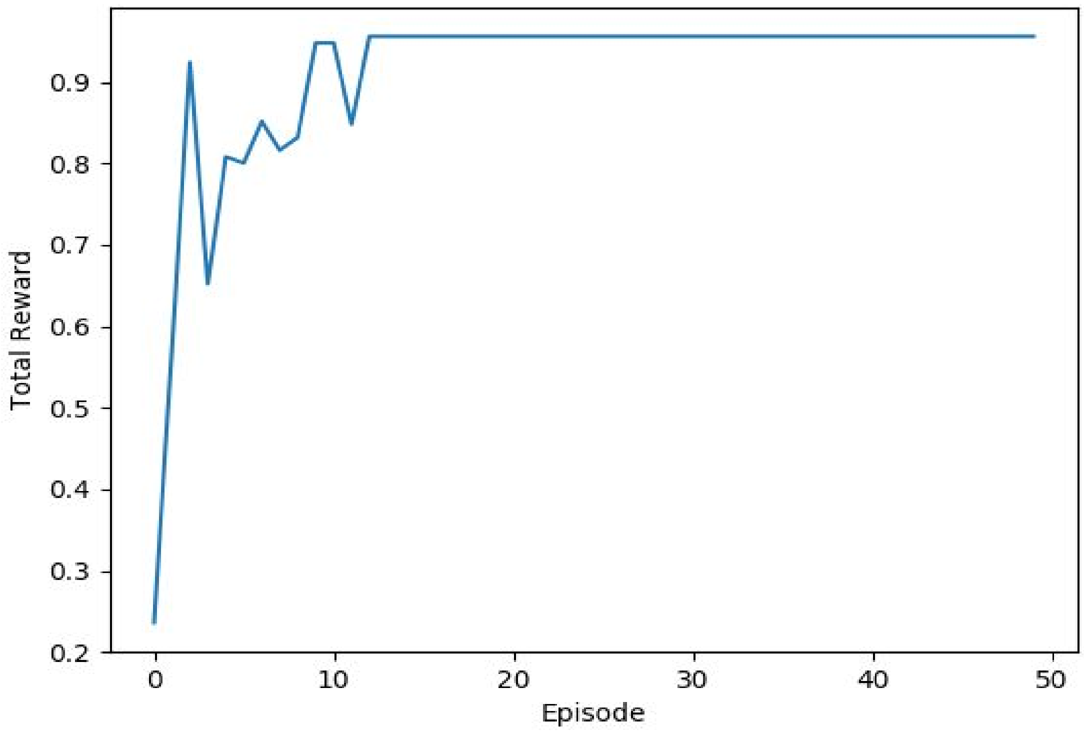
Programming Assignment - 4

Kshitij Srivastava MT18099

Question 3:

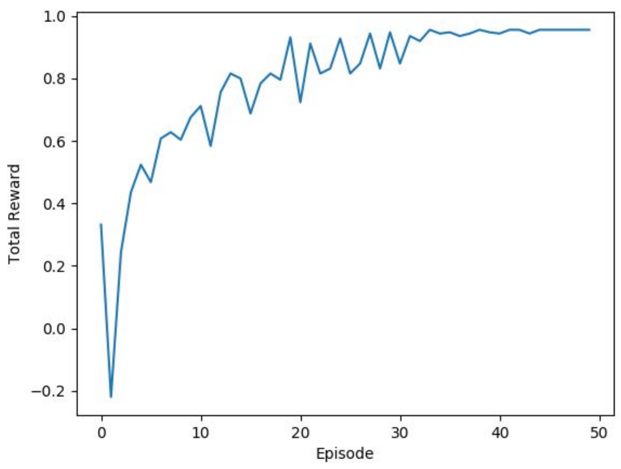
(a) Learning Curve:



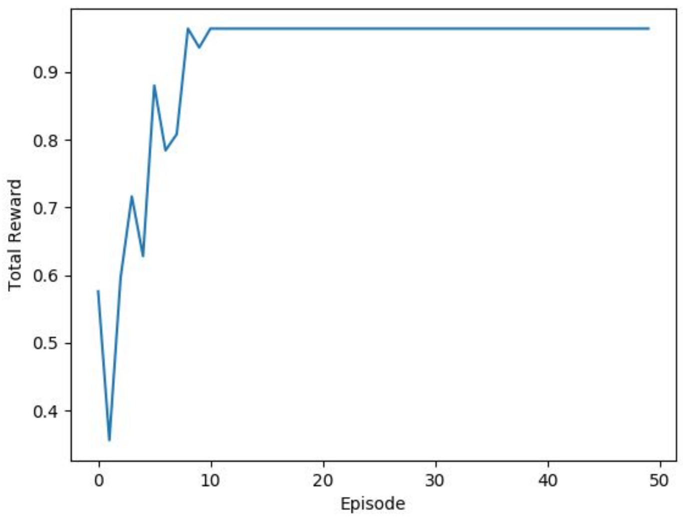
(b) Hyperparameters:

- Gamma (γ) is the Discount factor, which is responsible for deciding how far in future the algorithm looks for rewards. Its value is from 0 to 1. If the value is 0, the algorithm focuses on immediate results instead of future results.
- Alpha (α) is the learning rate. It has a direct relationship with the updation of Q-table values. If it is closer to 0, the algorithm learns slowly and if it is closer to 1, the algorithm learns quickly.

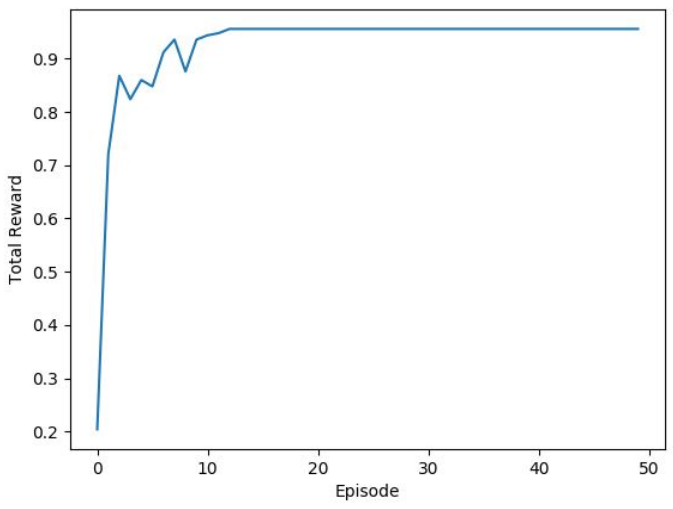
- Alpha = 0.1



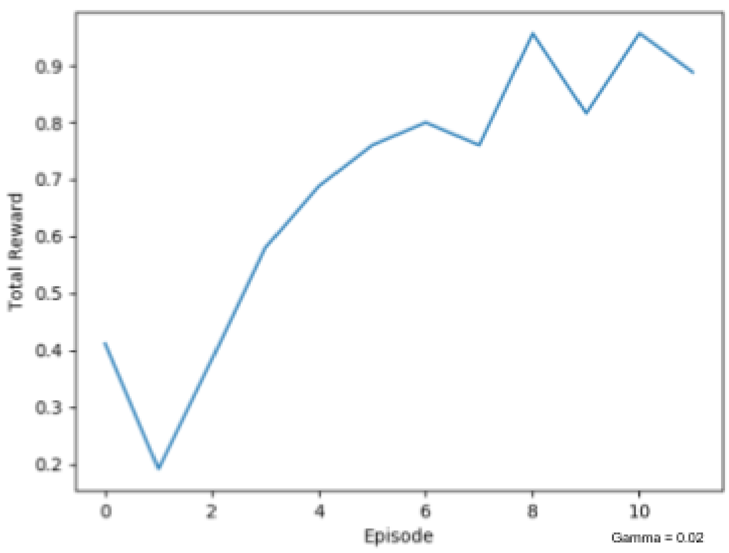
- Alpha = 0.5



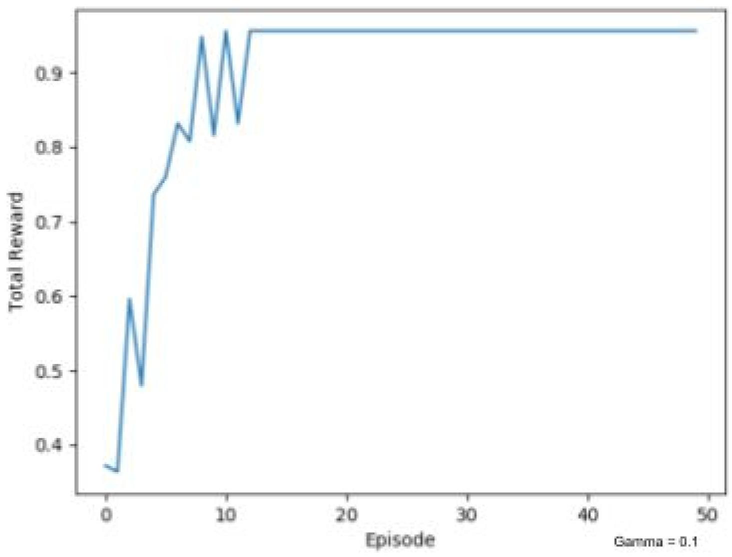
- Alpha = 0.9



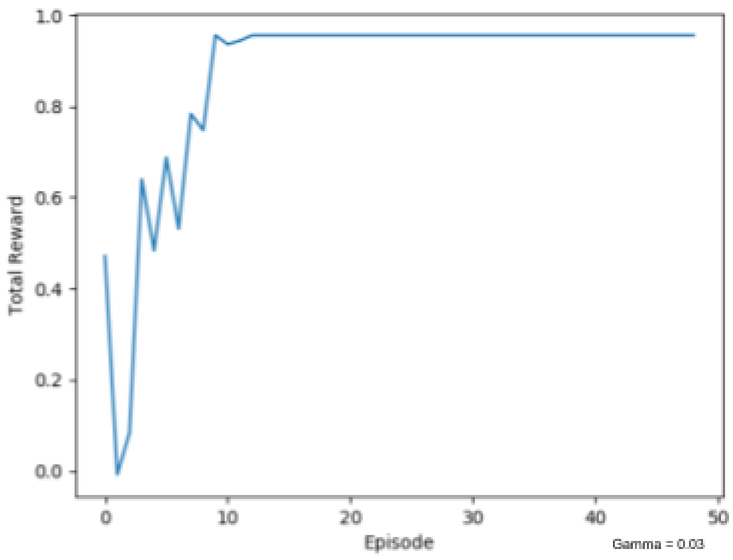
- <u>Plot for gamma values is given on the next page</u> :



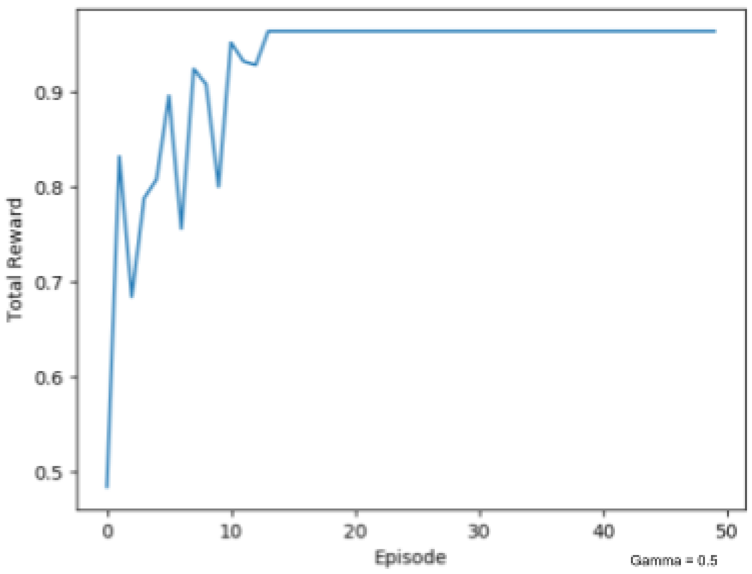
Gamma = 0.02



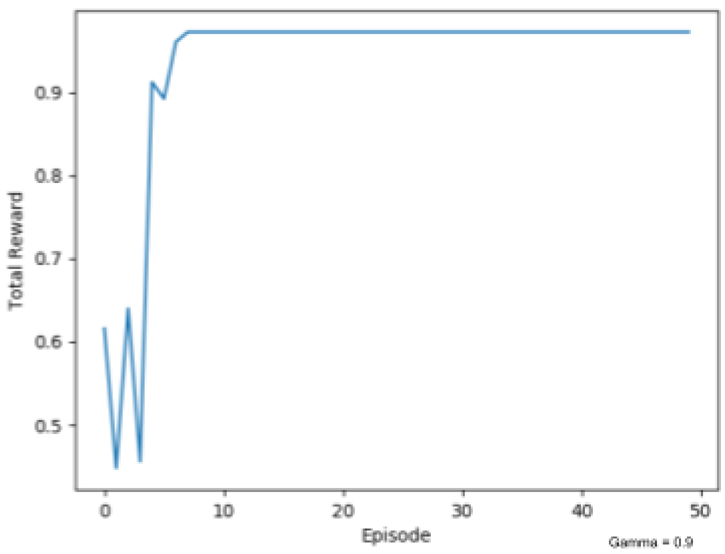
Gamma = 0.1



Gamma = 0.03



Gamma = 0.5

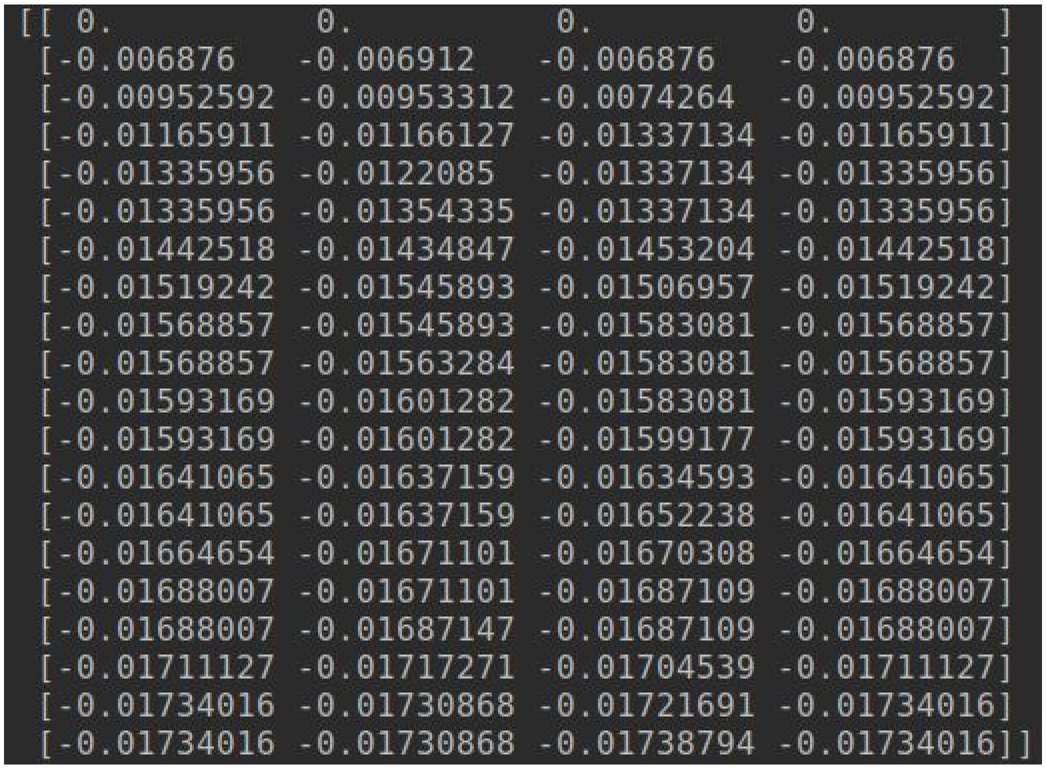


Gamma = 0.9

- On increasing the value of alpha, the learning rate of the algorithm increases, i.e., the algorithm is able to learn quickly, as is evident from the graphs.
- On increasing the value of gamma, the algorithm is able to see farther into the states for future rewards. Thus, it learns more efficiently and the maximum reward value converges faster.

(c) Changes in probability of a state's actions :

State [0][0] actions:

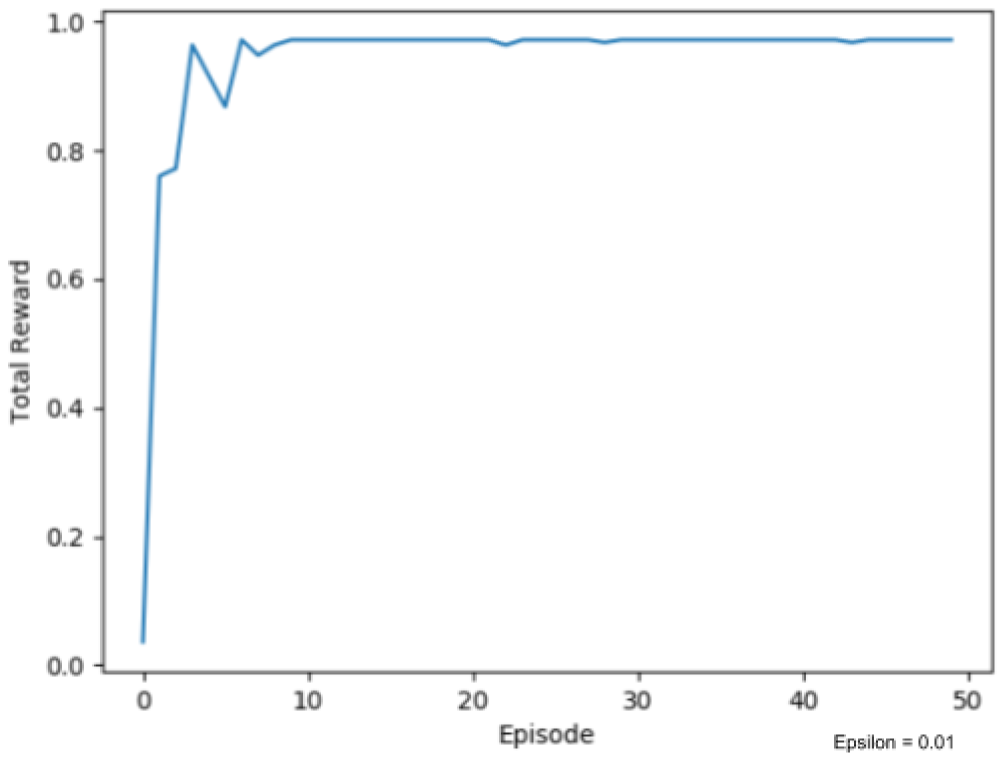


```
[[ 0.          0.          0.          0.        ]
 [-0.006876   -0.006912   -0.006876   -0.006876  ]
 [-0.00952592 -0.00953312 -0.0074264  -0.00952592]
 [-0.01165911 -0.01166127 -0.01337134 -0.01165911]
 [-0.01335956 -0.0122085  -0.01337134 -0.01335956]
 [-0.01335956 -0.01354335 -0.01337134 -0.01335956]
 [-0.01442518 -0.01434847 -0.01453204 -0.01442518]
 [-0.01519242 -0.01545893 -0.01506957 -0.01519242]
 [-0.01568857 -0.01545893 -0.01583081 -0.01568857]
 [-0.01568857 -0.01563284 -0.01583081 -0.01568857]
 [-0.01593169 -0.01601282 -0.01583081 -0.01593169]
 [-0.01593169 -0.01601282 -0.01599177 -0.01593169]
 [-0.01641065 -0.01637159 -0.01634593 -0.01641065]
 [-0.01641065 -0.01637159 -0.01652238 -0.01641065]
 [-0.01664654 -0.01671101 -0.01670308 -0.01664654]
 [-0.01688007 -0.01671101 -0.01687109 -0.01688007]
 [-0.01688007 -0.01687147 -0.01687109 -0.01688007]
 [-0.01711127 -0.01717271 -0.01704539 -0.01711127]
 [-0.01734016 -0.01730868 -0.01721691 -0.01734016]
 [-0.01734016 -0.01730868 -0.01738794 -0.01734016]]
```

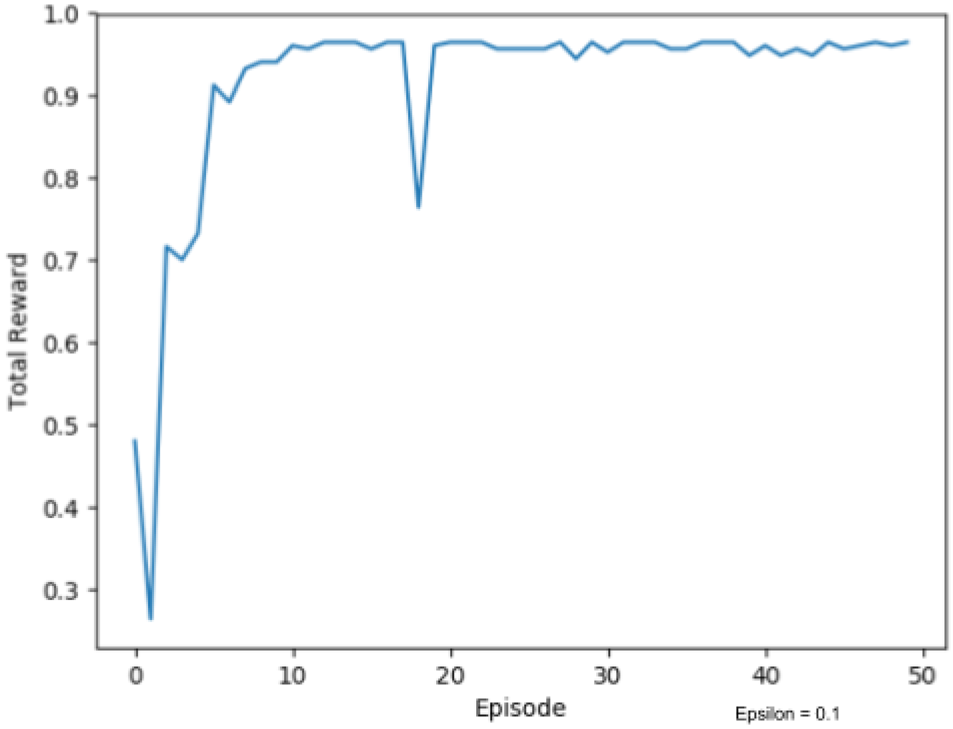
- As observed from the q-table for State[0][0] actions, we can see that the value of each of the four actions is being updated as the learning progresses.

(d) Random Exploration :

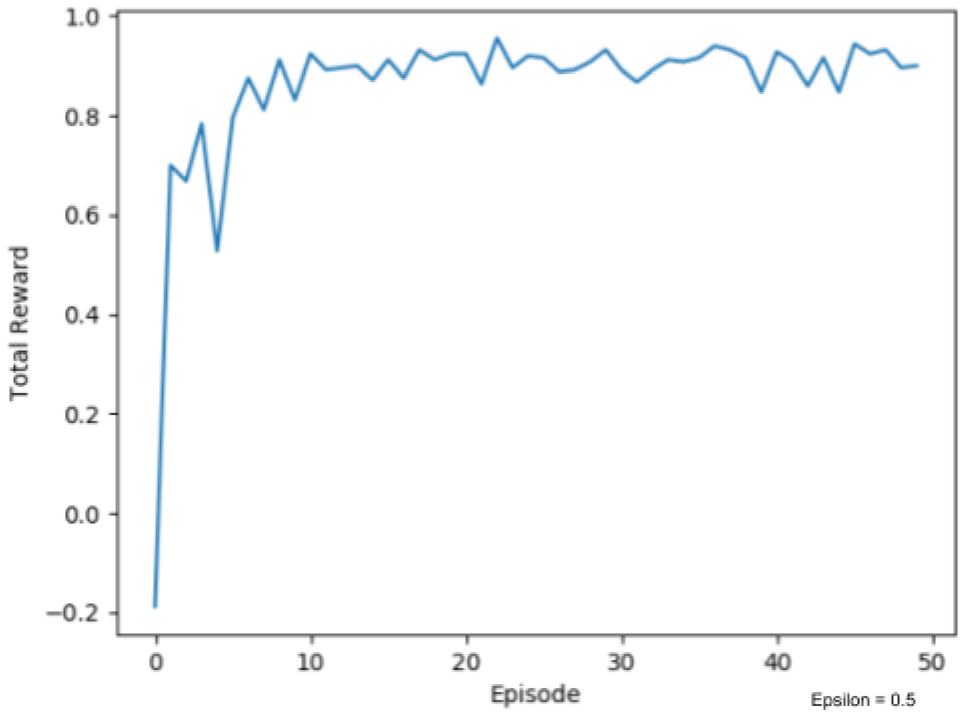
- Random exploration can be introduced by using a constant (epsilon) for it.
- We randomly generate a value. If the generated value is above below the constant value, the agent takes a random move at the present state otherwise follows the state from the Q-table.
- The plot for different values of epsilon is attached below.

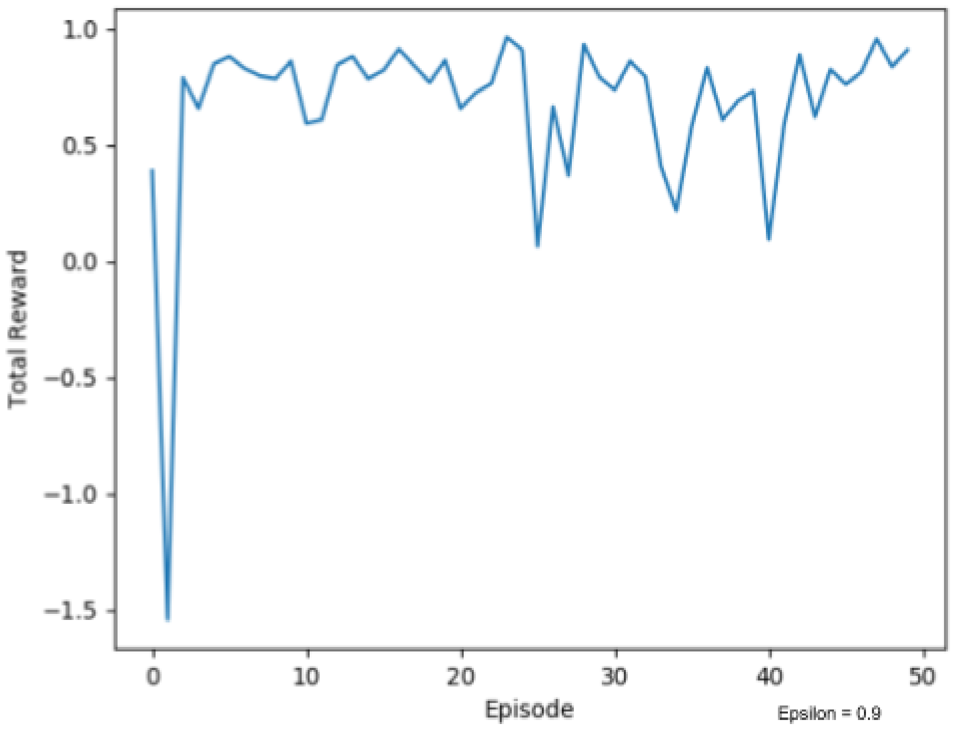


Epsilon = 0.01



Epsilon = 0.1

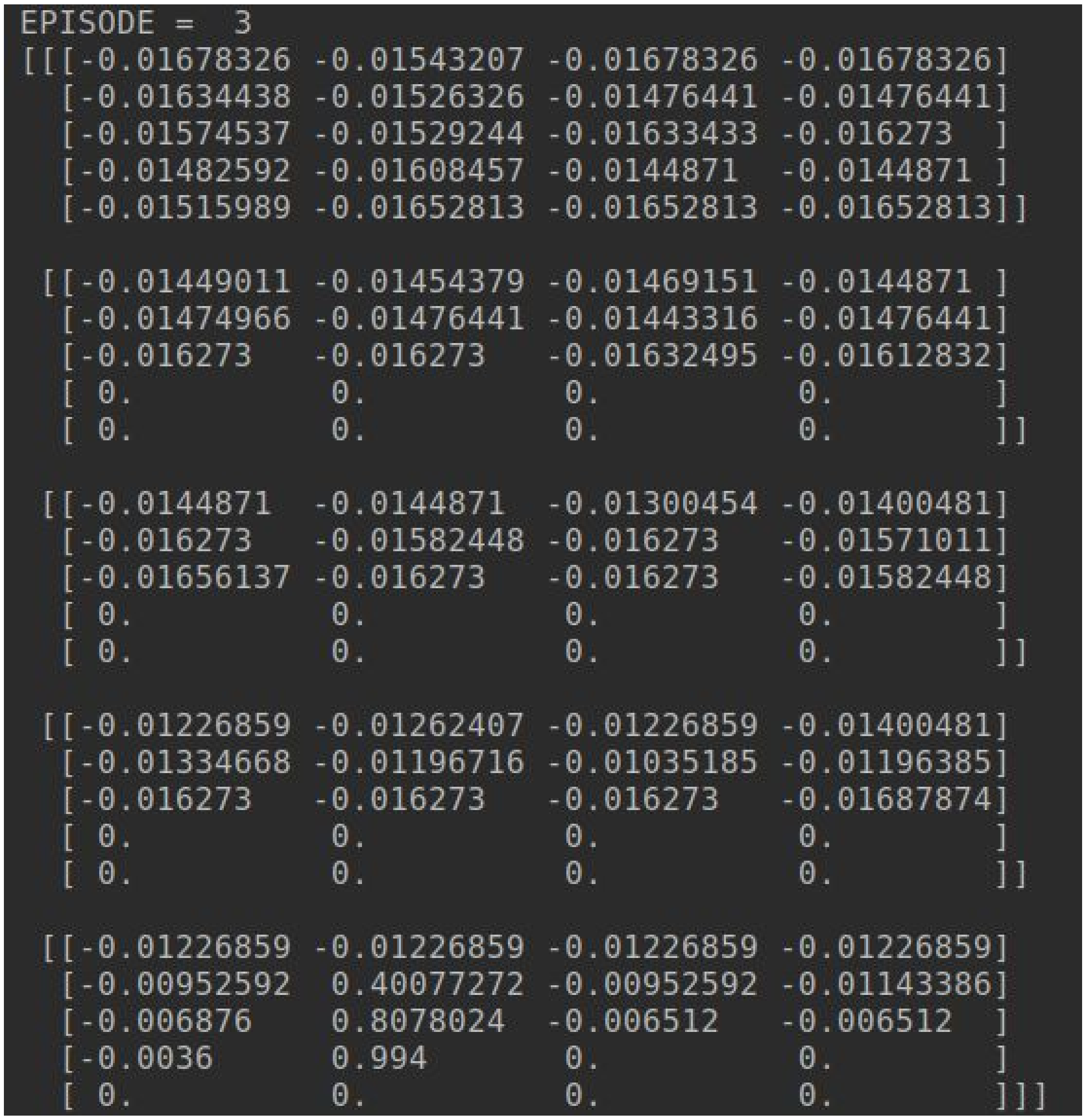




- With small values of epsilon, the agent takes very few exploratory steps. However, as we increase the value of epsilon, the agent starts taking more and more random steps because of which the total reward is seen to vary quite a lot.

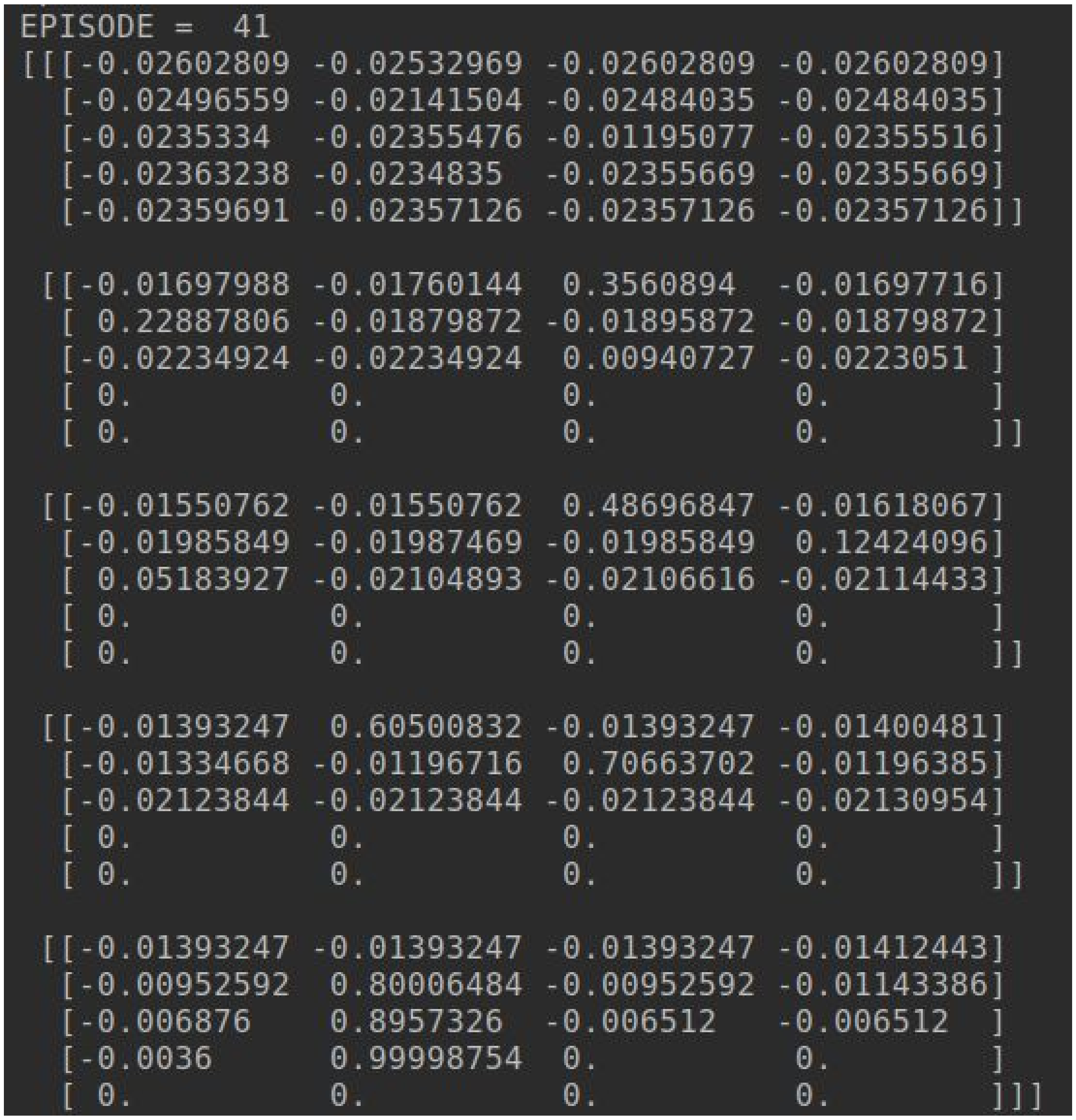
(e) Intermediate state observations :

Episode 3 Q_table:



```
EPISODE =  3
[[[-0.01678326 -0.01543207 -0.01678326 -0.01678326]
  [-0.01634438 -0.01526326 -0.01476441 -0.01476441]
  [-0.01574537 -0.01529244 -0.01633433 -0.016273  ]
  [-0.01482592 -0.01608457 -0.0144871  -0.0144871 ]
  [-0.01515989 -0.01652813 -0.01652813 -0.01652813]]

 [[-0.01449011 -0.01454379 -0.01469151 -0.0144871 ]
  [-0.01474966 -0.01476441 -0.01443316 -0.01476441]
  [-0.016273   -0.016273   -0.01632495 -0.01612832]
  [ 0.          0.          0.          0.        ]
  [ 0.          0.          0.          0.        ]]

 [[-0.0144871  -0.0144871  -0.01300454 -0.01400481]
  [-0.016273   -0.01582448 -0.016273   -0.01571011]
  [-0.01656137 -0.016273   -0.016273   -0.01582448]
  [ 0.          0.          0.          0.        ]
  [ 0.          0.          0.          0.        ]]

 [[-0.01226859 -0.01262407 -0.01226859 -0.01400481]
  [-0.01334668 -0.01196716 -0.01035185 -0.01196385]
  [-0.016273   -0.016273   -0.016273   -0.01687874]
  [ 0.          0.          0.          0.        ]
  [ 0.          0.          0.          0.        ]]

 [[-0.01226859 -0.01226859 -0.01226859 -0.01226859]
  [-0.00952592  0.40077272 -0.00952592 -0.01143386]
  [-0.006876    0.8078024  -0.006512   -0.006512  ]
  [-0.0036      0.994       0.          0.        ]
  [ 0.          0.          0.          0.        ]]]
```

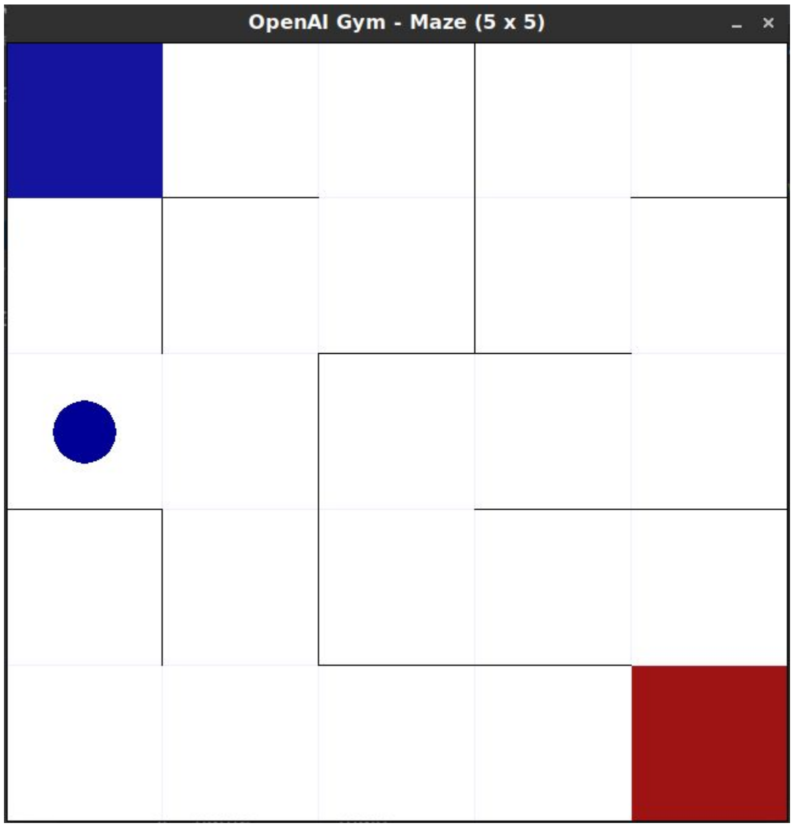
Episode 41 Q_table:



```
EPISODE =  41
[[[-0.02602809 -0.02532969 -0.02602809 -0.02602809]
  [-0.02496559 -0.02141504 -0.02484035 -0.02484035]
  [-0.0235334  -0.02355476 -0.01195077 -0.02355516]
  [-0.02363238 -0.0234835  -0.02355669 -0.02355669]
  [-0.02359691 -0.02357126 -0.02357126 -0.02357126]]

 [[-0.01697988 -0.01760144  0.3560894  -0.01697716]
  [ 0.22887806 -0.01879872 -0.01895872 -0.01879872]
  [-0.02234924 -0.02234924  0.00940727 -0.0223051 ]
  [ 0.          0.          0.          0.        ]
  [ 0.          0.          0.          0.        ]]

 [[-0.01550762 -0.01550762  0.48696847 -0.01618067]
  [-0.01985849 -0.01987469 -0.01985849  0.12424096]
  [ 0.05183927 -0.02104893 -0.02106616 -0.02114433]
  [ 0.          0.          0.          0.        ]
  [ 0.          0.          0.          0.        ]]

 [[-0.01393247  0.60500832 -0.01393247 -0.01400481]
  [-0.01334668 -0.01196716  0.70663702 -0.01196385]
  [-0.02123844 -0.02123844 -0.02123844 -0.02130954]
  [ 0.          0.          0.          0.        ]
  [ 0.          0.          0.          0.        ]]

 [[-0.01393247 -0.01393247 -0.01393247 -0.01412443]
  [-0.00952592  0.80006484 -0.00952592 -0.01143386]
  [-0.006876    0.8957326  -0.006512   -0.006512  ]
  [-0.0036      0.99998754  0.          0.        ]
  [ 0.          0.          0.          0.        ]]]
```

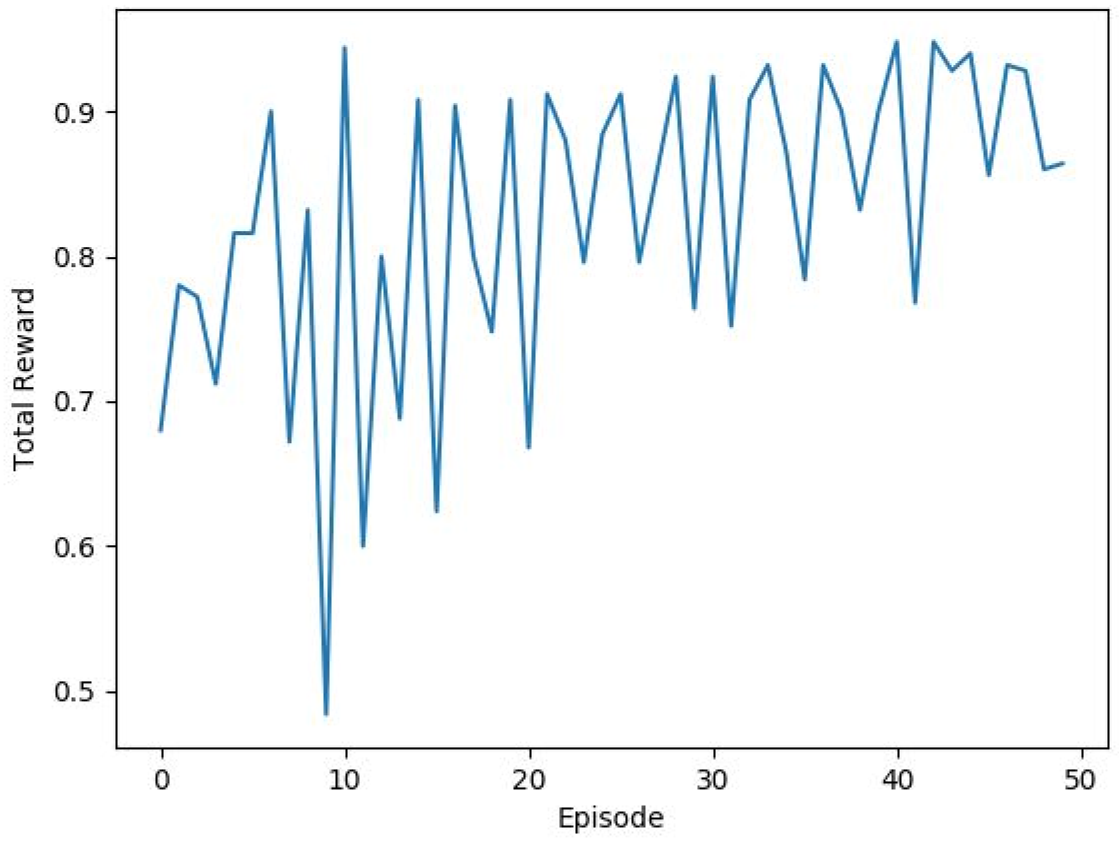
Question 4:

- While testing the agent trained in question-3 on a new maze, the agent gets stuck at the same state as it is not able to find the best move.
- There can be various reasons for this. It might be because there is no valid move available, even if the Q-table suggests one.
- Or there is a move available according to the maze but the agent isn't aware of it.
- The agent is not able to make any move in such a case and gets stuck in the maze.



- **Solution:**
    - The problem can be solved by updating the Q-table for the new maze as well.
    - So initially, the agent moves according to the old Q-table but as it progresses, it updates its Q-table and learns the best path for the new maze.
    - Hence, updating the Q-table is a crucial step for an agent to adapt to newer environments.

SARSA :



- We can see that the agent explores a lot more than Q-learning as the action for each state is taken in a epsilon-greedy manner instead of simply choosing the action with the largest Q-value.