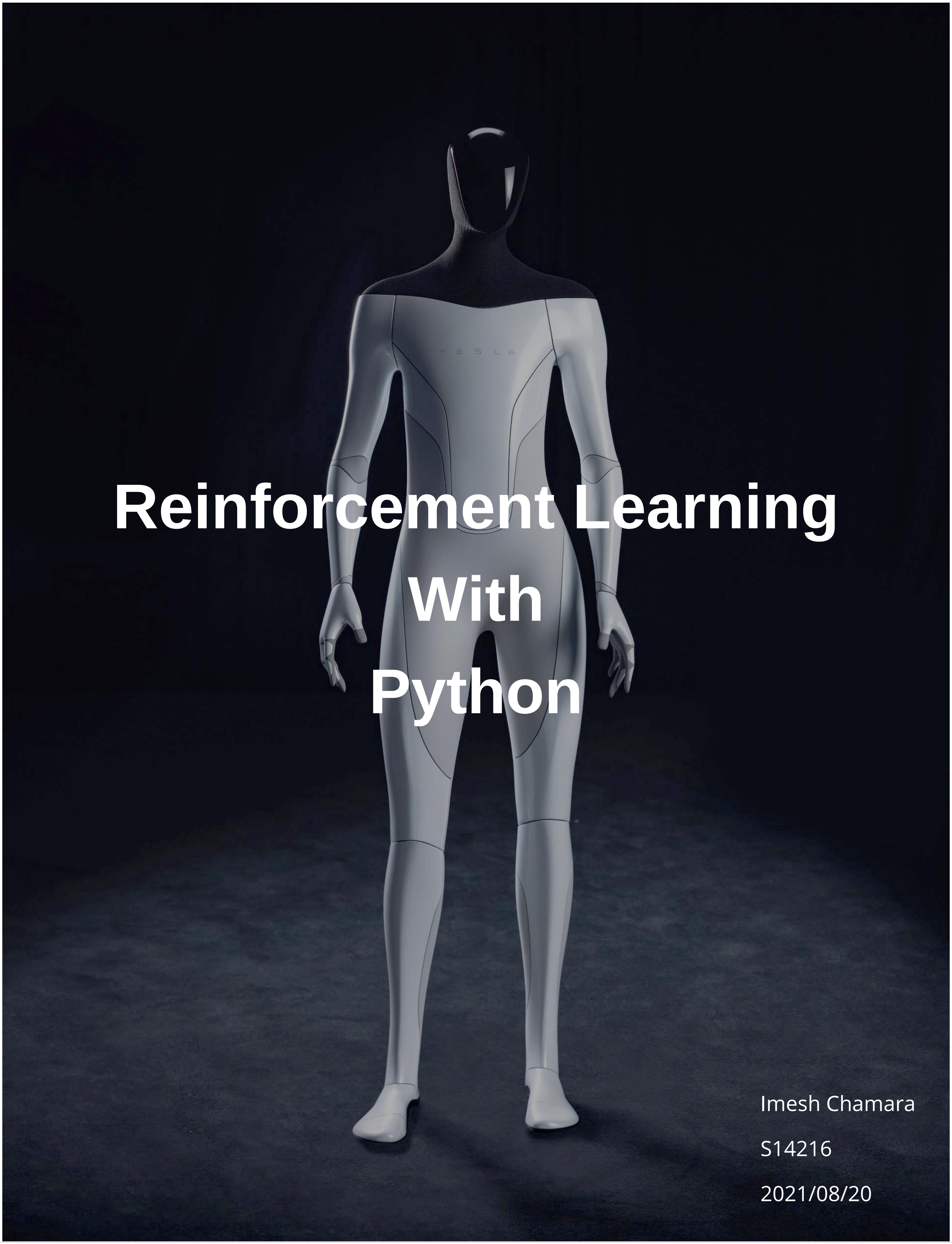
Reinforcement Learning With Python



Imesh Chamara

S14216

2021/08/20

Introduction

Reinforcement learning can be considered as one of the machine learning models that is used in autonomous prosthetic legs and autonomous cars. When considering elements that can be seen in reinforcement learning, there are five main elements. They are,

1) The agent and the environment
2) A policy
3) A reward signal
4) A value function
5) A model of the environment

Q-learning is a model-free reinforcement learning algorithms and languages like Python and GNU Octave can be used to implement Q-learning. In this assignment Python is used for coding.

There are few advantages and disadvantages in reinforcement learning.

Advantages

- It is innovative.
- Doesn't need large labeled dataset.
- It's goal oriented learning algorithm

Disadvantages

- Need lot of data and lot of computation.
- Not preferable for solving problems.

Content

1) Part 1 - Practice the code examples given in https://www.viralml.com/video-content.html?v=nSxaG_Kjw_w using Jupyter or Google Colab (http://colab.research.google.com/).

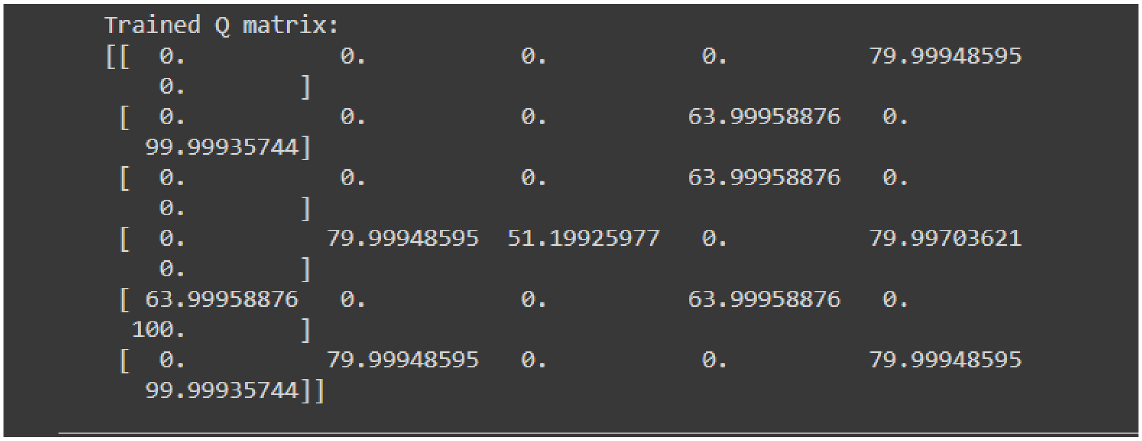
   Google Colab link: https://colab.research.google.com/drive/1-wk7okXHBf5dO7pM8PJTMTuLmt785h3A?usp=sharing

2) Part 2 - Model the Example 1 problem discussed in the lecture note using Python and compare the outputs/results (Octave vs. Python)

Google Colab link: https://colab.research.google.com/drive/1rBHQvWu4l2_YmLTCmbYlHL5Lo37HuY0z?usp=sharing

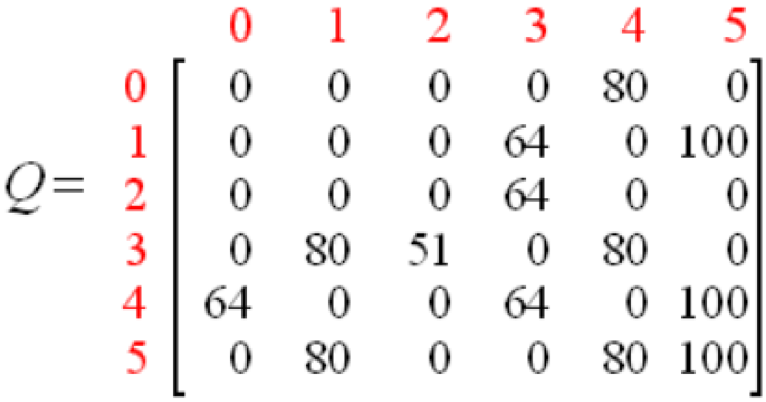
Most efficient path when starting point is 2: [2, 3, 4, 5]

Comparison of outputs



```
Trained Q matrix:
[[  0.           0.           0.           0.          79.99948595
    0.        ]
 [  0.           0.           0.          63.99958876   0.
   99.99935744]
 [  0.           0.           0.          63.99958876   0.
    0.        ]
 [  0.          79.99948595  51.19925977   0.          79.99703621
    0.        ]
 [ 63.99958876   0.           0.          63.99958876   0.
  100.        ]
 [  0.          79.99948595   0.           0.          79.99948595
   99.99935744]]
```

Output Q table when using Python (gamma = 0.8)

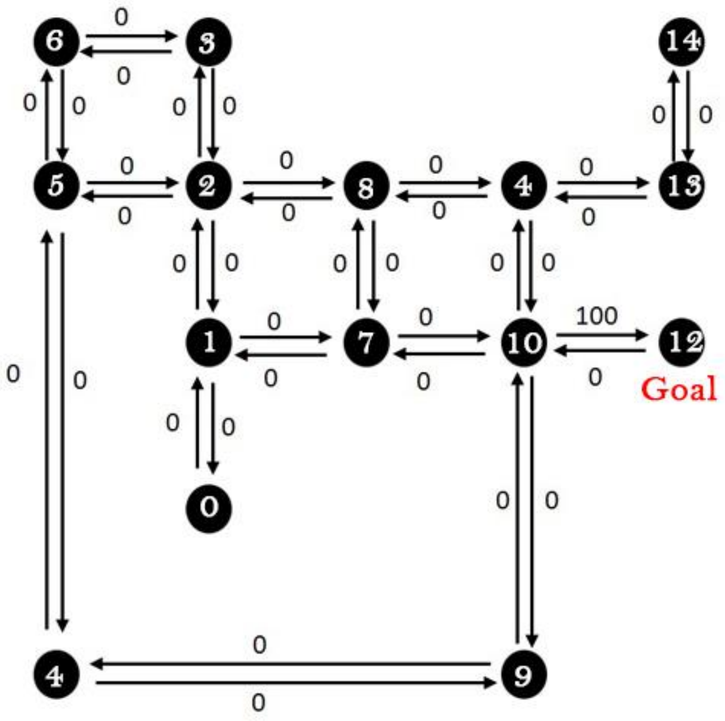


$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\ \left[\begin{array}{cccccc} 0 & 0 & 0 & 0 & 80 & 0 \\ 0 & 0 & 0 & 64 & 0 & 100 \\ 0 & 0 & 0 & 64 & 0 & 0 \\ 0 & 80 & 51 & 0 & 80 & 0 \\ 64 & 0 & 0 & 64 & 0 & 100 \\ 0 & 80 & 0 & 0 & 80 & 100 \end{array}\right] \end{array}$$

Output Q table when using GNU Octave (gamma = 0.8)

Python	GNU Octave
There are numbers with decimal points in the Q matrix.	There are no numbers with decimal points in the Q matrix.

3) Part 3 - Give your answer by modeling the Exercise problem (maze solving) given in the lecture note.



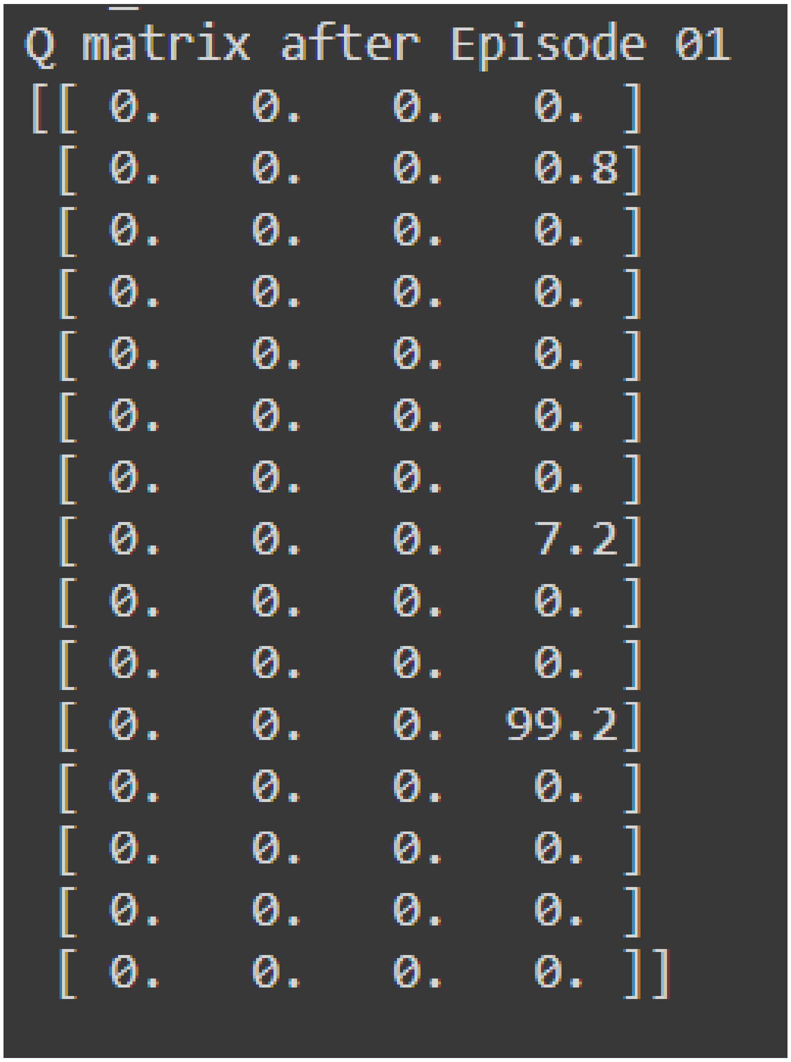
Google Colab link:
https://colab.research.google.com/drive/1R7XDt0DLKpn5mn6rC16vBrarvT9HAAQy?usp=sharing

Most efficient path:
[0, 1, 7, 10, 12]
[array([0]), array([3]), array([3]), array([3])]

Resulting Q tables

- Start = (0, 0); Actions: UP, RIGHT, RIGHT, RIGHT

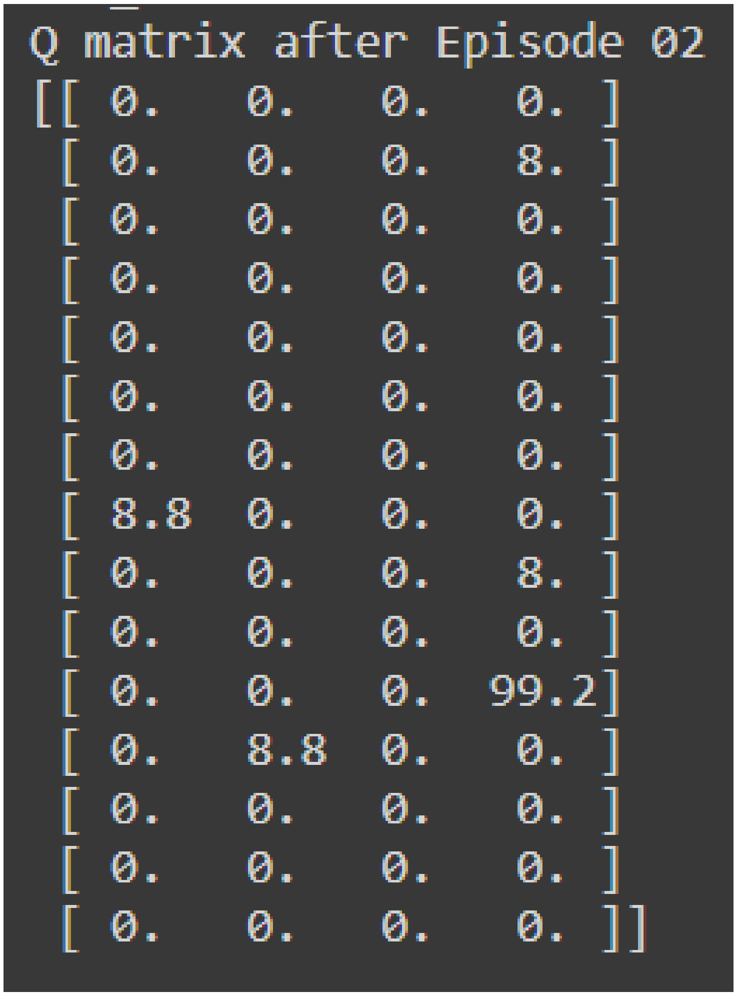
  Episode_1 = [0,1,7,10,12]



```
Q matrix after Episode 01
[[ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0.8]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   7.2]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.  99.2]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]]
```

- Start = (0, 2), Actions: DOWN, RIGHT, UP, RIGHT, DOWN, RIGHT

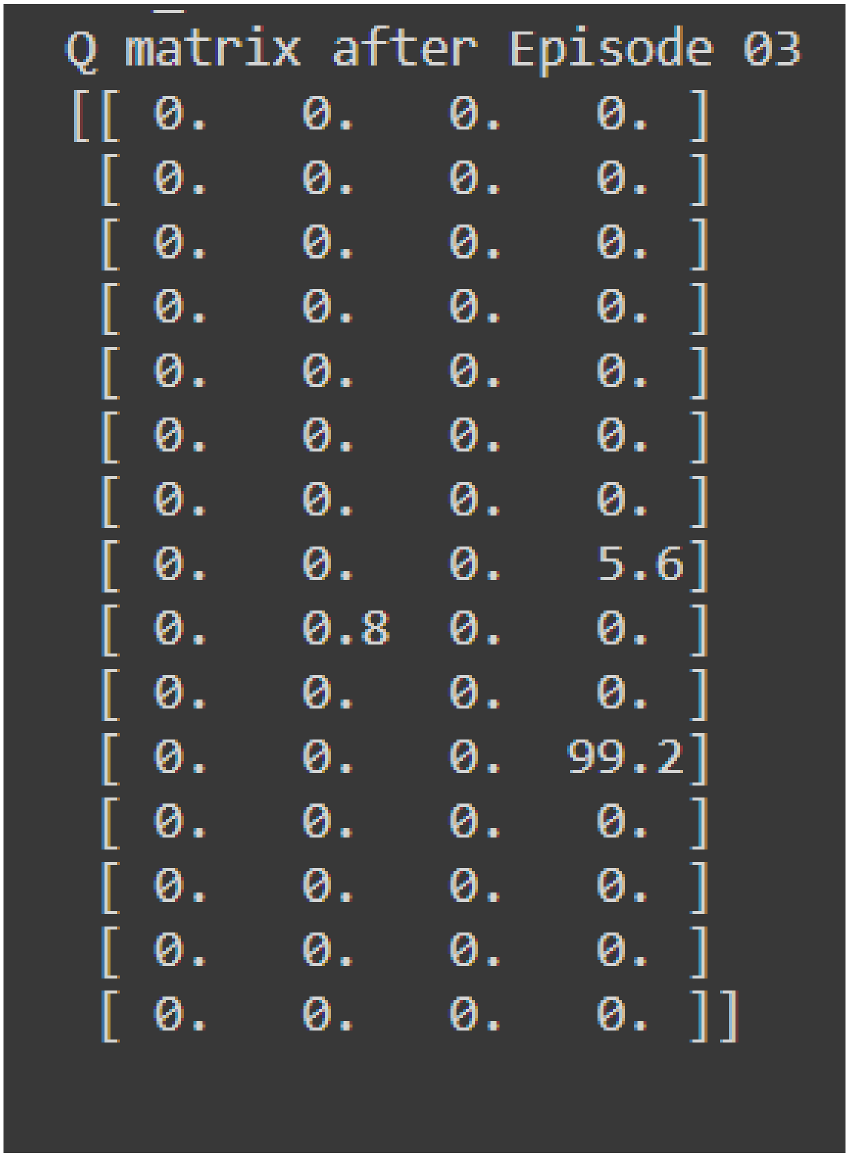
  Episode_2 = [2,1,7,8,4,10,12]



```
Q matrix after Episode 02
[[ 0.   0.   0.   0. ]
 [ 0.   0.   0.   8. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 8.8  0.   0.   0. ]
 [ 0.   0.   0.   8. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.  99.2]
 [ 0.   8.8  0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]]
```

- Start = (1, 2); Actions: DOWN, RIGHT, RIGHT

Episode_3 = [8,7,10,12]



```
Q matrix after Episode 03
[[ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   5.6]
 [ 0.   0.8  0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.  99.2]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]
 [ 0.   0.   0.   0. ]]
```

Reference

(1) Amunategui, M.; Amunategui, M. Reinforcement Learning - A Simple Python Example and A Step Closer to AI with Assisted Q-Learning https://www.viralml.com/video-content.html?v=nSxaG_Kjw_w. (accessed Aug 21, 2021).

(2) codezup. Advantages of Reinforcement Learning – Artificial Intelligence https://codezup.com/advantages-of-reinforcement-learning-artificial-intelligence/ (accessed Aug 21, 2021).

(3) Shyalika, C. A Beginners Guide to Q-Learning https://towardsdatascience.com/a-beginners-guide-to-q-learning-c3e2a30a653c. (accessed Aug 21, 2021).