########################
Problem Set 4 | Reinforcement
Group Submission
Members:
Yasser Parambathkandy - (G01294910)
Indranil Pal - (G01235186)
Date: 10/28/2022

---

1. Run Value Iteration by hand for 10 iterations on the given environment. Make sure you note down the utility at each iteration separately to keep things organized.

Iteration	**U Values**			
	S1	S2	S3	S4
1	0.01	0.01	0.01	1.05
2	0.381	-0.035	0.381	1.473
3	0.5665	0.1297	0.5665	1.6819
4	0.66757	0.22141	0.66757	1.78518
5	0.718521	0.271477	0.718521	1.8367095
6	0.718521	0.271477	0.758521	1.8367095
7	0.74385	0.31265	0.78585	1.8641
8	0.75845	0.32725	0.80055	1.8781
9	0.7655	0.3347	0.8076	1.88515
10	0.76905	0.33825	0.81115	1.88865

2. Write some code to automate the utility updates for this environment. Make sure your code agrees with the work you did by hand in the previous question.
Output of code below. Attached is the code.

```
-------start iteration: 1
U values : [0.010000000000000005, 0.010000000000000009, 0.010000000000000009, 1.0499999999999998]
End of iteration  1
-------start iteration: 2
U values : [0.38099999999999995, 0, 0.38099999999999995, 1.4729999999999999]
End of iteration  2
-------start iteration: 3
U values : [0.5682499999999999, 0.13144999999999998, 0.5682499999999999, 1.6819]
End of iteration  3
-------start iteration: 4
U values : [0.667745, 0.22228499999999998, 0.667745, 1.7852674999999998]
End of iteration  4
-------start iteration: 5
U values : [0.7186085, 0.2715995, 0.7186085, 1.8367576250000002]
End of iteration  5
-------start iteration: 6
U values : [0.7442134500000002, 0.29695380000000005, 0.7442134500000002, 1.8624713562500002]
End of iteration  6
-------start iteration: 7
U values : [0.7570469050000002, 0.3097437425000001, 0.7570469050000002, 1.8753227828125003]
End of iteration  7
-------start iteration: 8
U values : [0.7634686455000002, 0.31615829437500015, 0.7634686455000002, 1.881747597515625]
End of iteration  8
```

-------start iteration: 9
U values : [0.766680386, 0.3193688051937501, 0.766680386, 1.8849598511570314]
End of iteration 9
-------start iteration: 10
U values : [0.7682864000225, 0.32097461395968757, 0.7682864000225, 1.8865659523206642]
End of iteration 10

Process finished with exit code 0

3. How long does it take for the utility to converge? Pick a threshold for small differences and see how long it takes until none of the utility values for any of the states is changing more than this threshold.

Threshold	Stopped at Iteration
0.1	5
0.01	8
0.001	11
0.0001	15

Code output below, attached is the code :

-------start iteration: 1
U values : [0.010000000000000005, 0.010000000000000009, 0.010000000000000009, 1.0499999999999998]
End of iteration 1
-------start iteration: 2
U values : [0.38099999999999995, 0, 0.38099999999999995, 1.4729999999999999]
End of iteration 2
-------start iteration: 3
U values : [0.5682499999999999, 0.13144999999999998, 0.5682499999999999, 1.6819]
End of iteration 3
-------start iteration: 4
U values : [0.667745, 0.22228499999999998, 0.667745, 1.7852674999999998]
End of iteration 4
-------start iteration: 5
U values : [0.7186085, 0.2715995, 0.7186085, 1.8367576250000002]
End of iteration 5
-------start iteration: 6
U values : [0.7442134500000002, 0.29695380000000005, 0.7442134500000002, 1.8624713562500002]
End of iteration 6
-------start iteration: 7
U values : [0.7570469050000002, 0.3097437425000001, 0.7570469050000002, 1.8753227828125003]
End of iteration 7
-------start iteration: 8
U values : [0.7634686455000002, 0.31615829437500015, 0.7634686455000002, 1.881747597515625]
End of iteration 8
-------start iteration: 9
U values : [0.766680386, 0.3193688051937501, 0.766680386, 1.8849598511570314]
End of iteration 9
-------start iteration: 10
U values : [0.7682864000225, 0.32097461395968757, 0.7682864000225, 1.8865659523206642]
End of iteration 10
-------start iteration: 11
U values : [0.7690894316273751, 0.32177761070810945, 0.7690894316273751, 1.887368998545424]
End of iteration 11
-------start iteration: 12
U values : [0.769490951534944, 0.3221791247677243, 0.769490951534944, 1.8877705209268096]
End of iteration 12
-------start iteration: 13
U values : [0.7696917121858573, 0.32237988442911103, 0.7696917121858573, 1.8879712819938115]
End of iteration 13
-------start iteration: 14
U values : [0.7697920926282731, 0.3224802647050914, 0.7697920926282731, 1.888071662506508]
End of iteration 14
-------start iteration: 15

U values : [0.7698422828692715, 0.3225304549179775, 0.7698422828692715, 1.8881218527593422]
Threashold : 0.0001 , Stopped at iteration 15

4. Run Q-Learning by hand on the same environment using a learning rate of _ = 0:5, an initial Q-table of all zeros, and the following experience traces (given as (s; a; s0; r) tuples):

(a) (s2;Up; s1;□0:04)

Q(s,a)←(1-α)Q(s,a) + α(r+γmaxa' Q(s',a'))				
α	0.5			
γ	0.5			

Initial Q

	UP	DOWN	LEFT	RIGHT
s1	0	0	0	0
s2	0	0	0	0
s3	0	0	0	0
s4	0	0	0	0

Actions

(s; a; s0; r)	Source	Action	Destination	Reward	Q	Formula	Value
(s2;Up; s1;□0:04)	S2	UP	S1	-0.04	(s2, UP)	(1-0.5)Q(s2,UP) + 0.5(0.04+0.5 * maxa' Q(s',a'))	-0.02

Q after first step

	UP	DOWN	LEFT	RIGHT
s1	0	0	0	0
s2	-0.02	0	0	0
s3	0	0	0	0
s4	0	0	0	0

(b) (s1;Right; s4; 1:0)

(s; a; s0; r)	Source	Action	Destination	Reward	Q	Formula	Value
(s1;Right; s4; 1:0)	S1	RIGHT	S4	1		(1-0.5)Q(s1,RIGHT) + 0.5(1+0.5 * maxa' Q(s',a'))	0.5

Q after this step

	UP	DOWN	LEFT	RIGHT
s1	0	0	0	0.5
s2	-0.02	0	0	0
s3	0	0	0	0
s4	0	0	0	0

(c) (s2;Right; s3;□0:04)

(s; a; s0; r)	Source	Action	Destination	Reward	Q	Formula	Value
(s2;Right; s3;□0:04)	S2	RIGHT	S3	-0.04		(1-0.5)Q(s2,RIGHT) + 0.5(-0.04+0.5 * maxa' Q(s',a'))	-0.02

Q after this step

	UP	DOWN	LEFT	RIGHT

s1	0	0	0	0.5
s2	-0.02	0	0	-0.02
s3	0	0	0	0
s4	0	0	0	0

(d) (s_3;Up; s_2;0:04)

(s; a; s0; r)	Source	Action	Destination	Reward	Q	Formula	Value
(s3;Up; s2;0:04)	S3	UP	S2	-0.04		(1-0.5)Q(s3,UP) + 0.5(-0.04+0.5 * maxa' Q(s',a'))	-0.02

Q after this step

	UP	DOWN	LEFT	RIGHT
s1	0	0	0	0.5
s2	-0.02	-0.02	0	-0.02
s3	-0.02	0	0	0
s4	0	0	0	0

(e) (s_2;Up; s_1;0:04)

(s; a; s0; r)	Source	Action	Destination	Reward	Q	Formula	Value
(s2;Up; s1;0:04)	S2	UP	S1	-0.04	(s2, UP)	(1-0.5)Q(s2,UP) + 0.5(-0.04+0.5 * maxa' Q(s',a'))	0.095

Q after this step

	UP	DOWN	LEFT	RIGHT
s1	0	0	0	0.5
s2	0.095	0	0	0
s3	0	0	0	0
s4	0	0	0	0

(f) (s_1;Right; s_4; 1:0)

(s; a; s0; r)	Source	Action	Destination	Reward	Q	Formula	Value
(s1;Right; s4; 1:0)	S1	RIGHT	S4	1		(1-0.5)Q(s1,RIGHT) + 0.5(1+0.5 * maxa' Q(s',a'))	0.75

Q after this step

	UP	DOWN	LEFT	RIGHT
s1	0	0	0	0.75
s2	0.095	0	0	0
s3	0	0	0	0
s4	0	0	0	0

5. Assuming that the world resets after the agent visits state s_4, and the agent starts in state s_2, do these experience traces suggest that this is a greedy agent? Why or why not?

Yes , it is a greedy agent as it constantly performs the action that is believed to yield the highest expected reward.. But is it not an Epsilon-Greedy as Epsilon-Greedy tries to balance exploration and exploitation by choosing between exploration and exploitation randomly. The epsilon-greedy, where epsilon refers to the probability of choosing to explore, exploits most of the time with a small chance of exploring. In this case, there is no exploration happening

6. If a greedy agent were being used to generate experience traces for Q-Learning in this environment, would we be guaranteed to visit every state (in the limit)? What single aspect of the environment

could be changed to flip your answer (yes to no, no to yes)?

No, it is not guaranteed to visit every state as it will stop at reaching goal state.
May be setting the initial Q-values are set sufficiently large will try to visit all states.