

Programming Assignment 1 – AI I – Spring 2018

The purpose of this assignment is to implement the search algorithms: breadth first search (BFS), depth first search (DFS), iterative deepening search (IDE), and A* search.

The setting: we will use the 8puzzle as the problem that needs to be solved.

1. *Preparation*: formalize the 8puzzle problem as a search problem.
2. *Preparation for the implementation*: In this assignment, we will represent a **state** of this problem as a **string of 9 characters**.

For example, the string

12345678* corresponds to the configuration

1	2	3
4	5	6
7	8	

*13456782 corresponds to the configuration

	1	3
4	5	6
7	8	2

134*25678 corresponds to the configuration

1	3	4
	2	5
6	7	8

3. *Implementation*: Write a program in JAVA, C, C++, Python, ... to solve the 8-puzzle problem using the above search algorithms. However, you should make sure that the TA can compile and test your program on the computer of the department (Room 128).

You can adapt the code available from the website of the textbook (so you do not need to re-implement the algorithms). However, your code **must** use the state representation as defined in Item 2. You can get the latest code for the book in different languages at <https://github.com/aimacode>.

For the A* algorithm, you need to implement two heuristics: the number of misplaced tiles and the total Manhattan distance.

Your program **must**

- **display** on the screen the states that **will be added to the fringe and the state selected for expansion** in each iteration. Each configuration should be displayed as a string (as shown above). Different configurations should be displayed on different lines;
 - **report** the number of states in the **fringe** and **the number of states that have been expanded** at the end of the search;
 - **be** able to run on a terminal via command line; and
 - **accept** two parameters, the initial state and the goal state.
4. *Submission*: please upload the following in a **single zip file** that contains the following:
- a. A write-up of your answer for the first item.
 - b. A README file for compiling/running your program.
 - c. A print out of the code that you use for computing the successor state function.
 - d. A print out of the code that you use for checking the goal state.

Name your submission using the following convention: "Last_Name_ID" (last name followed by your ID).

Grading (100 for full marks):

- 5 points – 1st Question
- 20 points for each search strategy (A* gets 40 points, 20 for each heuristic)

Points will be deducted for not following instruction!

Deadline: 11:59pm, 2/15/2018.