Heuristic Search

1) Solve Problem **??** for A* with a) the Manhattan distance heuristic and b) the straight line distance heuristic. The Manhattan distance heuristic between two grid cells (x_1, y_1) and (x_2, y_2) is $|x_1 - x_2| + |y_1 - y_2|$ (the length of a shortest path between the two cells, assuming that there are no obstacles on the grid). For instance, the Manhattan distance between A1 and E3 is $|1-5|+|1-3| = 6$.

2) We are given a sequence of integers and want to sort them in ascending order. The only operation available to us is to reverse the order of the elements in some prefix of the sequence. For instance, by reversing the first three elements of (1 2 3 4), we get (3 2 1 4). This problem is also known as the "pancake flipping" problem. We model this problem as a search problem, where each state corresponds to a different ordering of the elements in the sequence. Given an initial sequence (2 4 1 3), in which order does A* expand the states, using the breakpoint heuristic described below? Assume that ties are broken toward states with larger g-values, and, if there are still ties, they are broken in lexicographic order. That is, (2 1 4 3) is preferred to (2 4 1 3).

Breakpoint heuristic: A breakpoint exists between two consecutive integers if their difference is more than one. Additionally, a breakpoint exists after the last integer in the sequence if it is not the largest integer in the sequence. For instance, in (2 1 4 3), there are two breakpoints: one between 1 and 4 (since their difference is more than 1), the other after 3 (since it is at the end and of the sequence and is not the largest integer in the sequence). The breakpoint heuristic is the number of breakpoints in a given sequence. (Bonus question: Is this heuristic a) admissible and b) consistent? Why?)

3) Does A* always terminate if a finite-cost path exists? Why?

4) Given two consistent heuristics h_1 and h_2, we compute a new heuristic h_3 by taking the maximum of h_1 and h_2. That is, $h_3(s) = max(h_1(s), h_2(s))$. Is h_3 consistent? Why?

5) In the arrow puzzle, we have a series of arrows pointing up or down, and we are trying to make all the arrows point up with a minimum number of action executions. The only action available to us is to chose a pair of adjacent arrows and flip both of their directions. Using problem relaxation, come up with a good heuristic for this problem.

6) Explain why heuristics obtained via problem relaxation are not only admissible but also consistent.

7) What are the advantages and disadvantages of a) uniform-cost search and b) pure heuristic search over A* search?

8) Give a simple example that shows that A* with inconsistent but admissible heuristic values is not guaranteed to find a cost-minimal path if it expands every state at most once.

9) Solve Problem **??** for Iterative Deepening search.

10) Develop an iterative-deepening variant of A* (call Iterative Deepening A*), that is, a version of A* that finds cost-minimal paths for consistent heuristics and uses a series of depth-first searches to keep its memory consumption small. (Hint: If you are stuck, look at the lecture slides on heuristic search.) Solve Problem **??** for this version of A* with the (consistent) Manhattan distance heuristic.

11) (Thanks to Ariel Felner.) Given two admissible heuristics h_1 and h_2, we compute a new heuristic h_3 by taking the maximum of h_1 and h_2 (that is, $h_3(s) = max(h_1(s), h_2(s))$); and another heuristic h_4 by taking the sum of h_1 and h_2 (that is, $h_4(s) = h_1(s) + h_2(s)$). Is h_3 admissible? Why? Is h_4 admissible? Why?

12) (Thanks to Ariel Felner.) Develop an admissible heuristic for the following variants of the 15-puzzle:

a) Moving tile x costs x instead of 1.

b) Moving a row of tiles left or right as well as moving a column of tiles up and down costs 1 regardless of the number of tiles that are moved. For example, if a row's initial configuration is (B,1,2,3), where B is the blank (that is, missing tile), then it costs 1 to change this configuration to any of the following configurations: (1,B,2,3) or to (1,2,B,3) or to (1,2,3,B).