Explorative Question 1:

The heuristic value for a game state is fundamentally different than A* search.

In A* search, everything is dependent on 1 agent and therefore, the heuristic for this search guides the agent to the goal state. The heuristic is better, if it is closer to the actual cost of going from a node to a goal state.

In game theory search, there are more than 1 agent (usually 2 player, zero-sum) and their respective actions and their values are both dependent on the first agent and the second agent. The heuristic for this type of search is better, if it can accurately predict the state of the game, where we reach a terminal state. In another words, the good heuristic estimates the result of the game accurately, if a certain actions happen sequentially.

Explorative Question 2:

1.

The reason for this behaviour is that the game tree is generated at the beginning of the game and all states are already evaluated and based on those states, the Pacman's move is pre-determined. The evaluation function for this part uses the getScore() method, which accounts for the time as well (each frame through time reduces the score). If all the actions of pacman yield to a death situation, it will tend to choose the one with highest score when it dies, meaning that it wants to minimize the time passage(minimizing the time penalty), and so, it wants to die earlier. As a result, it will have a suicidal behaviour.

2.
(a) Will the search result in the same strategy in cases (i) and (ii)? Not Same
(b) Will the search result in the same strategy in cases (i) and (iii)? Not Same
(c) Will the search result in the same strategy in cases (ii) and (iii)? Not Same

Explorative Question 3:

1.(a)
Minimax runs in b^d.
For best case of alpha-beta pruning: we explore b nodes in the first layer, 1 node in the second layer, b nodes in the 3rd layer and so on. As a result, in each 2 layers, we explore b + 1 nodes, which is O(b). Since we explored b nodes in 2 layers, we are effectively exploring $\frac{b}{2}$ nodes in each layer. → We can search up to depth **2d** in the same amount of time.

1.(b)
Depth of d → the same depth as without alpha-beta pruning.

2.
False