

COMP 472  
Fall 2020

Assignment 2  
The X-Puzzle

Victor Soledad  
#21297627

## The x-puzzle

Example Puzzle 2 x 4

-----

4	2	5	
7	1	6	3

-----

Solve with different algorithms:

1. Uniform Cost (UCS)  
**slowest but best results**
2. Greedy Best First (GBFS)  
**fastest but not best results**
3. Algorithm A\* (A\*)  
**best results but slower**

## Dual solutions

All the algorithms compared the states with one of two solutions

1	2	3	4
5	6	7	0

1	3	5	7
2	4	6	0

## Heuristics used for GBFS and A\*

$h_1$  = Sum of misplaced tiles

Simple heuristic that discriminates those solutions that are further from the final ideal solutions

$h_2$  = Manhattan move cost calculation with wrapping and corners

More complex heuristic that calculates the sum of all steps required per tile to reach its desired position.

## Scaling up

Python program is able to scale up and try to solve any sized puzzle

Example: 4x4 or 5x5

- **Very long computation times**
- **Fastest Algorithm GBFS**
- **Most never finished as the solution time is exponential**

=====

Puzzle 4 x 4

-----

```
| 9 7 8 1 |  
| 14 10 13 6 |  
| 12 15 5 |  
| 4 3 2 11 |
```

-----

Searching Algorithm: Greedy Best First (GBFS)

Solving for Heuristic: 2

Current Loop: 1000 | Time: 0.4617655277252197 secs

Stack Size: 2130 | Cost Range: [9 - 35]

Current Loop: 2000 | Time: 1.4152143001556396 secs

Stack Size: 4214 | Cost Range: [11 - 35]

Current Loop: 3000 | Time: 2.8374111652374268 secs

Stack Size: 6145 | Cost Range: [11 - 35]

=====

SOLUTION FOUND

Current Loop: 3542

Solution Time: 3.8656833171844482 secs

Solution Path: ['WRAP LEFT', 'LEFT', 'LEFT', 'DOWN', 'RIGHT', 'RIGHT', 'DIAGONAL UP-LEFT', 'DOWN', 'RIGHT', 'UP', 'UP', 'LEFT', 'LEFT', 'UP', 'LEFT',  
'WRAP LEFT', 'DOWN', 'WRAP RIGHT', 'DOWN', 'RIGHT', 'UP', 'LEFT', 'DOWN', 'WRAP LEFT', 'DOWN', 'WRAP RIGHT', 'RIGHT', 'UP', 'LEFT', 'DOWN', 'WRAP  
LEFT', 'UP', 'UP', 'UP', 'LEFT', 'LEFT', 'LEFT', 'WRAP LEFT', 'DOWN', 'DOWN', 'DOWN', 'DIAGONAL DOWN-RIGHT', 'RIGHT', 'DOWN', 'DOWN', 'LEFT', 'UP',  
'RIGHT', 'DOWN', 'RIGHT', 'DOWN', 'RIGHT', 'DIAGONAL UP-LEFT', 'LEFT', 'DOWN', 'RIGHT', 'RIGHT', 'UP', 'UP', 'WRAP RIGHT', 'RIGHT', 'RIGHT', 'RIGHT',  
'DOWN', 'DOWN', 'LEFT', 'LEFT', 'UP', 'UP', 'RIGHT', 'DOWN', 'LEFT', 'DOWN', 'RIGHT', 'UP', 'UP', 'RIGHT', 'DOWN', 'LEFT', 'DOWN', 'RIGHT', 'UP', 'UP',  
'LEFT', 'DOWN', 'RIGHT', 'DOWN', 'DIAGONAL UP-LEFT', 'RIGHT', 'DOWN']

-----

```
| 1 5 9 13 |  
| 2 6 10 14 |  
| 3 7 11 15 |  
| 4 8 12  |
```

-----

=====

=====

Puzzle 5 x 5

-----

```
| 17 13 15 9 10 |  
| 7 4 24 11 2 |  
| 19 23 3 20 5 |  
| 21 16 12 6 18 |  
| 8 14 1 22 |
```

-----

Searching Algorithm: Greedy Best First (GBFS)

Solving for Heuristic: 2

.  
. .  
. .  
. .  
. .  
. .  
. .  
. .  
. .

Current Loop: 111000 | Time: 7002.655311346054 secs

Stack Size: 202074 | Cost Range: [13 - 79]

Stoped

**Most 5x5 puzzles took too long with GBFS**

## **Analysis of 50 puzzles of size 2x4**

With each of the 3 algorithms and both heuristics

- **500 output files**  
50 puzzles x 5 algorithms x 2 output files
- **60 seconds time to live** for each algorithm  
After 60 seconds the solution was discarded
- By optimizing the algorithm data structures, it was possible to slightly lower execution time, regardless, **many of the UCS solutions reached the maximum execution time**
- **18 puzzles had no solution with UCS**
- **The fastest algorithm was GBFS with the 2<sup>nd</sup> heuristic**
- **A\* Took longer but had better results**

Analysis of Algorithm: ucs

=====

**Average Solution Length: 6.2**

Total Solution Length: 310

**Average Search Length: 9329.4**

Total Search Length: 466470

Average No Solution: 0.36

**Total No Solution: 18**

**Average Execution Time: 33.207913341522215**

**Total Execution Time: 1660.3956670761108**



Analysis of Algorithm: gbfs-h1

=====

**Average Solution Length: 19.5**

Total Solution Length: 975

**Average Search Length: 148.26**

Total Search Length: 7413

Average No Solution: 0.0

Total No Solution: 0

**Average Execution Time: 0.02231849193572998**

**Total Execution Time: 1.115924596786499**

Analysis of Algorithm: gbfs-h2

=====

**Average Solution Length: 19.98**

Total Solution Length: 999

**Average Search Length: 65.32**

Total Search Length: 3266

Average No Solution: 0.0

Total No Solution: 0

**Average Execution Time: 0.019668397903442384**

**Total Execution Time: 0.9834198951721191**

Analysis of Algorithm: astar-h1

=====

**Average Solution Length: 10.06**

Total Solution Length: 503

**Average Search Length: 1841.56**

Total Search Length: 92078

Average No Solution: 0.0

Total No Solution: 0

**Average Execution Time: 1.7546976280212403**

**Total Execution Time: 87.73488140106201**

Analysis of Algorithm: astar-h2

=====

**Average Solution Length: 10.1**

Total Solution Length: 505

**Average Search Length: 1304.28**

Total Search Length: 65214

Average No Solution: 0.0

Total No Solution: 0

**Average Execution Time: 1.0817732429504394**

**Total Execution Time: 54.08866214752197**