Reference code: https://github.com/WilliamLambertCN/CS188-Homework/tree/master/PJ4_tracking

Q1)

In the first test, ghost movements are completely randomized due to it's using a randomagent. getPositionDistribution function returns the probabilities of the next possible ghost location distributions without considering the current location of the ghost. This leads to having equal probability for getting into each legitimate location. In the second test, ghost movements are to the south direction due to it's using gosouthagent. When the time increases, the more possible for the ghost to get into valid south locations.

Q2)

When an observation occurs, the beliefs are updated by multiplying them with the new probabilities. In the second test, the agent observation doesn't change because the agent remains in its location. The ghost position cannot be determined because corners can't be observed by pacman. In the third test, the agent has the capability to move. This way, it obtains observations from the environment and updates the beliefs. Using beliefs, the ghost location can be found by filtering out the possibilities for the ghost.

Q3)

In the beginning the particles are initialized uniformly and this leads to having many possible locations. When time increases, observations of particles gets updated. Weights are used to collect the particles. Using particles, the possibility of the locations are calculated. At a particular time, the weights of the particular equals to 0 which results in making the agent in a position that it can't infer possibilities. At that particular time, the particles are re-initialized uniformly. Increasing the number of particles can't be a solution due to the fact that elapsed time isn't taken into account.

Q4)

Since collected data is used within the exact inference, it achieved better accuracy values as opposed to the approximate inference. In approximate inference, probabilities are determined by using the samples. The time complexity of approximate inference is better than exact inference because it can easily calculate by using the probabilities. When the elapsed time is also taken into consideration 5000 particles makes sense because ghosts can be found with a good amount of accuracy and less time.

Q5)

In the DBN question, the implementation is very similar to the approximate inference part. One of the main differences is the new distributions are calculated regarding the prior ghost particles and ghosts are dependent among themselves. To satisfy a uniform initialization, the matrixes for each ghost are mixed. Each particle is a ghost beliefs list as ghosts are employed to represent each one. getPositionDistribution function is called to update particles then a sample is obtained.

The particles are updated by calling the getObservationProb function for the observation of each ghost. Weights are updated using these values for all of the particles.