Part 2: Experimentation
By Tan Xian Jun
NetID: xtan11

1a) For every single episode, the total reward achieved at the end is 0. This is because the program will take the greedy path every single time without trying out other routes, and does not learn. As the best course of action for each state is not known to the agent, the reward at the end of each episode is 0.

1b) After changing epsilon, there is a change in the total reward received at the end of each episode. The rewards are greater than 0. This is because a degree of randomness is added when choosing the next best action. Instead of choosing the action with the best associated utility Q every single time an action is to be taken, sometimes it would take a random action at a 10% chance to do so. This will allow new actions to be taken instead of the same ones every single time, allowing the program to try out and learn about the Q values of other actions and therefore produce better results.

1c) After trying a larger epsilon of 0.5, the reward at the end of each episode is now negative. This is due to the epsilon being too large. With an epsilon of 0.5, half the time the actions chosen will be random. Instead of the epsilon providing a chance for the agent to learn better actions, it will instead choose random actions instead of the optimal one half the time.

2a) The resulting reward at the end of each episode is negative, showing poor performance. This is due to the agent not being aware of the thief and not being able to assess its state accurately, as the state does not reflect the presence of the thief. Thus the actions chosen and the resulting policy is suboptimal.

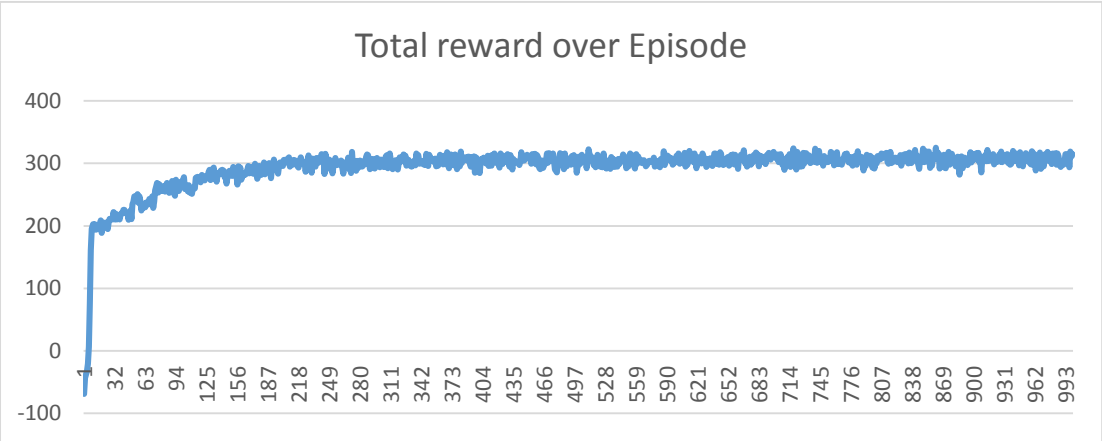
2b) The resulting reward is now positive, over 250 on average. This shows that with knowledge of the thief, the state the agent is in is more accurately depicted in, and better actions to deal with each particular state can be learnt and added to the policy.

2c) Raising the epsilon from 0.05 to 0.1 drops the average total reward to around 210. Raising it to 0.2 drops the average total reward even lower to around 100.
Raising the learning rate to from 0.1 to 0.2 has little effect, and with similar results when it is raised to 0.3. Raising it to 0.6 drops the average reward to 40, with an episode with a failing reward of -6.0.
Resetting the epsilon to 0.05 and the learning rate to 0.1, raising the learning rate has the similar result of lowering the reward earned. Lowering the learning rate to 0.03 however has improved the reward earned. Setting it to zero however makes all the rewards at the end negative.
Setting the learning rate to 0.03 and the epsilon to 0.02 improves the reward even further, resulting in an average of 310.

In conclusion, the best epsilon should be 0.02. Any lower or higher and the average reward will fall. The best learning rate similarly should be around 0.03.

3) The plotted graph can be seen below. I noticed that the rewards gain jumped quickly and then tapered off, like a logarithmic function. This showed that the number of episodes does not matter much after the first 300 episodes as the improvements in rewards earned showed diminishing

returns.



The graph below shows the average total reward of 10 simulation over episode. The result is a smooth logarithmic curve. This means that increasing the number of episodes is pointless after a certain number of episodes, as the total rewards achieved will not improve by much. This can be explained, as after the initial gains in reward by correcting the Q values of each state and action are done, there is not much left to improve on.