

# COMP 472 A2

## Group 9 Presentation

---

MATTHEW SEGAL 40031839

EVAN MATEO 27725167

LEO BENAC 40090034

JOANNA LIN 40055253

# Heuristics

---

Heuristic	Admissibility	Monotonicity	Optimality	Speed
Sum permutation	✗ (3)	✗	worst	fastest
Hamming distance	✓ (2)	✓	best	slowest
Modified Manhattan distance (divided by 2)	✓ (1)	✓	best	fast

# Heuristics

---

## Admissibility of Manhattan distance

3	1	2
4	5	6
7	8	9

Start



1	3	2
4	5	6
7	8	9



1	2	3
4	5	6
7	8	9

Goal

Actual Cost = 2

### Heuristics

Manhattan distance = 4 = 2 ≤ 2 **✗** → inadmissible

Modified Manhattan distance = 4 / 2 = 2 ≤ 2 **✓** → admissible

# Analysis of Algorithms ( 20 puzzles, 60s time limit )

---

A\*

## Sum permutation

Metric	Average	Total
Solution path length	13.5	270
Search path length	96.3	1926
Cost	188.7	3774
No solution	0	0
Execution time	1.24	24.89

## Hamming

Metric	Average	Total
Solution path length	9.8	157
Search path length	222	3552
Cost	86.19	1379
No solution	4	4
Execution time	5.58	89.29

## Modified Manhattan

Metric	Average	Total
Solution path length	10.4	208
Search path length	145.6	2912
Cost	93.5	1870
No solution	0	0
Execution time	5.05	100.96

# Analysis of Algorithms ( 20 puzzles, 60s time limit )

---

## DFS

Metric	Average	Total
Solution path length	Inf	Inf
Search path length	Inf	Inf
Cost	Inf	Inf
No solution	20	20
Execution time	Inf	Inf

## Iterative deepening DFS

Metric	Average	Total
Solution path length	Inf	Inf
Search path length	Inf	Inf
Cost	Inf	Inf
No solution	20	20
Execution time	Inf	Inf

# Scaled Up Experiments ( 20 puzzles 4 x 4, 30min time limit )

---

## A\* w/ Modified Manhattan

Metric	Average	Total
Solution path length	21	84
Search path length	1198.5	4794
Cost	414	1656
No solution	16	16
Execution time	765.4	3061.63

Puzzles w/ solutions: 3, 7, 10, 11

# Puzzles

---

## Puzzles used for analysis

**3x3**

((9, 2, 8), (6, 5, 1), (7, 3, 4))  
((6, 5, 2), (8, 3, 9), (1, 7, 4))  
((2, 7, 3), (1, 6, 5), (4, 8, 9))  
((5, 9, 2), (6, 3, 4), (8, 7, 1))  
((3, 9, 6), (8, 2, 7), (5, 1, 4))  
((6, 9, 3), (8, 7, 2), (4, 1, 5))  
((1, 8, 7), (6, 9, 2), (4, 3, 5))  
((5, 6, 9), (7, 3, 8), (4, 2, 1))  
((1, 7, 6), (9, 4, 8), (2, 3, 5))  
((4, 5, 3), (9, 6, 2), (1, 8, 7))  
((6, 4, 2), (3, 1, 7), (5, 8, 9))  
((7, 8, 2), (9, 5, 4), (6, 3, 1))  
((5, 2, 9), (3, 1, 7), (4, 6, 8))  
((9, 4, 5), (3, 1, 6), (2, 8, 7))  
((9, 8, 6), (5, 2, 3), (4, 1, 7))  
((1, 6, 7), (9, 5, 8), (3, 4, 2))  
((5, 6, 3), (8, 4, 2), (1, 9, 7))  
((7, 1, 8), (9, 5, 4), (3, 2, 6))  
((8, 2, 5), (7, 6, 1), (4, 9, 3))  
((8, 5, 1), (4, 7, 3), (2, 6, 9))

**4x4**

((11, 6, 4, 8), (12, 15, 16, 14), (1, 5, 13, 10), (3, 7, 9, 2))  
((7, 16, 11, 4), (5, 2, 10, 1), (6, 13, 15, 3), (12, 8, 9, 14))  
((1, 11, 12, 2), (13, 8, 10, 7), (5, 3, 9, 4), (16, 14, 6, 15))  
((1, 16, 10, 4), (11, 15, 2, 12), (5, 7, 8, 3), (9, 13, 6, 14))  
((2, 4, 16, 8), (12, 13, 14, 5), (9, 15, 7, 11), (3, 10, 1, 6))  
((2, 14, 8, 11), (6, 12, 9, 3), (5, 1, 10, 15), (7, 13, 4, 16))  
((15, 3, 10, 13), (14, 4, 5, 16), (1, 7, 12, 11), (2, 8, 6, 9))  
((7, 13, 4, 8), (12, 2, 11, 9), (10, 15, 6, 5), (1, 16, 3, 14))  
((1, 5, 7, 9), (14, 2, 13, 12), (6, 11, 3, 4), (10, 15, 16, 8))  
((11, 6, 5, 9), (15, 12, 1, 8), (10, 13, 4, 14), (3, 2, 7, 16))  
((1, 10, 3, 15), (6, 2, 8, 14), (11, 16, 4, 5), (13, 7, 12, 9))  
((9, 6, 8, 7), (3, 16, 4, 5), (13, 15, 12, 1), (2, 10, 11, 14))  
((11, 4, 15, 7), (8, 14, 2, 13), (3, 12, 9, 6), (5, 16, 10, 1))  
((9, 2, 12, 15), (5, 4, 8, 16), (3, 14, 1, 13), (7, 11, 6, 10))  
((3, 13, 5, 9), (8, 15, 12, 11), (4, 16, 1, 14), (6, 7, 2, 10))  
((16, 1, 5, 7), (12, 11, 8, 13), (9, 2, 15, 14), (4, 6, 3, 10))  
((9, 5, 14, 7), (10, 2, 8, 3), (6, 11, 13, 1), (15, 16, 4, 12))  
((11, 8, 6, 5), (12, 7, 10, 13), (1, 14, 3, 9), (4, 2, 15, 16))  
((5, 11, 1, 4), (14, 6, 7, 2), (15, 16, 8, 3), (12, 13, 9, 10))  
((6, 12, 2, 13), (11, 1, 14, 8), (10, 5, 9, 4), (3, 16, 7, 15))