CAP 5636: Assignment 4 Explanations

Problem 1:

Implement an explicit policy for the mountain car environment without using any learning algorithm. Explain in detail your reasoning behind your policy and run several test episodes to measure its performance.

Explanation: In this problem, the action variable can have 3 values, 0 1 or 2, representing 'left', 'no push' and 'right' respectively. The observation variable can have 2 values, position and velocity. The explicit policy that I have used is if the velocity is negative then I return 0 where it knows that it needs to move left else, return 2 where it moves right.

PSEUDO CODE:

```
if velocity < 0: return 0
elif velocity >= 0:
return 2
```

Problem 2:

Implement an explicit policy for the cartpole environment without using any learning algorithm. Explain in detail your reasoning behind your policy and run several test episodes to measure its performance.

Explanation: In this cartpole problem, the aim is to balance the pole on the cart. The action variable may take two values 1, if you want the cart to move right and 0, if you want it to move left. The observation number will either be positive or negative. Here, I have used velocity as the important parameter (as a policy). If the velocity is negative I returned 0 (go left) else 1 (go right).

PSEUDO CODE:

```
action = 0 if state < 0 else 1
action = explicit_policy(observation[2])
```

Problem 3:

Apply the cross-entropy method to mountain car. Explain how many episodes are needed to learn a good policy. Explain which reward you use (original, modified).

Explanation: The Cross-Entropy method is a technique frequently used for rare event simulation and optimization. The basic steps of cross-entropy consist of generating random data samples and maintaining a distribution of good samples according to some scoring mechanism to generate new samples from. Cross-entropy can be applied to reinforcement learning by learning (optimizing) a value function. The cross-entropy method for MountainCar problem took 140 Episodes and 47 iterations to achieve an average score of 90.83.