



# Assignment 1

## Sokoban

Parallel Programming  
2025/09/19



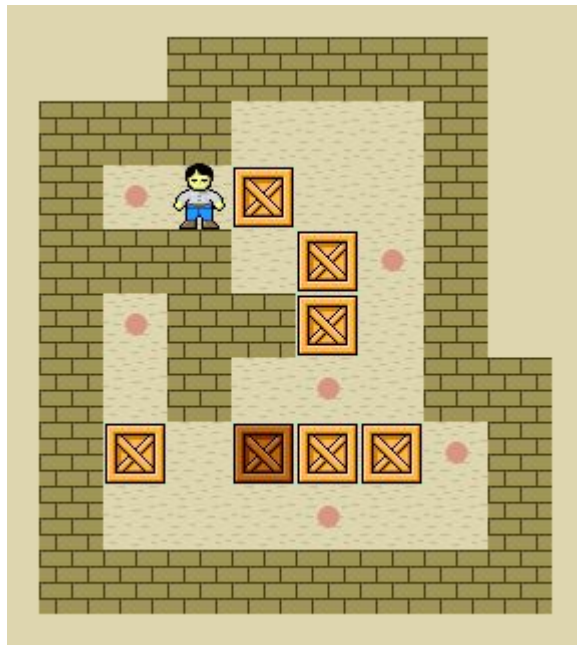


<https://hackmd.io/@pango0/hw1>

# Sokoban

- 倉庫番, sōko-ban

“a puzzle video game genre in which the player pushes crates or boxes around in a warehouse, trying to get them to storage locations.” - Wikipedia



# Goal

- Implement a solver for Sokoban called `hw1.cpp`
- Parallelize it with threads
  - Pthread or [std::thread](#)
  - OpenMP
- Only `c++` is allowed!

# Input

- o : The player stepping on a regular tile
- O : The player stepping on a target tile
- x : A box on a regular tile
- X : A box on a target tile
- : Nothing on a regular tile
- . : Nothing on a target tile
- # : Wall
- @ : A fragile tile (no box allowed)
- ! : The player stepping on a fragile tile

`samples/01.txt`

```
#####
```

```
#   XOX..#
```

```
#       #####
```

```
#####
```

# Output

- Print a valid sequence of actions that pushes all the boxes to the target tiles
- Directions: "W", "A", "S", "D"
- e.g. : DDAAASAAWDDDD for 01.txt

```
#####  
#  xOx..#  
#  #####  
#####  
current action sequence:  
solved: False
```

# Resources

- `/work/b11902044/pp25/hw1`
  - `example_solver.py`
  - `samples`
  - `play.py`
  - `validate.py`
  - `Makefile`

```
cp -r /work/b11902044/pp25/hw1 .
```

# Resources

- `/work/b11902044/pp25/hw1`
  - `example_solver.py`
  - `samples`
  - `play.py`
  - `validate.py`
  - `Makefile`

## Sample test cases

- `01.txt`
- `02.txt`
- `...`
- `25.txt`



# Resources

- /work/b11902044/pp25/hw1
  - example\_solver.py
  - samples
  - **play.py**
  - validate.py
  - Makefile

Play the game interactively

```
$ module load python
```

```
$ python3 /work/b11902044/pp25/hw1/play.py \
/work/b11902044/pp25/hw1/samples/01.txt
```

```
#####
```

```
#  xox..#
```

```
#  #####
```

```
#####
```

```
current action sequence:
```

```
solved: False
```

```
W/A/S/D to move; U to undo; Q to quit
```

# Resources

- `/work/b11902044/pp25/hw1` Check whether the output is correct
    - `example_solver.py`
    - `samples`
    - `play.py`
    - `validate.py`
    - `Makefile`
- ```
$ srun -A ACD114118 -c6 -o answer.txt ./hw1 01.txt
$ python3 /work/b11902044/pp25/hw1/validate.py 01.txt
answer.txt

Output: OK
```

# Resources

- /work/b11902044/pp25/hw1
  - example\_solver.py
  - samples
  - play.py
  - validate.py
  - **Makefile**

A simple way to compile your program. [Tutorial](#)

Put **Makefile** and **hw1.cpp** into the same directory

```
$ make
```

to compile your hw1.cpp as hw1

```
$ make clean
```

to remove the executable file

# Compile & Execute

Compile

```
$ g++ -std=c++17 -O3 -pthread -fopenmp hw1.cpp -o hw1
```

Execute

```
$ ./hw1 /path/to/testcase
```

```
$ srun -A ACD114118 -n1 -c6 ./hw1 /work/b11902044/pp25/hw1/samples/01.txt
```

# Judge

- Type `hw1-judge` in the directory of `hw1.cpp`
- Scoreboard:  
<http://140.112.187.55:7771/scoreboard/hw1/>

```
[ipc22t01@apollo31 hw1]$ hw1-judge
Looking for hw1.cc: OK
Looking for Makefile: Not Found
using fallback: /home/ipc22/snare/hw1/Makefile: OK
Running: /usr/bin/make -C /home/ipc22/ipc22t01/.judge.724388966 hw1
make: Entering directory '/home/ipc22/ipc22t01/.judge.724388966'
g++ -std=c++17 -O3 -pthread -fopenmp hw1.cc -o hw1
make: Leaving directory '/home/ipc22/ipc22t01/.judge.724388966'
02.txt 0.27 accepted
03.txt 0.27 accepted
01.txt 0.27 accepted
04.txt 0.37 accepted
05.txt 0.27 accepted
07.txt 0.32 accepted
06.txt 0.92 accepted
08.txt 0.42 accepted
09.txt 0.37 accepted
10.txt 0.42 accepted
11.txt 0.47 accepted
12.txt 0.32 accepted
13.txt 0.42 accepted
14.txt 0.32 accepted
15.txt 0.37 accepted
16.txt 0.17 accepted
18.txt 0.32 accepted
17.txt 0.42 accepted
19.txt 0.37 accepted
20.txt 0.32 accepted
21.txt 0.27 accepted
Removing temporary directory /home/ipc22/ipc22t01/.judge.724388966
Scoreboard: not updating {21 6.66} -x→ {21 7.58}
```

# Report

Answer the following questions, in either English or Traditional Chinese.

1. Briefly describe your implementation.
2. What are the difficulties encountered in this homework? How did you solve them?  
(You can discuss about hard-to-optimize hotspots, or synchronization problems)
3. What are the strengths and weaknesses of pthread and OpenMP?
4. (Optional) Any suggestions or feedback for the homework are welcome.

# Submission

- Due: **Thu, 2025/10/2 23:59**
- Zip the following files to a single archived file name **<studentID>.tar** and submit it to NTU Cool:
  - hw1.cpp
  - report.pdf
  - Makefile (optional)
- **Please note that the first character of your studentID must be lowercase.**

# Hints

Other test cases:

<http://sneezingtiger.com/sokoban/levels.html>

Algorithm: BFS, A-star, ...

check `/work/b11902044/pp25/hw1/example_solver.py`

OpenMP manual:

<https://www.openmp.org/wp-content/uploads/OpenMP-RefGuide-6.0-OMP60SC24-web.pdf>

Discuss with your friends!

Plagiarisms are PROHIBITED



# Q & A

Feel free to ask if you have any questions.