Project 2: Multi-Agent Pacman

CS6300: Artificial Intelligence, Spring 2018 **Jake Pitkin**
University of Utah

1 Reflex Agent

1. **What were the features you used for your evaluation function?**

   My reflex agent uses both food position and ghost position in its evaluation function. If a action would place Pacman in the same position as a ghost, that action is not taken. If an action will take Pacman to food and there is no ghost, that action is always taken. Finally if there is no food directly next to Pacman, he moves towards the food with the smallest Manhattan distance. To maximize completion time and to avoid ghosts, my agent never selects the *Stop* action. The following scores are returned to achieve this behavior:

Situation	**Score**
Action is 'Stop'.	0
Action leads directly to a ghost.	0
Action leads directly to a food and not a ghost.	2
No neighboring food, tend towards the closest food.	1 / min(food manhattan distances)

Table 1: Evaluation function return values.

Note: situation 3 returns a score of 2 so it always beats situation 4 when the best manhattan distance is 1 (after the action is performed).

2 Minimax

1. **When Pacman believes that his death is unavoidable, he will try to end the game as soon as possible because of the constant penalty for living. Give an explanation as to why the Pacman rushes to the closest ghost in this case ?**

   Pacman will attempt to maximize the minimum values presented to him. When all the available options are death, he will pick the choice that maximizes his score (dying as quickly as possible).

3 Expectimax

1. **You should find that your ExpectimaxAgent wins about half the time, while your AlphaBetaAgent always loses. Explain why the behavior here differs from the minimax case.**

   In the minimax case with alpha beta pruning, we assume that the ghosts make the optimal decisions at every move. In this case, they don't act optimally but rather action randomly with equal probability across all their actions.

4 Evaluation Function

1. **What features did you use for your new evaluation function?**

   I used a linear combination of the following features for my evaluation function:

Feature	**Weight**
The game score.	1
Reciprocal of the Manhattan distance of the closest food.	1
Reciprocal of the sum of the Manhattan distances of all remaining capsules.	1
The number of remaining capsules	-21

Table 2: New evaluation function.

   I found the game score to be very expressive of the quality of a game state and is the largest factor in my evaluation function. It pushes Pacman to eat ghosts and to not die. To encourage Pacman to move towards food, he is rewarded for being close to food.

   I wanted to try to maximize my average score. I found that killing the ghost twice is the best way to do this. To encourage Pacman to do this I put a reward for being closer to capsules and a penalty for capsules remaining on the map. This gave me an average score of 1233.9 with the test seed.

5 Self Analysis

1. **What was the hardest part of the assignment for you?**

   The trickiest part of this assignment for me was limiting the depth of the minimax function.

2. **What was the easiest part of the assignment for you?**

   The rest of the assignment was straight forward given good pseudo-code from the lecture slides and being accustomed to the framework's API from completing project 1.

3. **What problem(s) helped further your understanding of the course material?**

   I think the material was covered.

4. **Did you feel any problems were tedious and not helpful to your understanding of the material?**

   I didn't, this assignment went smoothly.

5. **What other feedback do you have about this homework?**

   No real feedback, the instructions are clear and the unit tests are helpful.