1)

a) The final reward is always 0 because the robot remains in the starting location. With a purely greedy policy, the robot "learned" that it should avoid the slippery tiles at all costs (and therefore makes no deliveries). With some randomness the robot would have learned that the reward for crossing the tiles outweighs the risk.

b) The rewards are negative at first, but climb steadily until reaching consistently high scores (100+) by episode 25. The robot mostly gets scores between 100 and 200, with some outliers off both ends. This indicates that the robot slowly learned over time which options were the best choices and was able to make deliveries as it was expected to.

c) With epsilon at 0.5, the robot became too erratic to function correctly. The robot consistently got negative scores, with a rare positive score peaking at 5. The robot did not appear to be learning, and when PolicySimulator was run it decided to run into a wall repeatedly.

2)

a) In the world with the thief (and no knowledge of its location), the robot runs very poorly. The robot consistently scored in the [-20, -10] range. With no knowledge of the thief, the entire center column becomes a dangerous (but mandatory) crossing for the robot. The robot's decided policy is to simply remain in the starting corner. Without knowledge of where the thief is, the expected value of the reward for visiting any tile the thief could visit becomes too low for the robot to choose to cross it.

b) With knowledge of the thief's location, the robot performs immensely better. Starting in the negatives, the robot achieves consistently positive scores by episode 10. After 100 episodes, the robot consistently receives a reward in the upper 200's. By knowing the thief's location, the robot can much more accurately determine the risk of visiting a thief tile. The robot can get past when the thief is not nearby and wait when he is.

c)

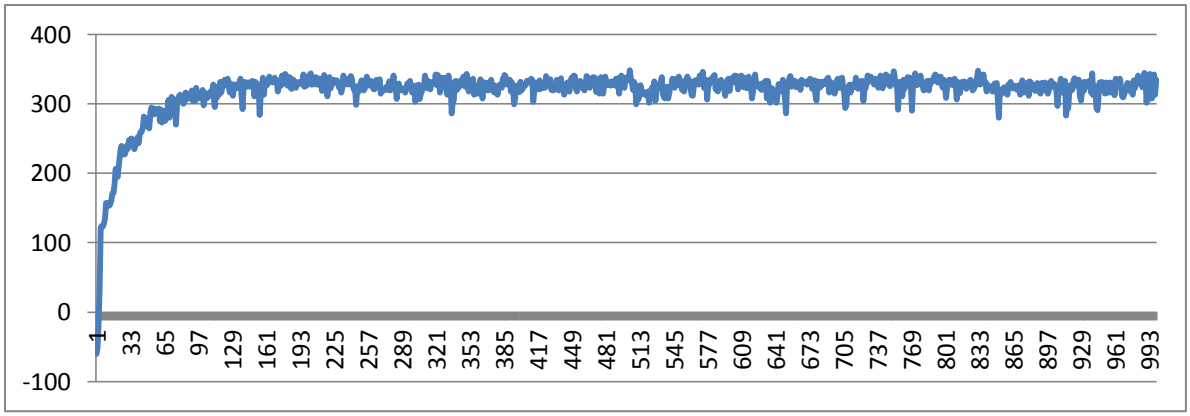
Epsilon	Learning Rate	Average Score (last 10 episodes)
0.05	0.1	290
0.1	0.1	215.15
0.15	0.1	124.75
0.2	0.1	84.15
0.25	0.1	33
0.3	0.1	-4.35
0.01	0.1	336.75
0.02	0.1	317.25
0.03	0.1	306

0.005	0.1	336.15
0.01	0.05	320
0.01	0.1	336.75
0.01	0.15	336.35
0.01	0.2	326.9

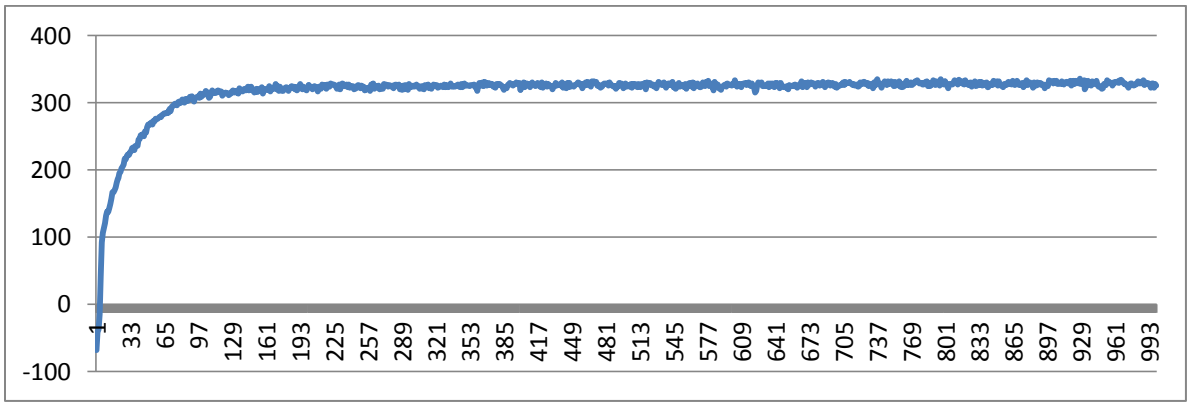
In order to determine the robot's ap[proximate effectiveness, I averaged the robot's rewards for its final 10 episodes. Using this value, I concluded that epsilon = .01 and learning rate = .1 are the most effective values.

3)

One run:



Ten runs (averaged):



The bot learns quickly at first, rapidly going from negative scores to 300+. After ~100 episodes, the robot's learning appears to plateau, as its score increases only incrementally for the remaining 900 episodes