

Assignment 3

Total Points 40

In this problem you are to output the solution to the 8-puzzle problem, i.e., to find the sequence of moves from some initial configuration/state of the puzzle to a final state. As for the moves consider moving the blank tile moving UP, DOWN, LEFT, RIGHT in that order. Each move has cost = 1.

Figure 1 below shows example initial and final states.

Initial State:

1	4	2
	5	3
6	7	8

Final State:

1	2	
5	4	3
6	7	8

Figure 1: an example initial and final state

1. [10 points] Show by hand the execution of A* search algorithm. Initially the frontier will have the initial state (name it state 0) with its f-value. Number the subsequent states and compute their f-values. In each step show: a. which path is selected, b. whether it is the optimal path, and c. the contents of the frontier at the completion of the step.

The heuristic function to be used: $h(n) = \text{number of misplaced tiles}$, where n is a state. For example, for the initial state, $h(\text{initial}) = 3$ (as tiles 2, 4 and 5 are not in the correct positions)

2. [30 points] In this problem you will implement the Depth first Branch and Bound algorithm.

a. [5 points] Write a pseudo-code of the “generic” Depth First Branch and Bound algorithm. Update the generic search algorithm presented in class for this.

b. [25 points] Write a code to implement the algorithm to solve the 8-puzzle problem.

Input format: To input the instance of the problem, assume the cell numbers are as shown in figure 2. Then the initial state in figure 1 can be input by the following sequence: 1 4 2 -1 5 3 6 7 8, where the i -th cell of the puzzle contains the i -th number in the sequence and -1 represents the blank tile. The final state of the example above can be input as: 1 2 -1 5 4 3 6 7 8.

cell#1	cell#2	cell#3
cell#4	cell#5	cell#6
cell#7	cell#8	cell#9

Figure2: Cell numbering

[see the next page for a sample input and output]

Sample input:

1 4 2 -1 5 3 6 7 8 //a sequence of 9 numbers representing the initial state.

1 2 -1 5 4 3 6 7 8 //a sequence of 9 numbers representing the final state.

Sample output:

PATH FOUND // if no path exists print: "NO PATH FOUND" and end output

Moves: RIGHT -> UP -> RIGHT

Path:

1 4 2

-1 5 3

6 7 8

--->

1 4 2

5 -1 3

6 7 8

--->

1 -1 2

5 4 3

6 7 8

--->

1 2 -1

5 4 3

6 7 8

Submission Instructions:

1. You will submit a pdf file containing the solution to problem 1 and 2(a). The pdf file should be named yourID_assign2.pdf (e.g., 172001001_assign2.pdf)

2. For the coding problem (2(b)) you can use any language of your choice (preferably python). You will submit the file containing code (filename should contain your id) and another file containing the output of the sample input given above and the output of **two more instances** of the problem (i.e., two additional initial and final state pairs).

3. In addition to having your ID in the filenames, make sure you write your name and ID on the first page of the pdf file as well as in the file containing the code.