Readme

This project was completed by Neem, John, Wynter, Irvin and Sanskriti. We were able to complete this project in class time itself where we coded the program together. I think we equally contributed to this project.

For question 1, I think we were able to solve that pretty quickly. We started by coding computeQValues from action method which was pretty straight forward because of the formula we had discussed in class before. Then we coded the getActionFromValues function which relied on us to use the computeQValues function first. I think initially we were talking about getActionFromValues first but we quickly realized that it required the other function to be completed first. Then we started coding the initializer. We ran into a few problems there where our initialization was giving logic error. We had initially initialized the max value to 0 and computed but that gave us an error because if all the number were negative one of our if statement broke. Then we initialized the max value to -inf to start with but ran into another logic error which we could not figure out what it was. Fortunately, an idea came up where instead of initializing the max value we just added all the values in a list and took the max from the list. This solved the problem, and we passed the tests correctly.

We found that when we hold discount constant and decrease the noise, we will be passing the bridge and when we increase the noise, we are more likely to not even go the first step correctly. This makes sense though because noise is the chance on going some other direction when an intended direction is chosen. So, a lower noise means that we will go towards the destination most of the time and higher means the opposite. With discount we found out that if the discount is set to 0, our policy was always pointing N. With lower discount values (< 0.5) we were not passing the bridge, but the bridge values were becoming less negative compared to when discount value was high.

For noise we found out that if noise is between 0 and 0.15, we would cross the bridge and with discount we could not find a range and our guess is that with by just changing the discount we cannot pass the bridge.

For 3, we looked at the words to decide if we want to increase or decrease the values of discount and noise. So, if we prefer the close exit, the discount would be low meaning we would be a bit short sighted. If we prefer the distant exit, the discount would be high meaning the later positions would have more value, or we would be a bit far sighted. Similarly, if we want to risk the cliff we would decrease the noise and if we avoid the cliff we would slightly increase the noise. For 3d since we want to avoid both exists and cliff we just increase the living reward by a high positive number.