A PROJECT REPORT ON

MULTI-AGENT PAC-MAN

Project 02: Artificial Intelligence (CSE 537)

Instructor: Professor I.V. Ramakrishnan

BY

Nisha Gandhi (111496495)

Department of Computer Science

Stony Brook University

2017-2018

1. Reflex Agent (Question 1)

Logic applied: The reflex agent function returns a score based on the Pacman's distance from the ghost and its distance from the food. Higher the food distance should return a lesser sore and consequently lesser the distance from the food will return a higher score. The distance from ghost returns a negative value. Therefore lesser the distance from the ghost should return a higher negative value and a much bigger distance will return a lesser negative value. Hence, we compute the reciprocal of the distance of the ghost from the Pacman. These two distances together add up to return the score of the reflex agent along with the initial value of score that is set as ***successorGameState.getscore()***.

Return values: The score computed.

2. Minimax Agent (Question 2)

Logic applied: Minimax agent has been implemented using a recursive function, taking the current depth and the current game state as the parameters. Each layer in the depth (move) consists of one pacman and multiple ghosts (accessed by ***getNumAgents*** function). For each state, its legal actions are generated and the function is called recursively with its successor state. The maximum or the minimum value is chosen between the value returned by the recursive function and the earlier maximum/minimum value (respectively for pacman and the ghosts , pacman – maximum, ghosts – minimum). The leaf nodes(the end states) of the minimax tree returns the score (accessed by ***evalutionFunction***).

Return values: Returns the minimax action the pacman should take in order for it to win.

3. Alpha-Beta Pruning (Question 3)

Logic applied: Alpha-beta pruning is implemented with two helper functions: one (***alpha_value***) for the pacman, and the second (***beta_value***) for the ghosts. These helper functions generate the legal actions and successors for that particular agent. It then recursively calls the ***alpha_value*** and ***beta_value*** for the pacman and the ghosts respectively. The leaf nodes return the evaluation score. If returned to a min node, beta value gets updated if the value returned is lesser, and if returned to a max node, alpha value gets updated if the value returned is greater. The moment alpha rises above beta, we stop and prune the particular sub-tree. This enables faster and efficient traversal of the minimax tree. It also helps in avoiding the expansion of nodes which do not make any difference.

Return values: Returns the minimax action using alpha-beta pruning for the pacman in order for it to win.

4. Expectimax Agent (Question 4)

Logic applied: Expectimax agent does not assume that the adversary will always make the optimal decision. These agents take into account the fact that the adversary might make sub-obtimal decisions at times. Hence instead of choosing the minimum value, it returns the probabilistic behavior (the average) of the ghost actions.

Return values: Returns the action for the pacman to take in order for it to win taking into consideration the fact that the adversary is not perfect.