Q-learning on WWU CS floor

The goal of this project was to apply the unsupervised reinforcement learning technique known as q-learning to train an agent to go from any room in the WWU cs floor to the room labeled 441 on the floor map. The first step in doing so was creating a state space which was modeled after a 52 x 16 grid representing the floor and all of the rooms. States were either rooms or grid squares that made it possible to go from room to room (parts of the hallway). Every state label (a number) corresponds to a location on the grid. After doing so the R-Table was created which showed with state-action pairs were possible and indicated the rewards of going from state to state. In our case the goal was for an agent to end up in a certain room so all actions the led to that room had a reward of 100. The next step was to populate the q-table so that the agent could learn which actions would lead it to the desired state. This was done by using all of the targets which were rooms on the floor and then taking random actions and seeing what rewards that all of those actions led to. For each target an episode would end when an action was taken and it led us to our desired state.

There was an issue with taking random actions at every step because some rooms are far away from the desired state so by taking random actions at all steps the goal the state was not reached after a long period of time. To deal with this i made it so actions were not random at every step. I included an array that would keep track of the number of times a state was visited and a parameter that would decrease the likelihood that a random action would be taken and increase the chance of visiting a state that has not been visited often. By doing so the goal state was reached in a more reasonable amount of time and many states were still explored.

The algorithm used to populate the q-table is the following :

Q(state, action) = R(state, action) + Gamma * Max[Q(next state, all actions)]

I experimented with changing Gamma to range between 0 and 1, a higher number putting more emphasis on future rewards, but this did not end up changing the results very much. The results of performing the learning trials with each target ended up as expected. Using the q-table to find a path between each target and the goal state showed that the paths taken were such that the least amount of actions would be taken to reach the goal state. This can be easily seen, albeit not proven by the visual path display.

References :

Watkins and Dayan, C.J.C.H., (1992), 'Q-learning.Machine Learning'
http://www.mnemstudio.org/path-finding-q-learning-tutorial.htm