

CENG 462

Artificial Intelligence

Spring 2023-2024

Assignment 2

Regulations

1. The homework is due by **23:55 on April 7th, 2024**. Late submission is not allowed.
2. Submissions are to be made via ODTUClass, do not send your homework via e-mail.
3. Upload your solution as a single **Python** file named *yourStudentId_HW2.py*. **Submissions that violate the naming convention will incur a grade reduction of 10 points.**
4. Send an e-mail to **garipler@metu.edu.tr** if you need to get in contact.
5. **This is an individual homework, which means you have to answer the questions on your own. Any contrary case including but not limited to getting help from automated tools, sharing your answers with each other, extensive collaboration etc. will be considered as cheating and university regulations about cheating will be applied.**

Question

In this homework, you are expected to write a **Python** script implementing **A* Algorithm** to solve a sliding puzzle where there are n white tiles (W) and n black tiles (B) and an empty cell (E). Below is an example sliding block puzzle with $n=3$:

W	W	W	B	B	B	E
---	---	---	---	---	---	---

Figure1. *Sliding Puzzle Example*

The goal of the puzzle is to have all of the white and black tiles from a given initial configuration to a given final configuration by using the following moves:

- A tile may move into an adjacent empty cell with unit (1) cost.
- A tile may hop over at most two other tiles into an empty cell with a cost equal to the number of tiles hopped over.

You are expected to design and employ an admissible heuristic function. Further, write a short comment to the very top of your script, specifying your heuristic function and why it is admissible.

Input

Your script is expected to take three inputs from the terminal:

n (i.e. number of white tiles or black tiles)

initial configuration

final (goal) configuration

Note that each configuration corresponds to a string of length $2n+1$ (n white tiles, n black tiles, 1 empty cell).

Sample Input

3

WWEWBBB

WWWBBBE

Output

The first line of the output of your script should contain a tuple of two integers k and m such that k is the length of the solution path found and m is its total cost. If no solution is found, k and m must be set to 0.

The second line of your output should contain the (ordered) list of states in the solution path. Note that the list starts with the (given) initial state, ends with the (given) final state, and contains $k + 1$ states in total.

Sample Output

(2,3)

[WWEWBBB, WWWEBBB, WWWBBBE]