CS-661 AI Assignment 5

1. To what does greedy search degenerate when the heuristic function is the accumulated path cost so far $h(n) = g(n)$? What if we take $h(n) = -g(n)$?

2. Invent heuristic functions for the following problems. Try to do so by coming up with a relaxed problem. Show that the heuristic is *optimistic* (i.e. never overestimates the cost).

- Solving an algebraic equation. In this problem assume we have an algebraic *engine* which can properly manipulate equations, and we wish to cause it to isolate a given variable.

- Traversing a maze. Here an agent must travel through the maze from an initial position to a goal position.

- The knapsack problem. In this NP-complete problem, one is given a collection of items of known weight, and a knapsack which can hold only a given maximal weight. The agent must fill the knapsack with the most weight possible without surpassing the knapsack's capabilities.

3. The following is a possible heuristic function for the **X** player in TicTacToe.

For every row, column and diagonal with only **X**s add the number of **X**s in that row, column or diagonal to the heuristic function.
For every row, column and diagonal with only **O**s add the number of **O**s in that row, column or diagonal to the heuristic function.

Code (in a language of your choice) an agent which performs greedy search using this heuristic. This entails identifying all legal moves, evaluating the heuristic for each of these, and selecting the best move. Does this agent play well? Should the agent look further than one (half) move ahead?