

### **Overview**

The objective of this assignment was to do questions 3-5, filling in the AlphaBetaAgent function, ExpectimaxAgent, and betterEvaluationFunction so that they would work in accordance to the provided auto-grader.

### **AlphaBetaAgent**

This part was similar to MiniMaxAgent; The goal of this was to create a multiagent Minimax function but with alpha beta pruning.

It is MiniMaxAgent, but checks and updates alpha and beta values so as to see whether or not one branch needed to be cut. In ghostMin, it is checked whether the minimum value is less than alpha, and if so, returns that value. If not, it updates beta. PacmanMax is similar, but it is the opposite. The maximum value is compared; if it is greater than beta, it returns the max value. If not, alpha is updated.

This implementation passes all the autograder's tests, even though the game on smallClassic is lost, with a score of 84 after 2 seconds.

### **Expectimax**

This part was similar to MiniMaxAgent; The goal of this was to create a multiagent Expectimax function.

The same as MinimaxAgent, with only ghostMin being different. Instead of taking the minimum value, it takes the average of all the values, hence why the function is called ghostAvg instead.

This implementation passes all the autograder's tests, even though the game on smallClassic is lost, with a score of 84 after 2 seconds. This result is identical to AlphaBetaAgent.

## **betterEvaluationFunction**

The goal of this part was to fill in the betterEvaluationFunction. The premise of the evaluation is the score evaluation of the current game state, the distance of the closest food and the closest ghost, and the number of pellets left on the board. Some adjustments to the weight of each component was done. Mostly copied from my own evaluation function in ReflexAgent, the difference being the score evaluation of the current game state.

This implementation completes the autograder with a record of 10/10 wins, with an average of about 994 points. Given the average is greater than 500 but less than 1000, the autograder gives 5/6 points.

## **Thoughts**

Both AlphaBetaAgent and Expectimax were the same as the Minimax function, just with some minor adjustments. So, these two functions were easy, as the base of it was already completed in part 1. Similarly, the betterEvaluationFunction was nearly identical to ReflexAgent's evaluation function. However, more work could be done on it to push the average that six extra points it needs to score 6/6.