**Auto-Grader points:**

Q1: 6/6

Q2: 1/1

Q3: 5/5

Q4: 5/5

Q5: 3/3

Q6: 1/1

Q7: 1/1

Q8: 3/3

Total: 25/25

**Explanations:**

Q1: This question provides an output with multiple boxes containing values and arrows, where Q-values are numbers in square quarters and the policies are arrows out from each square. Here the main iteration is run in the ValuesIterationAgent construction and it runs 100 times. If the current state is the terminal state then we append the value ‘0’ to the temp array. To compute the ‘q’ values, we keep adding the probability, which is multiplied by the inbuilt function, getReward(). Now, to compute the action from values, we compute the Q values and provide the state and action as the parameters. If temp is >= the values in the box, a = action.

Q2: Here, we only need to change the noise in one of the files. Noise refers to how often an agent ends up in an unintended successor state when they perform an action. If we change the noise to 0, our problem is solved.

Q3: In this question we need to change the values of discounts, noises and LivingRewards in the analysis file. The values that we choose should have the property that, if our agent followed its optimal path given, it would exhibit the given behaviour. If we do not achieve this then we need to return ‘Not Possible’.

Q4: For this question we need to write a Q-learning agent which does very little construction, but instead learns by trial and error method. In the getQvalue method, if the state and action is not present in the dictionary then we add it to it and initialize it with 0.0. Now, to compute values from q values, if the length of legal actions is empty, then we just need to return 0. Now, we need to record qvalues for each legal\_actions and return it.

Q5: This question is the continued part of the previous question. We have to edit the getAction function. First, we get all the legal actions using the inbuilt function self.getLegalActions(state). Further, we used a flipcoin() function to simulate a binary variable with the probability p of success, which returns True for p and False for 1-p. If true, we randomly choose from legal\_actions.

Q6: This question basically asks us to find an epsilon and LR for which it is highly likely that the optimal policy will be learned after 50 iterations. If it is not available, then we need to return ‘Not Possible’.

Q7: Here, PacMan will play the game in two phases, training and testing. We did not have to change any code in order to complete this question.

Q8: For this question, we need to complete the ApproximateQagent class in qlearningagent.py file. I used the following formula to update my weight vectors:   
 wi←wi+α⋅difference⋅fi(s,a)  
 difference=(r+γmaxa′Q(s′,a′))−Q(s,a)

**AUTOGRADER:**

Starting on 11-6 at 19:46:55

Question q1

===========

\*\*\* PASS: test\_cases\q1\1-tinygrid.test

\*\*\* PASS: test\_cases\q1\2-tinygrid-noisy.test

\*\*\* PASS: test\_cases\q1\3-bridge.test

\*\*\* PASS: test\_cases\q1\4-discountgrid.test

### Question q1: 6/6 ###

Question q2

===========

\*\*\* PASS: test\_cases\q2\1-bridge-grid.test

### Question q2: 1/1 ###

Question q3

===========

\*\*\* PASS: test\_cases\q3\1-question-3.1.test

\*\*\* PASS: test\_cases\q3\2-question-3.2.test

\*\*\* PASS: test\_cases\q3\3-question-3.3.test

\*\*\* PASS: test\_cases\q3\4-question-3.4.test

\*\*\* PASS: test\_cases\q3\5-question-3.5.test

### Question q3: 5/5 ###

Question q4

===========

\*\*\* PASS: test\_cases\q4\1-tinygrid.test

\*\*\* PASS: test\_cases\q4\2-tinygrid-noisy.test

\*\*\* PASS: test\_cases\q4\3-bridge.test

\*\*\* PASS: test\_cases\q4\4-discountgrid.test

### Question q4: 5/5 ###

Question q5

===========

\*\*\* PASS: test\_cases\q5\1-tinygrid.test

\*\*\* PASS: test\_cases\q5\2-tinygrid-noisy.test

\*\*\* PASS: test\_cases\q5\3-bridge.test

\*\*\* PASS: test\_cases\q5\4-discountgrid.test

### Question q5: 3/3 ###

Question q6

===========

\*\*\* PASS: test\_cases\q6\grade-agent.test

### Question q6: 1/1 ###

Question q7

===========

Beginning 2000 episodes of Training

Reinforcement Learning Status:

Completed 100 out of 2000 training episodes

Average Rewards over all training: -510.51

Average Rewards for last 100 episodes: -510.51

Episode took 1.65 seconds

Reinforcement Learning Status:

Completed 200 out of 2000 training episodes

Average Rewards over all training: -511.39

Average Rewards for last 100 episodes: -512.27

Episode took 2.22 seconds

Reinforcement Learning Status:

Completed 300 out of 2000 training episodes

Average Rewards over all training: -508.54

Average Rewards for last 100 episodes: -502.85

Episode took 2.62 seconds

Reinforcement Learning Status:

Completed 400 out of 2000 training episodes

Average Rewards over all training: -482.23

Average Rewards for last 100 episodes: -403.30

Episode took 3.11 seconds

Reinforcement Learning Status:

Completed 500 out of 2000 training episodes

Average Rewards over all training: -437.89

Average Rewards for last 100 episodes: -260.50

Episode took 3.05 seconds

Reinforcement Learning Status:

Completed 600 out of 2000 training episodes

Average Rewards over all training: -411.81

Average Rewards for last 100 episodes: -281.44

Episode took 3.14 seconds

Reinforcement Learning Status:

Completed 700 out of 2000 training episodes

Average Rewards over all training: -395.71

Average Rewards for last 100 episodes: -299.09

Episode took 2.92 seconds

Reinforcement Learning Status:

Completed 800 out of 2000 training episodes

Average Rewards over all training: -369.76

Average Rewards for last 100 episodes: -188.09

Episode took 3.08 seconds

Reinforcement Learning Status:

Completed 900 out of 2000 training episodes

Average Rewards over all training: -338.37

Average Rewards for last 100 episodes: -87.32

Episode took 3.05 seconds

Reinforcement Learning Status:

Completed 1000 out of 2000 training episodes

Average Rewards over all training: -304.22

Average Rewards for last 100 episodes: 3.15

Episode took 3.28 seconds

Reinforcement Learning Status:

Completed 1100 out of 2000 training episodes

Average Rewards over all training: -278.08

Average Rewards for last 100 episodes: -16.61

Episode took 3.17 seconds

Reinforcement Learning Status:

Completed 1200 out of 2000 training episodes

Average Rewards over all training: -236.04

Average Rewards for last 100 episodes: 226.29

Episode took 3.13 seconds

Reinforcement Learning Status:

Completed 1300 out of 2000 training episodes

Average Rewards over all training: -199.60

Average Rewards for last 100 episodes: 237.72

Episode took 2.96 seconds

Reinforcement Learning Status:

Completed 1400 out of 2000 training episodes

Average Rewards over all training: -169.89

Average Rewards for last 100 episodes: 216.33

Episode took 3.11 seconds

Reinforcement Learning Status:

Completed 1500 out of 2000 training episodes

Average Rewards over all training: -143.44

Average Rewards for last 100 episodes: 226.89

Episode took 3.03 seconds

Reinforcement Learning Status:

Completed 1600 out of 2000 training episodes

Average Rewards over all training: -122.18

Average Rewards for last 100 episodes: 196.79

Episode took 3.05 seconds

Reinforcement Learning Status:

Completed 1700 out of 2000 training episodes

Average Rewards over all training: -101.09

Average Rewards for last 100 episodes: 236.32

Episode took 3.16 seconds

Reinforcement Learning Status:

Completed 1800 out of 2000 training episodes

Average Rewards over all training: -81.76

Average Rewards for last 100 episodes: 246.83

Episode took 3.16 seconds

Reinforcement Learning Status:

Completed 1900 out of 2000 training episodes

Average Rewards over all training: -68.70

Average Rewards for last 100 episodes: 166.43

Episode took 3.00 seconds

Reinforcement Learning Status:

Completed 2000 out of 2000 training episodes

Average Rewards over all training: -54.44

Average Rewards for last 100 episodes: 216.40

Episode took 3.11 seconds

Training Done (turning off epsilon and alpha)

---------------------------------------------

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 499

Pacman emerges victorious! Score: 495

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 503

Pacman emerges victorious! Score: 499

Reinforcement Learning Status:

Completed 100 test episodes

Average Rewards over testing: 500.00

Average Rewards for last 100 episodes: 500.00

Episode took 3.14 seconds

Average Score: 500.0

Scores: 503.0, 503.0, 503.0, 499.0, 499.0, 503.0, 503.0, 503.0, 495.0, 503.0, 495.0, 503.0, 499.0, 499.0, 503.0, 499.0, 503.0, 499.0, 495.0, 503.0, 503.0, 503.0, 503.0, 503.0, 503.0, 503.0, 503.0, 503.0, 499.0, 499.0, 503.0, 495.0, 503.0, 495.0, 499.0, 503.0, 503.0, 503.0, 495.0, 503.0, 499.0, 503.0, 503.0, 503.0, 503.0, 499.0, 503.0, 503.0, 495.0, 495.0, 503.0, 503.0, 499.0, 503.0, 499.0, 499.0, 495.0, 503.0, 499.0, 499.0, 503.0, 499.0, 499.0, 499.0, 495.0, 495.0, 495.0, 499.0, 495.0, 495.0, 503.0, 503.0, 499.0, 503.0, 495.0, 499.0, 503.0, 495.0, 499.0, 495.0, 503.0, 499.0, 495.0, 503.0, 495.0, 503.0, 495.0, 503.0, 503.0, 495.0, 503.0, 503.0, 499.0, 495.0, 495.0, 499.0, 495.0, 503.0, 503.0, 499.0

Win Rate: 100/100 (1.00)

Record: Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win

\*\*\* PASS: test\_cases\q7\grade-agent.test (1 of 1 points)

\*\*\* Grading agent using command: python pacman.py -p PacmanQAgent -x 2000 -n 2100 -l smallGrid -q -f --fixRandomSeed

\*\*\* 100 wins (1 of 1 points)

\*\*\* Grading scheme:

\*\*\* < 70: 0 points

\*\*\* >= 70: 1 points

### Question q7: 1/1 ###

Question q8

===========

\*\*\* PASS: test\_cases\q8\1-tinygrid.test

\*\*\* PASS: test\_cases\q8\2-tinygrid-noisy.test

\*\*\* PASS: test\_cases\q8\3-bridge.test

\*\*\* PASS: test\_cases\q8\4-discountgrid.test

\*\*\* PASS: test\_cases\q8\5-coord-extractor.test

### Question q8: 3/3 ###

Finished at 19:48:02

Provisional grades

==================

Question q1: 6/6

Question q2: 1/1

Question q3: 5/5

Question q4: 5/5

Question q5: 3/3

Question q6: 1/1

Question q7: 1/1

Question q8: 3/3

------------------

Total: 25/25

**GITLOCK:**