Part 3:

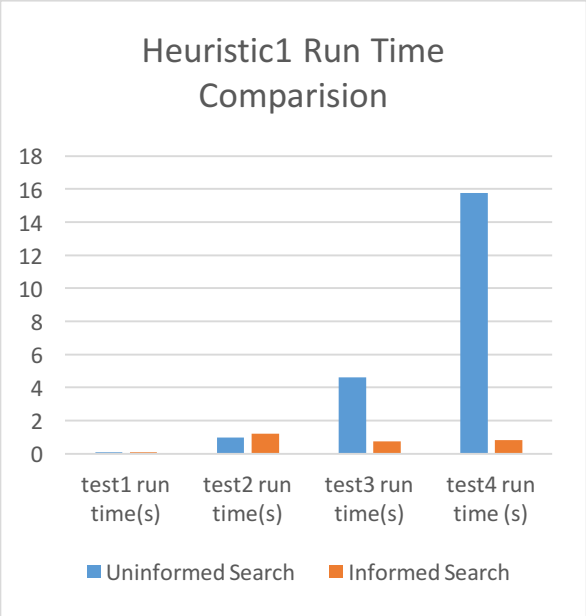
The basic algorithm of the BFS (uninformed search) for 8 puzzle problem is to visit every single state until it reaches the goal state. By doing this, first, you have to find all the possible movements (children) of the start state and remove it right after you store its movements into the list. Then, you need to go over each child and find its children again, and add them into your list only if they have not been visited before. After doing the same step over and over, you will finally reach the goal state and the process terminates.

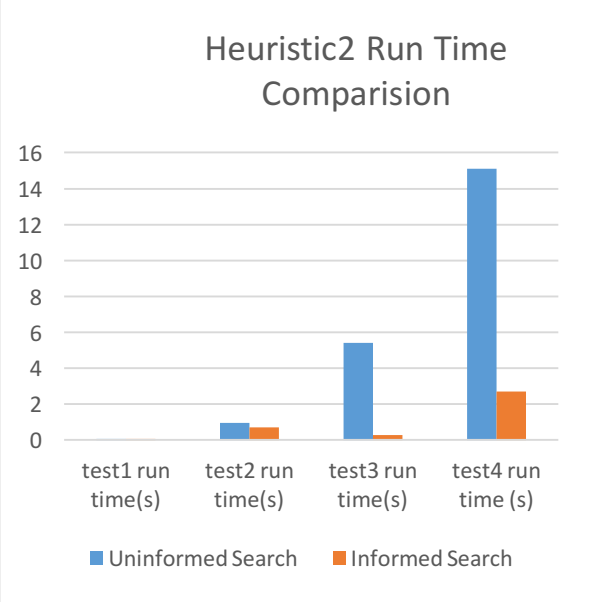
The A * algorithm (informed search) uses a completely different way which enhances the efficiency of the method. To accomplishing this, first, you have to make a heuristic method which calculates the sum of each tile is from its goal state. Then you add this number to the depth of the state, which satisfies the formula f(N) = g(N) + h(N). Instead of going over each state in uninformed search method, you select the child which has the minimum value and store the rest of children into list, then you find the children of the selected child and compare them to the previous children in the list. In this method, it provides the faster way which can approach the goal state.

In order of proving this, I tested four cases from the minimum difficulty one to the maximum difficulty one, which are:

test1 = State(values = [[4,1,3],[" ",2,5],[7,8,6]])
test2 = State(values = [[8,2,3],[" ",4,7],[1,6,5]])
test3 = State(values = [[6," ",1],[2,3,7],[5,4,8]])
test4 = State(values = [[7,8,6],[4,3," "],[2,5,1]]) .

I compared these 4 test cases for heuristic 1 and heuristic 2 respectively and collected the data as below:





The graphs obviously show the running time differences between uninformed search and informed search, after applying the A * algorithm by using heuristic functions, the search methods filter out most of the states which are far from the goal state. Therefore, the number of states that will be visited each time reduce dramatically and speed up the running time.