

### Assignment 3: Written Answers

SeungDoo Yang  
1002421140  
CSC384

#### Question 1.

1. The game state is dynamic. The game state heuristic is “randomly” determined due to the incomplete information of the game. The game state heuristic also needs to consider its competitors’ play. For example, in case of Pacman, the game state heuristic depends on where its competitors (the ghosts), which are dynamically moving, are.

In normal search, we are only concerned with the location of the goal state (how close am I to the goal state?) in the tree of states. Therefore, in A\* search, heuristic value is mostly deterministic.

In summary, the game state heuristic focuses on short term goals that results in the state with better chance of winning. A\* search heuristic focuses on long term goals and efficiently guide towards the goal state.

For the heuristic to be good in either situation, it should satisfy the required/desired properties of both states. To satisfy A\* search property, the heuristic should be admissible or monotonic. A heuristic like this would give optimal estimates at each state. To become a desired heuristic in game state, it should be giving promising estimates after considering various future possibilities. It would consider short and long-term goals and efficiently guide towards the winning state.

#### Question 2.

1. Pacman would choose to make a suicidal move when there is no way of getting out of end/loss game state (for example, Pacman is surrounded by the ghosts in all directions) and there is no food that it can reach to. Then Pacman would choose to make a suicidal move in order to maximize the minimax score, because if the game is loss and there is no way of increasing the score anymore, it would choose to end the game earlier before the score decays further with time.
2. A) Not Same  
B) Not Same  
C) Not Same

#### Question 3.

1. A)  $2^d$   
B)  $d$ , in worst case scenario for alpha-beta pruning, it would not prune any node in the search tree which would be the same thing as just applying minimax algorithm on the tree.
2. False