## Part 2: (Minimax explorative)

1. In the game of Pacman, minimax will want to minimize the loss for Pacman on every move the ghosts of the game make and will maximize the score of Pacman if possible. Pacman has suicidal tendencies when death is imminent because the maximizing play in this case is to die as quickly as possible or else the ghosts on the board will get more score (bad for Pacman) and Pacman will lose score as the game that has already been decided goes on. This is not the goal of Minimax with Pacman; thus, Pacman will choose to kill itself as soon as possible in order to maximize its score and minimize the score of the ghosts.

## Part 3: (Alpha-beta Pruning)

1. a) Alpha-beta has a time complexity of $b^{d/2}$. Thus, in the same amount of time, alpha-beta would be able to reach a depth of 2d.

   b) In the worst case, alpha-beta will be as bad as minimax, thus only being able to reach a depth of d.

2. **FALSE,** because we will always receive the value of the best move in both minimax and alpha-beta. In alpha-beta's case, we just prune the rest of the unnecessary part of the tree which minimax would have traversed but wouldn't have changed the best move value. We just return the best move faster with alpha-beta pruning while in minimax, minimax will traverse the rest of the tree but find that the value that alpha-beta would have got is indeed the best move.

Part 4: (Expectimax)

1. a) **TRUE.** Since the nodes under the max nodes are now based on weighted averages instead of being min nodes, we get that the value of each weighted average node can only be the same or greater than the min node value. Thus, we will get cases where $v_e$ will be greater than or equal to $v_m$ when we are maximizing.

   b) **TRUE.** By optimal minimax definition, we will always achieve at least a payoff of $v_m$ because the minimax will treat the chance nodes as regular min nodes. By a), we know that the chance node's smallest payoff will be at least $v_m$.

   c) **FALSE**. By a) and b), we know that the payoff is at least $v_m$. We also know that by definition $v_e$ is the expected value of the node. Since there is probabilistic behaviour involved, we can end up with results that are lower than $v_e$ (which is not $v_e$) but NOT lower than $v_m$. Thus, not guaranteeing us getting $v_e$.