RollNo:20171184
2018121005

TEAM_32

ASSIGNMENT-2

Q2.

1a.

Input
DiscountFactor = 0.1
StepCost = -32/10

Output
iteration : 4
delta : 0.005606400000000011
utilities:

3.2	0	-0.68	32.0
-3.009	-3.511	-3.294	-0.68
-3.511	-3.552	0	-3.321
-3.552	-3.555	-6.4	-3.553

policy:

```
3.2   -    E     32.0
N     W    E     N
N     N    -     N
N     W    -6.4  E
```

Observation: Here with the Discount Factor – 0.1, Taking the start state as (3,0) we can clearly see that the policy ends up in the state where the utility is “3.2” (0,0). But there can be a better policy which can end up in state with value “32.0”. policy is (3,0) > (2,0) > (1,0) > (0,0).

1b.

Input
DiscountFactor = 0.99
StepCost = -32/10

Output

iterations : 27
delta : 7.999758934396084e-05
utilities:

3.2	0	27.135	32.0
12.637	18.417	23.283	27.135
9.432	13.673	0	22.807
5.568	7.547	-6.4	15.793

policy:

```
3.2   -     E     32.0
E     E     E     N
E     N     -     N
N     N     -6.4  N
```

Observation: Here with the Discount Factor as -0.99, taking the start state as (3,0) the policy gives the path to highest valued end state “32.0” (0,3). Which has highest utility. policy is (3,0) > (2,0) > (2,1) > (1,1) > (1,2) > (1,3) > (0,3). With higher Discount value we reach the policy with least error.

2a.

Input
DiscountFactor = 0.99
StepReward = 32

Output
iterations : 1262
delta : 0.0001002606154543173
utilities:

3.2	0	3199.99	32.0
3199.99	3199.99	3199.99	3199.99
3199.99	3199.99	0	3199.99
3199.99	3199.99	-6.4	3199.99

policy:

```
3.2   -     W     32.0
S     S     S     S
S     S     -     S
S     W     -6.4  E
```

Observation: Here As the Step Reward is positive the policy tries to move more and more without reaching end state. Taking (3,0) as start state the action is south which is not possible so it stays back in the same state increasing the utility with positive step reward. Policy is (3,0) > (3,0) > (3,0) ......>(3,0)

2b.

Input

DiscountFactor = 0.99
StepReward = -32/5

Output

iterations : 25
delta : 5.345956564539733e-05
utilities:

3.2	0	22.713	32.0
-2.182	6.092	15.358	22.713
-9.326	-2.773	0	14.45
-16.494	-10.863	-6.4	4.895

policy:

```
3.2   -     E     32.0
E     E     E     N
N     N     -     N
N     N     -6.4  N
```

Observation: Here we see the policy tries to reach the highest valued end state. And as the step reward is negetive it took less moves to reach end state with hiest utility. Consider the start state (3,0) we see it moving from (3,0) > (2,0) > (1,0) > (1,1) > (1,2) > (1,3) > (0,3)

2c.

Input

DiscountFactor = 0.99
StepReward = -32/4

Output

iterations : 21 delta : 5.914087632774567e-05 utilities:

3.2	0	20.502	32.0
-6.069	-0.026	11.395	20.502
-15.378	-10.592	0	10.271
-24.118	-15.669	-6.4	-0.554

policy:

```
3.2   -     E     32.0
N     E     E     N
N     N     -     N
N     E     -6.4  N
```

Observation: Here consider the policy with start state as (3,0). It is (3,0) > (2,0) > (1,0) > (0,0) . It ends up in end state with utility value “3.2” . As the Step cost is more than before and it takes more cost to move to end state with highest utility.

2d.

Input

DiscountFactor = 0.99
StepReward = -32

Output

iterations : 18 delta : 4.445219158810687e-05
utilities:

3.2	0	-12.666	32.0
-41.713	-81.997	-48.042	-12.666
-81.997	-89.317	0	-52.408
-89.317	-50.956	-6.4	-46.9

policy:

```
3.2   -     E     32.0
N     W     E     N
N     S     -     N
E     E     -6.4  W
```

Observation: Here we see the step cost is very high so the policy tries to reach an end state as soon as it can. The policy from start state as (3,0) is (3,0) > (3,1) > (3,2).

OVERALL OBSERVATION:

- As the value of Discount Factor increrase with constant Step Cost, The policy tries to reach better utilities.
- When the Step Reward is positive ,The policy tries to move increasing the utility without goal state.
- When the Step Reward is very high negetive value ,The policy tries to reach the goal state as soon as it can.