**Machine Learning Lab**

**Assignment 5**

**Name - Pritesh Kumar Sahani**

**Roll - 001811001005**

**Semester - 7**

**Year - 4**

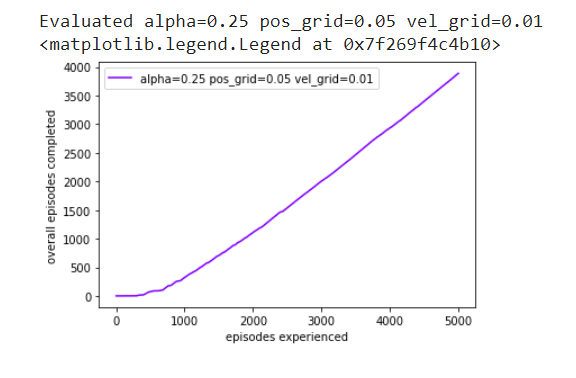
**Department - Information Technology**

**GITHUB LINK:** [**https://github.com/stepupgithub/Machine-Learning-Assignments**](https://github.com/stepupgithub/Machine-Learning-Assignments)

**ENTIRE ASSIGNMENT LINK (GOOGLE COLLAB + COMPARISON TABLE):** [**https://drive.google.com/drive/folders/10Rr-v3gUxJ4mo5A8ryoNqFCvSWHFpqHG?usp=sharing**](https://drive.google.com/drive/folders/10Rr-v3gUxJ4mo5A8ryoNqFCvSWHFpqHG?usp=sharing)

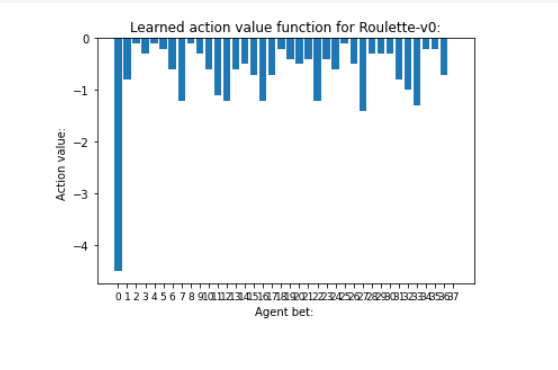
**1) *Reinforcement Learning***

**1.1) MOUNTAIN CAR**

****

**Mountain Car, a standard testing domain in Reinforcement Learning, is a problem in which an under-powered car must drive up a steep hill. Since gravity is stronger than the car's engine, even at full throttle, the car cannot simply accelerate up the steep slope. The car is situated in a valley and must learn to leverage potential energy by driving up the opposite hill before the car is able to make it to the goal at the top of the rightmost hill.**

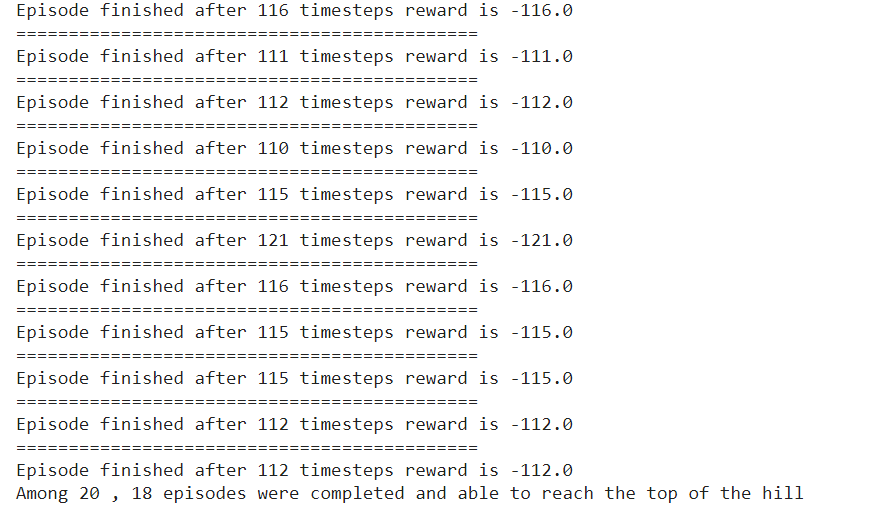
**1.2) ROULETTE**

****

**The game consists of a space of 36 random outcomes numbered 1 through 36. It is possible to bet on individual numbers or sets of them — even, odd, first third, second third, third third, first half, or second half.**

**2) *Deep Reinforcement Learning***

**2.1) MOUNTAIN CAR**

****

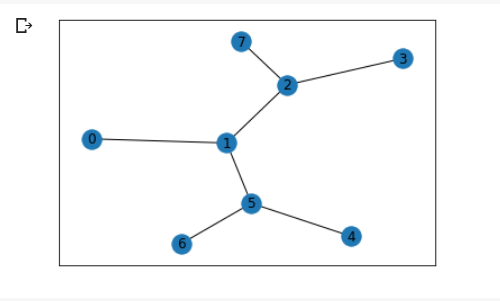
**In Q-learning we use a Q-table that outputs the maximum expected value of each action in each state, given the state and action. In Deep Q-learning, we use a neural network to estimate the Q-values of all the possible actions in each state.**

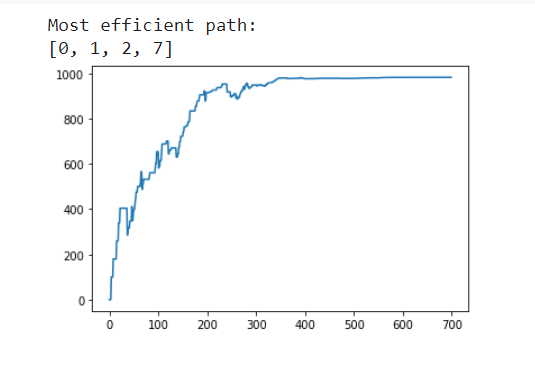
**2.2) ROULETTE**

****

**3) *Shortest Path Algorithm***

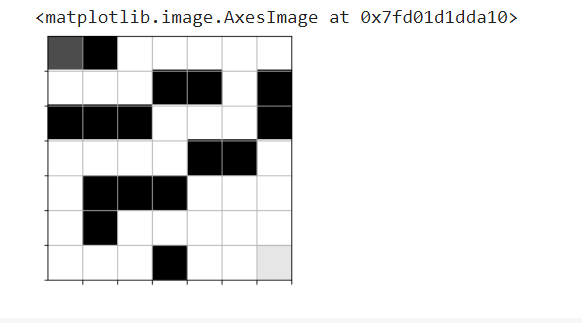
**3.1) REINFORCEMENT LEARNING**

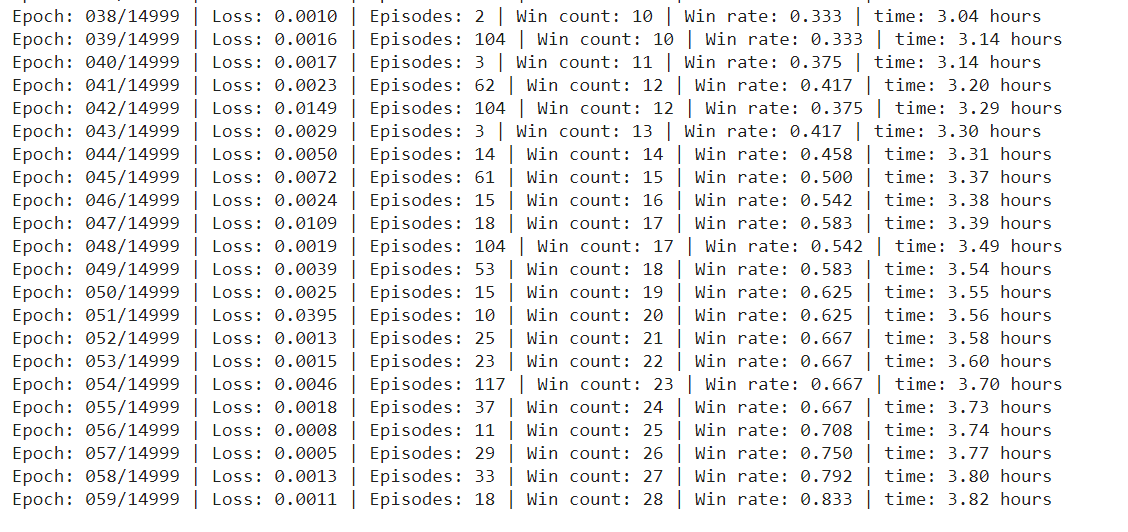
****

****

**Optimal path management is key when applied to disk I/O or network I/O. The efficiency of a storage or a network system depends on optimal routing of I/O. To obtain optimal path for an I/O between source and target nodes, an effective path finding mechanism among a set of given nodes is desired.**

**3.2) DEEP REINFORCEMENT LEARNING**

****

****

**Traditional maze puzzles have been used a lot in data structures and algorithms research and education. The well-known Dijkstra shortest path algorithm is still the most practical method for solving such puzzles, but due to their familiarity and intuititive nature, these puzzles are quite good for demonstrating and testing Reiforcement Learning techniques.**