

F29AI – CW2

2023 – 2024

Q - Learning [AJAY MENON, H00418802]

Defensive (W: 43, D: 7)

```
11 import static org.junit.Assert.assertEquals;
12
13 public class TestQLearning {
14     @Test
15     public void testDefensive() throws IllegalMoveException {
16         System.out.println("Against Defensive Agent");
17         int[] results = TestPolicyIterationAgent.playAgainstEachOther(new QLearningAgent(), new DefensiveAgent(), 50);
18         System.out.println("Wins: " + results[0] + " Losses: " + results[1] + " Draws: " + results[2]);
19         assertEquals(0, results[1]);
20     }
21 }
22
23 // @Test
24 // public void testAggressive() throws IllegalMoveException {
25 //
26 // }
```

Finished after 1.939 seconds

Runs: 1/1 Errors: 0 Failures: 0

TestQLearning [Runner: JUnit 4] (1.939 s)

testDefensive (1.939 s)

Playing move: X(0,0)

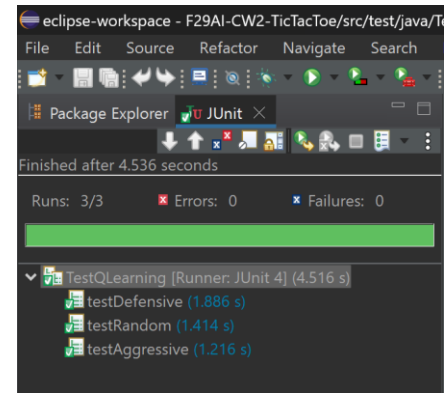
Playing defensive move

Playing move: O(0,1)

Playing move: X(1,1)

X won!

Wins: 43 Losses: 0 Draws: 7



Aggressive (W: 49, D: 1)

```
24 public void testAggressive() throws IllegalMoveException {
25     System.out.println("Against Aggressive Agent");
26     int[] results = TestPolicyIterationAgent.playAgainstEachOther(new QLearningAgent(), new AggressiveAgent(), 50);
27     System.out.println("Wins: " + results[0] + " Losses: " + results[1] + " Draws: " + results[2]);
28     assertEquals(0, results[1]);
29 }
30
31 // @Test
32 // public void testRandom() throws IllegalMoveException {
33 //     System.out.println("Against Random Agent");
34 //     int[] results = TestPolicyIterationAgent.playAgainstEachOther(new QLearningAgent(), new RandomAgent(), 50);
35 //     System.out.println("Wins: " + results[0] + " Losses: " + results[1] + " Draws: " + results[2]);
36 //     assertEquals(0, results[1]);
37 // }
```

Finished after 1.829 seconds

Runs: 1/1 Errors: 0 Failures: 0

TestQLearning [Runner: JUnit 4] (1.819 s)

testAggressive (1.819 s)

Playing move: X(0,1)

Playing move: O(0,0)

Playing move: X(2,1)

X won!

Wins: 49 Losses: 0 Draws: 1

Random (W: 48, D: 2)

```
34 public void testRandom() throws IllegalMoveException {
35     System.out.println("Against Random Agent");
36     int[] results = TestPolicyIterationAgent.playAgainstEachOther(new QLearningAgent(), new RandomAgent(), 50);
37     System.out.println("Wins: " + results[0] + " Losses: " + results[1] + " Draws: " + results[2]);
38     assertEquals(0, results[1]);
39 }
40
41 // @Test
42 // public void testDefensive() throws IllegalMoveException {
43 //     System.out.println("Against Defensive Agent");
44 //     int[] results = TestPolicyIterationAgent.playAgainstEachOther(new QLearningAgent(), new DefensiveAgent(), 50);
45 //     System.out.println("Wins: " + results[0] + " Losses: " + results[1] + " Draws: " + results[2]);
46 //     assertEquals(0, results[1]);
47 // }
```

Finished after 1.869 seconds

Runs: 1/1 Errors: 0 Failures: 0

TestQLearning [Runner: JUnit 4] (1.872 s)

testRandom (1.872 s)

Playing move: X(0,2)

Playing move: O(0,1)

Playing move: X(1,2)

X won!

Wins: 48 Losses: 0 Draws: 2