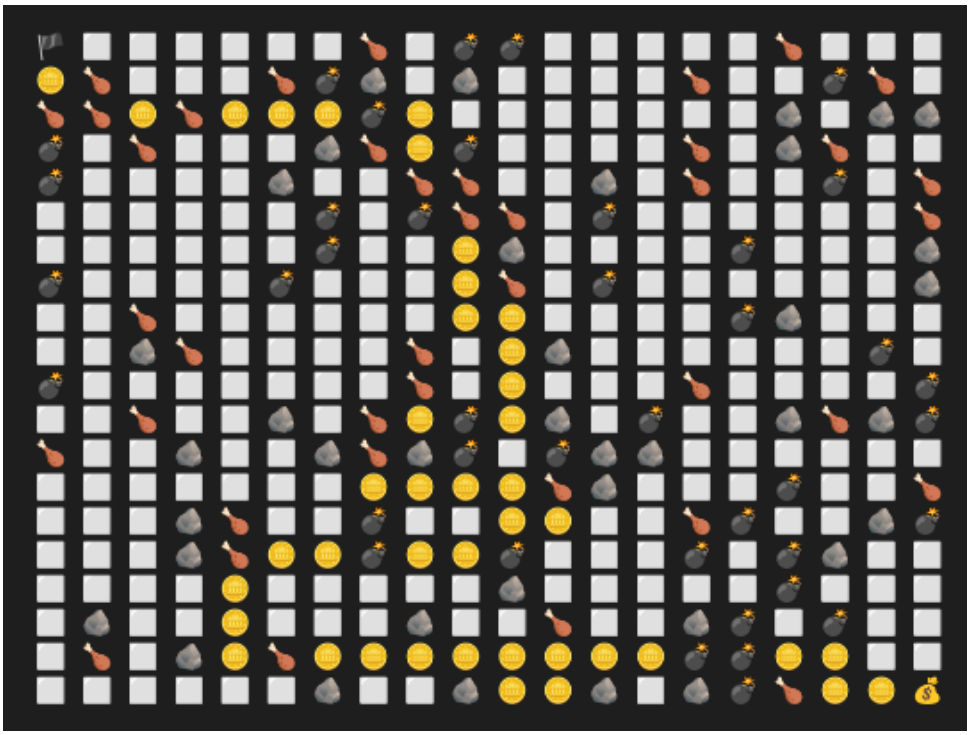

Figure 4: Path Learnt by the agent: Config 2 Scenario 1

```
      Treasure secured! You sailed to fortune!
          Count  Reward
Goal          1  2000.0
Trap          8  -200.0
Boost        11  1100.0
Obstacle      0     0.0
Step         62  -310.0
Revisit       0     0.0
Total            2590.0

Total Allowed Ateps: 1000
```

Reward Configuratin 2:

The pickle file for this scenario is `23651_enabled_2.pkl`.

The agent is trained with the following reward configuration:

```
# ========== REWARDS ==========
REWARD_GOAL = 1000 # Reward for reaching goal .
REWARD_TRAP = -50 # Trap cell .
REWARD_OBSTACLE = -1000 # Obstacle cell .
REWARD_REVISIT = -100 # Revisiting same cell .
REWARD_ENEMY = -100 # Getting caught by enemy .
REWARD_STEP = -2 # Per - step time penalty .
REWARD_BOOST = 50 # Boost cell .
#####################################
```

Training:

```
New best at episode 0: 1000 steps and Reward -314572.00
Episode 0/10000 - Epsilon: 0.9900 - Total Steps: 1000 - Episode
   Reward: -314572.00 - Best Reward: -314572.00
New best at episode 2: 1000 steps and Reward -286588.00
New best at episode 3: 1000 steps and Reward -276660.00
New best at episode 4: 1000 steps and Reward -262514.00
New best at episode 6: 616 steps and Reward -159178.00
New best at episode 11: 368 steps and Reward -85650.00
New best at episode 16: 250 steps and Reward -74242.00
New best at episode 17: 277 steps and Reward -62390.00
New best at episode 21: 273 steps and Reward -56314.00
New best at episode 23: 226 steps and Reward -47106.00
New best at episode 24: 219 steps and Reward -46590.00
New best at episode 25: 181 steps and Reward -35650.00
New best at episode 26: 138 steps and Reward -28958.00
New best at episode 36: 151 steps and Reward -27762.00
New best at episode 42: 136 steps and Reward -18922.00
New best at episode 56: 99 steps and Reward -15730.00
New best at episode 62: 101 steps and Reward -14754.00
New best at episode 68: 82 steps and Reward -6626.00
New best at episode 75: 69 steps and Reward -3858.00
New best at episode 83: 65 steps and Reward -3698.00
New best at episode 86: 50 steps and Reward -2974.00
New best at episode 131: 67 steps and Reward -2002.00
New best at episode 139: 63 steps and Reward -1542.00
New best at episode 150: 52 steps and Reward 714.00
New best at episode 183: 42 steps and Reward 934.00
New best at episode 297: 46 steps and Reward 1082.00
New best at episode 391: 44 steps and Reward 1138.00
Episode 1000/10000 - Epsilon: 0.1000 - Total Steps: 74 - Episode
   Reward: -2586.00 - Best Reward: 1138.00
New best at episode 1399: 46 steps and Reward 1178.00
New best at episode 1640: 46 steps and Reward 1230.00
Episode 2000/10000 - Epsilon: 0.1000 - Total Steps: 93 - Episode
   Reward: -9002.00 - Best Reward: 1230.00
Episode 3000/10000 - Epsilon: 0.1000 - Total Steps: 120 - Episode
   Reward: -60494.00 - Best Reward: 1230.00
Episode 4000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -966574.00 - Best Reward: 1230.00
Episode 5000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -937830.00 - Best Reward: 1230.00
Episode 6000/10000 - Epsilon: 0.1000 - Total Steps: 511 - Episode
   Reward: -410310.00 - Best Reward: 1230.00
Episode 7000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -958142.00 - Best Reward: 1230.00
Episode 8000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -952702.00 - Best Reward: 1230.00
Episode 9000/10000 - Epsilon: 0.1000 - Total Steps: 1000 - Episode
   Reward: -963834.00 - Best Reward: 1230.00

Training completed. Total episodes: 9999
Hence, Number of Steps taken by the agent: 46
```

Path Learnt:

Figure 5: Path Learnt by the agent: Config 2 Scenario 2

```
Upload complete! You hacked the system!
        Count  Reward
Goal        1  1000.0
Trap        2  -100.0
Boost       8   400.0
Obstacle    0     0.0
Step       50  -100.0
Revisit     0     0.0
Total          1200.0

Total Allowed Ateps: 1000
```

Observations and Conclusions

Consider Scenario 1: (Traps and Boosts disabled)

- Path Efficiency:
  In training, the agent took 38 steps to reach the goal in both configuration 1 and 2
  In testing, the agent again took 38 steps to reach the goal in both configuration 1 and 2

- Penality for revisiting cells:
  The agent took a more direct route to the goal in configuration 2 as it had a higher time step penalty.

- Boosts and Traps were avoided completely.
- Config 2 took longer to reach the best episode (289) than config 1 (228), likely due to stricter penalities in the learning environment.
- Path taken by the agent during testing in config 1, seemed to be closer to the obstacles, indicating exploration behaviour, this might be due to its relatively lenient penalties.
- While in configuration 2, the path taken was strictly away from the obstacles indicating that exploitation behaviour was prioritized. Due to its harsh revisit and step penalties, the agent to learned a more disciplined, exploitative strategy early on.
- In both configs, identical behaviour was observed in both testing and training, indicating that the learnt policies were consistent and optimal.

Consider Scenario 2: (Traps and Boosts enabled)

- Path Efficiency:
  In training, the agent took 62 steps to reach the goal in config 1 and 46 steps in config 2
  In testing, the agent took 62 steps to reach the goal in config 1 and 50 steps in config 2
- The agent achieved a higher reward in config 1 (2590) than in config 2 (1200), eventhough config 1 had higher revisiting, step and obstacle penalties. This is due to the larger goal and boost rewards in Configuration 1, which encouraged the agent to explore more reward-dense paths, even at the cost of higher risk.
- The agent in Configuration 1 stepped on 8 traps and collected 11 boosts, whereas in Configuration 2, it only encountered 2 traps and collected 8 boosts. This shows that Configuration 1 indulged into strategic risk-taking for maximized gains, while Configuration 2 tried to stay more towards caution.
- The reward structure encouraged exploration in Config 1, leading to a more diverse path with more traps and boosts. In contrast, Config 2's structure led to a more conservative approach, with fewer traps and boosts encountered.
- Config 1 took longer to reach the best episode (982) than config 2 (391), likely due to its more complex reward structure, which required more exploration and learning.

Comparision between scenario 1 and scenario 2

- The analysis across both the scenarios, clearly suggest that the agent's learning and decision-making are significantly influenced by the reward configuration.

- In scenario 1 (Traps and Boosts disabled), the agent focussed on minimizing penalities and taking a more direct route to the goal. This resulted in efficient and shorter paths, but with less exploration.

- In scenario 2 (Traps and Boosts enabled), additional elements of "risk and rewards" were introduced, leading to more complex learning pattern.

- The agent's behaviour shifted towards exploration, leading to longer paths as the agent tried to balance the risks of traps with the rewards of boosts.

- We saw that in scenario 2, if we have a higher boost reward, the agent is more likely to take risks and explore traps (configuration 1), leading to more longer but rewarding path. This demostrates the "strategic risk-taking" behaviour of the agent.

- In contrast, higher penalities and lower boost rewards (configuration 2) led to a more cautious approach, with the agent trying to interact with fewer traps and boosts, resulting in shorter paths but lower rewards.