CS444 Approximate Reinforcement Learning

1. The robot in the maze below starts at location S. At each time step he has the option of taking one of two actions LEFT, or RIGHT. These actions deterministically move the robot one space to the left or right. When the robot exits state A, he receives a reward of 0.75. When he exits state B he receives a reward of 1.0. Any action taken in states A or B end the trial: the robot exits the maze and no more steps are taken. Assume a discount factor of $\gamma = .9$.

A			S						B

For this question you will use approximate Q-learn to find a policy for this environment. Recall that we can approximate a Q-function as follows:

$$Q(s,a) = \sum_{i=0}^{n} f_i(s,a) w_i$$

Where the $f_i(s,a)$ are feature functions and the w_i are weights that are associated with each feature.

For this domain we will use the following 5 features, and assume that all weights are initialized to 0:

$f_0(s,a) = 1$ (This is a bias feature)

$$f_1(s,a) = \begin{cases} 1, & \text{if } s \text{ is a state in the left half of the maze and } a \text{ is LEFT} \\ 0, & \text{otherwise} \end{cases}$$

$$f_2(s,a) = \begin{cases} 1, & \text{if } s \text{ is a state in the left half of the maze and } a \text{ is RIGHT} \\ 0, & \text{otherwise} \end{cases}$$

$$f_3(s,a) = \begin{cases} 1, & \text{if } s \text{ is a state in the right half of the maze and } a \text{ is LEFT} \\ 0, & \text{otherwise} \end{cases}$$

$$f_4(s,a) = \begin{cases} 1, & \text{if } s \text{ is a state in the right half of the maze and } a \text{ is RIGHT} \\ 0, & \text{otherwise} \end{cases}$$

The update equation for approximate Q-learning is:

$$w_i \leftarrow w_i + \alpha(r + \gamma \max_{a'} Q(s',a') - Q(s,a)) f_i(s,a)$$

Use a learning rate of .1 for the execises below.

- In the first trial, the agent randomly follows the following action sequence (starting from S):
  ⟨LEFT, RIGHT, LEFT, LEFT, LEFT, LEFT⟩
  What are the weights and estimated Q-values after this trial?

- In the second trial, the agent randomly follows the following action sequence (starting from S):
  ⟨RIGHT, LEFT⟩
  What are the weights and estimated Q-values after this trial?

- Is it possible to find the optimal policy using this set of features? Why or why not?

2. Consider the problem of using a genetic algorithm to search for an optimal policy for the MDP described above.
   - How could you encode policies for this problem as bit-strings? Be specific. How many bits does your encoding take?
   - Describe an appropriate fitness function for this problem.