# Theory of Computer Games (Fall 2022) Homework 1

NTU CSIE

Due: 2022/10/20 14:20

(NTU CSIE)

Theory of Computer Games (Fall 2022)

Due: 2022/10/20 14:20

▶ < ∃ >

1/13

3

### Outline







э

2/13



< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <



### New game - broken puzzle



(NTU CSIE)

Due: 2022/10/20 14:20

Image: A matrix

(3)

5/13

э



(a) Goal board

(b) Initial board

< □ > < □ > < □ > < □ > < □ >

3

- Implement the program to find the least number of slides so that the initial board is turned into goal board in 10 sec per task.
- Write a Report

### Homework Requirements

#### Input format

- n,m: number of row and column
- $1\sim 18$ : a tile that can be slid
- 0: empty
- -1: a nun-movable tile
- Goal board
- Example:

```
4 5 // n m

1 8 3 4 5 // tiles

6 7 2 -1 10

-1 12 -1 14 0

16 17 18 15 0

1 2 3 4 5 // goal board

6 7 8 -1 10

-1 12 -1 14 15

16 17 18 0 0
```

4 = + 4 = + 9 0 0

8/13

### Homework Requirements

### Output

output the least number of slides to solve the puzzles , and the sequence of moves.

- Output format
  - First line you should output the number of the least number of slides. Assume the ans is n.
  - And the following n lines, output the sequence of moves  $(a_i \ b_i)$ .
  - *a<sub>i</sub>*: index of tiles
  - b<sub>i</sub>: move direction
  - There is an example in homework files.
- Verifier
  - We will verify the sequence of moves is valid and whether it can solve the puzzle.

Usage: ./verifier -i [executable\_program] -d [input\_data]

-i [your program]

-d [test case]

Notice that the verifier only test your moves is valid, instead of the least number of slides.

(NTU CSIE)

Theory of Computer Games (Fall 2022)

Due: 2022/10/20 14:20

### Homework Requirements

- Test cases:
  - ► Cast 1, answer ≤ 10 steps: Implementation: Brute force
  - ► Cast 2, answer ≤ 20 steps: Implementation: A\*, IDA\* with heuristics
  - ► Cast 3, answer ≤ 30 steps: Implementation: Pattern database
  - Half of test cases are given
- Limitation:
  - Time limit is 10 sec
  - We will run your code on cise workstations
  - Upload the program to generate the pattern databases instead of uploading the pattern databases directly. And describe how to use your pattern databases in the report.
- There is a template about board and feel free to make any changes if you need.

(NTU CSIE)

Theory of Computer Games (Fall 2022) Due:

Due: 2022/10/20 14:20 10/13

▲ □ ▶ ▲ □ ▶ ▲ □ ▶

# Submission and Grading Policy

- Directory Hierarchy:
  - student\_id
    - ★ Makefile
    - ★ src // a folder contains all your codes
    - ★ report.pdf
- Compress your folder into a zip file and submit to https://www.csie.ntu.edu.tw/~tcg/2022/hw1.php
- Due to server limitation, the file size is restricted to 2 MB.
- Thread limit only one.
- The databases should be generated within 30 minutes, the file size is restricted to 1G.

- Your report should include but not limit to the following:
  - ▶ How to generate your pattern databases and how to use it.
  - How to compile your code.
  - What algorithms and heuristics you've implemented.
  - How the pattern database is designed
  - Experiment results and findings of your implementation.
- Your report should be named report.pdf.

- Test cases (90%)
  - 15 + 15 (hidden), 3 points per testcase.
  - If your sequence of moves can solve the puzzle but not the optimal solution, you can get partial points.
    - Your Point =  $\frac{s}{5} \times 0.8 \times 3$ , s: optimal solution, S: your solution
  - We will compile and run your code on csie workstations, therefore make sure your code can run on csie workstations.
- Report (10%)

A B M A B M