Report

April 10, 2022

1 Team Members

Name	ID	Department
Ahmed Ashraf	2022446758	Business
Abdelrhman Mohamed Abdelhady	2022513643	Intelligent Systems
Antonuose Gerges Nageh	20221903971	Intelligent Systems

2 Data Structures used

- Node `custom made structure`
- Tree `custom made structure`
- ArrayList
- Queue
- Stack
- Priority Queue
- HashTable or HashMap
- Enumeration
- ActionPath `custom made structure`

2.1 Action Enumeration

It's used to represent that action taken by the parent node to reach the child node and it contains the following

- `Up`
- `Down`
- `Left`
- `Right`

2.2 Creating the Node

The Node will contain the following data fields:

- `Parent:Node`
- `Children:ArrayList`
- `state:int[][]`
- `stringState:String` *which will be used to generate a hash code and compare it the goal state*
- `Direction:Action` *Action taken to reach this node*

- `depth:int`
- `missingTileRow:int`
- `missingTileCol:int`
- `cost:int`
- `secondaryCost:int`

It contains the following methods:

- `CreateStringBoard():String` which creates a string of the state of the node
- `addChild(Node child):void` which is a helper function when creating a child‘
- `CreateChild(int a, int b):void` Creates a child where paramaters a,b represent the position of missing tile
- `getRowCol(int value):int[]` returns an array of size two, the method is used in calculating the herustic functions
- `equals(Object obj):boolean Override` It overrides the default equals method
- `isGoal():boolean` checks if the Node state is the goal state
- `hashCode():int Override` Overrides the default hashCode method so that it returns the hash code of the string state
- `toString:String Override` Overrides the default toString method and makes use of the string builder object

2.3 Creating the ActionPath class

The main goal of this class is to use backtracking to print the path from the goal node to the root node with the neccessary information. The class uses the Stack data-structure -*LIFO*- where we are going to use the property that the last element that enters the stack is the first element that exits.

The methods the class contains are as follows:

- `getPath:Stack`
- `printPath:void`

`getpath` functions takes both the root node and goal node as inputs and pushes the goal node parents inside it till it reaches the root node.

`printPath` function returns nothing, it is only used for printing the path from the root to the goal.

2.4 Creating the Tree

The tree class will contain the search functions as well as the expand function for the nodes.

Moreover, it contains the $f_1\&f_2\&f_3\&f_5\&f_6\&f_7$ Comparator objects that will be used in the priority queue.

It has only one data field : `root:Node`

The inner classes:

- `f1` $\rightarrow$ $f_1(n) = g(n)$ where g is the cumulative cost
- `f2` $\rightarrow$ $f_2(n) = h(n)$ where h is the manhattan distance
- `f3` $\rightarrow$ $f_3(n) = h(n)$ where h is the euclidean distance
- `f4` $\rightarrow$ $f_4(n) = h(n)$ where h is the misplaced tiles herustic

- `f5` $\rightarrow f_5(n) = g(n) + h(n)$ where h is the manhattan distance
- `f6` $\rightarrow f_6(n) = g(n) + h(n)$ where h is the euclidean distance
- `f7` $\rightarrow f_7(n) = g(n) + h(n)$ where h is the mispalced tiles herustic

The methods:

- `expand(Node node):List<Node>` returns a list of the children of the given node i.e. it expands it.
- `breadthFirstSearch():boolean` returns true if the search is successful otherwise return false.
- `depthFirstSearch():boolean` returns true if the search is successful otherwise return false.
- `misplacedTiles(Node n):int` returns the number of misplaced tiles.
- `manhattanDistance(Node n):int` which calculates the manhattan distance.
- `euclideanDistance(Node n):int` which calculates the euclidean distance.
- `uniformCostSearch():boolean` implements the uniform cost search.
- `bestFirstSearch(int i):boolean` which implementes the Best first search, `i` parameter is used to determine which herustic to use
- `aStar(int i):boolean` which implements the A* star search, `i` parameter is used to determine which herustic to use

2.5 Creating the Main function

The main function contains checks for the correct input and choices while running the solver.

3 Assumtions

While creating the node the cost of each movement is 1 which not extremely beneficial, so we have created a secondary cost, which will be used in A* search instead of normal cost, is cumulation of numbers swapped for example, we have swapped 0 with 4 which makes $g(n) = 4$ then swapped 0 with 5 then $g(n) = 9$. This will gives a better estimate than normal cost of the movement

4 Breadth First Search : 1 2 3 4 0 5 6 7 8

Testing the Breadth first search.

Breadth frist search was implemented with an early goal test.

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 1
The root node
```

```
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       2       3
0       4       5
6       7       8

Direction Moved: Left
Depth          : 1
Primary   Cost : 1
Secondary Cost : 4
-----------------------
Current Node:

0       2       3
1       4       5
6       7       8

Direction Moved: Up
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
-----------------------
Current Node:

2       0       3
1       4       5
6       7       8

Direction Moved: Right
Depth          : 3
Primary   Cost : 3
Secondary Cost : 7
-----------------------
Current Node:

2       3       0
1       4       5
6       7       8

Direction Moved: Right
Depth          : 4
Primary   Cost : 4
Secondary Cost : 10
-----------------------
```

```
Current Node:

2       3       5
1       4       0
6       7       8

Direction Moved: Down
Depth          : 5
Primary   Cost : 5
Secondary Cost : 15
-----------------------
Current Node:

2       3       5
1       0       4
6       7       8

Direction Moved: Left
Depth          : 6
Primary   Cost : 6
Secondary Cost : 19
-----------------------
Current Node:

2       0       5
1       3       4
6       7       8

Direction Moved: Up
Depth          : 7
Primary   Cost : 7
Secondary Cost : 22
-----------------------
Current Node:

0       2       5
1       3       4
6       7       8

Direction Moved: Left
Depth          : 8
Primary   Cost : 8
Secondary Cost : 24
-----------------------
Current Node:

1       2       5
0       3       4
```

```
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 25
-----------------------
Current Node:

1       2       5
3       0       4
6       7       8

Direction Moved: Right
Depth          : 10
Primary   Cost : 10
Secondary Cost : 28
-----------------------
Current Node:

1       2       5
3       4       0
6       7       8

Direction Moved: Right
Depth          : 11
Primary   Cost : 11
Secondary Cost : 32
-----------------------
Current Node:

1       2       0
3       4       5
6       7       8

Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 37
-----------------------
Current Node:

1       0       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 13
```

```
Primary   Cost : 13
Secondary Cost : 39
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 14
Primary   Cost : 14
Secondary Cost : 40
-----------------------
Time: 72.0 millie seconds
Space: 3718
```

5 Depth First Search : 1 0 2 3 4 5 6 7 8

The implementation contains an early goal test even though that might not result in the optimal solution but it results in better search time. Since the solution of the depth first search is sub-optimal in most cases. An example file has been added which highlights the output of the algorithm with different input

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 0 2 3 4 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 2
The root node
1       0       2
3       4       5
6       7       8
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 1
```

```
Primary   Cost : 1
Secondary Cost : 1
-----------------------
Time: 0.0 millie seconds
Space: 2
```

6 Uniform Cost Search : 1 2 3 4 0 5 6 7 8

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 3
The root node
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       2       3
0       4       5
6       7       8

Direction Moved: Left
Depth          : 1
Primary   Cost : 1
Secondary Cost : 4
-----------------------
Current Node:

0       2       3
1       4       5
6       7       8

Direction Moved: Up
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
-----------------------
Current Node:
```

```
2       0       3
1       4       5
6       7       8

Direction Moved: Right
Depth          : 3
Primary   Cost : 3
Secondary Cost : 7
-----------------------
Current Node:

2       3       0
1       4       5
6       7       8

Direction Moved: Right
Depth          : 4
Primary   Cost : 4
Secondary Cost : 10
-----------------------
Current Node:

2       3       5
1       4       0
6       7       8

Direction Moved: Down
Depth          : 5
Primary   Cost : 5
Secondary Cost : 15
-----------------------
Current Node:

2       3       5
1       0       4
6       7       8

Direction Moved: Left
Depth          : 6
Primary   Cost : 6
Secondary Cost : 19
-----------------------
Current Node:

2       0       5
1       3       4
6       7       8
```

```
Direction Moved: Up
Depth          : 7
Primary   Cost : 7
Secondary Cost : 22
-----------------------
Current Node:

0       2       5
1       3       4
6       7       8

Direction Moved: Left
Depth          : 8
Primary   Cost : 8
Secondary Cost : 24
-----------------------
Current Node:

1       2       5
0       3       4
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 25
-----------------------
Current Node:

1       2       5
3       0       4
6       7       8

Direction Moved: Right
Depth          : 10
Primary   Cost : 10
Secondary Cost : 28
-----------------------
Current Node:

1       2       5
3       4       0
6       7       8

Direction Moved: Right
Depth          : 11
Primary   Cost : 11
```

```
Secondary Cost : 32
-----------------------
Current Node:

1       2       0
3       4       5
6       7       8

Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 37
-----------------------
Current Node:

1       0       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 13
Primary   Cost : 13
Secondary Cost : 39
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 14
Primary   Cost : 14
Secondary Cost : 40
-----------------------
Time: 9.0 millie seconds
Space: 1099
```

7 Best first Search : 1 2 3 4 0 5 6 7 8

7.1 Manhattan Distance

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
```

```
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 4
Choose the Heuristic function

1. Manhattan Distance
2. Euclidean Distance
3. Misplaced Tiles

Enter your choice: 1
The root node
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       2       3
4       5       0
6       7       8

Direction Moved: Right
Depth          : 1
Primary   Cost : 1
Secondary Cost : 5
-----------------------
Current Node:

1       2       0
4       5       3
6       7       8

Direction Moved: Up
Depth          : 2
Primary   Cost : 2
Secondary Cost : 8
-----------------------
Current Node:

1       0       2
4       5       3
6       7       8

Direction Moved: Left
Depth          : 3
Primary   Cost : 3
```

```
Secondary Cost : 10
-----------------------
Current Node:

1       5       2
4       0       3
6       7       8

Direction Moved: Down
Depth          : 4
Primary   Cost : 4
Secondary Cost : 15
-----------------------
Current Node:

1       5       2
4       3       0
6       7       8

Direction Moved: Right
Depth          : 5
Primary   Cost : 5
Secondary Cost : 18
-----------------------
Current Node:

1       5       0
4       3       2
6       7       8

Direction Moved: Up
Depth          : 6
Primary   Cost : 6
Secondary Cost : 20
-----------------------
Current Node:

1       0       5
4       3       2
6       7       8

Direction Moved: Left
Depth          : 7
Primary   Cost : 7
Secondary Cost : 25
-----------------------
Current Node:
```

```
0       1       5
4       3       2
6       7       8

Direction Moved: Left
Depth          : 8
Primary   Cost : 8
Secondary Cost : 26
-----------------------
Current Node:

4       1       5
0       3       2
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 30
-----------------------
Current Node:

4       1       5
3       0       2
6       7       8

Direction Moved: Right
Depth          : 10
Primary   Cost : 10
Secondary Cost : 33
-----------------------
Current Node:

4       1       5
3       2       0
6       7       8

Direction Moved: Right
Depth          : 11
Primary   Cost : 11
Secondary Cost : 35
-----------------------
Current Node:

4       1       0
3       2       5
6       7       8
```

```
Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 40
-----------------------
Current Node:

4       0       1
3       2       5
6       7       8

Direction Moved: Left
Depth          : 13
Primary   Cost : 13
Secondary Cost : 41
-----------------------
Current Node:

0       4       1
3       2       5
6       7       8

Direction Moved: Left
Depth          : 14
Primary   Cost : 14
Secondary Cost : 45
-----------------------
Current Node:

3       4       1
0       2       5
6       7       8

Direction Moved: Down
Depth          : 15
Primary   Cost : 15
Secondary Cost : 48
-----------------------
Current Node:

3       4       1
2       0       5
6       7       8

Direction Moved: Right
Depth          : 16
Primary   Cost : 16
Secondary Cost : 50
```

```
----------------------
Current Node:

3       0       1
2       4       5
6       7       8

Direction Moved: Up
Depth          : 17
Primary   Cost : 17
Secondary Cost : 54
----------------------
Current Node:

3       1       0
2       4       5
6       7       8

Direction Moved: Right
Depth          : 18
Primary   Cost : 18
Secondary Cost : 55
----------------------
Current Node:

3       1       5
2       4       0
6       7       8

Direction Moved: Down
Depth          : 19
Primary   Cost : 19
Secondary Cost : 60
----------------------
Current Node:

3       1       5
2       0       4
6       7       8

Direction Moved: Left
Depth          : 20
Primary   Cost : 20
Secondary Cost : 64
----------------------
Current Node:

3       1       5
```

```
0       2       4
6       7       8

Direction Moved: Left
Depth          : 21
Primary   Cost : 21
Secondary Cost : 66
-----------------------
Current Node:

0       1       5
3       2       4
6       7       8

Direction Moved: Up
Depth          : 22
Primary   Cost : 22
Secondary Cost : 69
-----------------------
Current Node:

1       0       5
3       2       4
6       7       8

Direction Moved: Right
Depth          : 23
Primary   Cost : 23
Secondary Cost : 70
-----------------------
Current Node:

1       2       5
3       0       4
6       7       8

Direction Moved: Down
Depth          : 24
Primary   Cost : 24
Secondary Cost : 72
-----------------------
Current Node:

1       2       5
3       4       0
6       7       8

Direction Moved: Right
```

```
Depth          : 25
Primary   Cost : 25
Secondary Cost : 76
-----------------------
Current Node:

1       2       0
3       4       5
6       7       8

Direction Moved: Up
Depth          : 26
Primary   Cost : 26
Secondary Cost : 81
-----------------------
Current Node:

1       0       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 27
Primary   Cost : 27
Secondary Cost : 83
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 28
Primary   Cost : 28
Secondary Cost : 84
-----------------------
Time: 14.0 millie seconds
Space: 142
```

7.2 Euclidean Distance

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
```

```
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 4
Choose the Heuristic function

1. Manhattan Distance
2. Euclidean Distance
3. Misplaced Tiles

Enter your choice: 2
The root node
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       0       3
4       2       5
6       7       8

Direction Moved: Up
Depth          : 1
Primary   Cost : 1
Secondary Cost : 2
-----------------------
Current Node:

1       3       0
4       2       5
6       7       8

Direction Moved: Right
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
-----------------------
Current Node:

1       3       5
4       2       0
6       7       8

Direction Moved: Down
Depth          : 3
Primary   Cost : 3
```

```
Secondary Cost : 10
-----------------------
Current Node:

1       3       5
4       0       2
6       7       8

Direction Moved: Left
Depth          : 4
Primary   Cost : 4
Secondary Cost : 12
-----------------------
Current Node:

1       3       5
0       4       2
6       7       8

Direction Moved: Left
Depth          : 5
Primary   Cost : 5
Secondary Cost : 16
-----------------------
Current Node:

0       3       5
1       4       2
6       7       8

Direction Moved: Up
Depth          : 6
Primary   Cost : 6
Secondary Cost : 17
-----------------------
Current Node:

3       0       5
1       4       2
6       7       8

Direction Moved: Right
Depth          : 7
Primary   Cost : 7
Secondary Cost : 20
-----------------------
Current Node:
```

```
3       5       0
1       4       2
6       7       8

Direction Moved: Right
Depth          : 8
Primary   Cost : 8
Secondary Cost : 25
-----------------------
Current Node:

3       5       2
1       4       0
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 27
-----------------------
Current Node:

3       5       2
1       0       4
6       7       8

Direction Moved: Left
Depth          : 10
Primary   Cost : 10
Secondary Cost : 31
-----------------------
Current Node:

3       5       2
0       1       4
6       7       8

Direction Moved: Left
Depth          : 11
Primary   Cost : 11
Secondary Cost : 32
-----------------------
Current Node:

0       5       2
3       1       4
6       7       8
```

```
Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 35
-----------------------
Current Node:

5       0       2
3       1       4
6       7       8

Direction Moved: Right
Depth          : 13
Primary   Cost : 13
Secondary Cost : 40
-----------------------
Current Node:

5       1       2
3       0       4
6       7       8

Direction Moved: Down
Depth          : 14
Primary   Cost : 14
Secondary Cost : 41
-----------------------
Current Node:

5       1       2
3       4       0
6       7       8

Direction Moved: Right
Depth          : 15
Primary   Cost : 15
Secondary Cost : 45
-----------------------
Current Node:

5       1       0
3       4       2
6       7       8

Direction Moved: Up
Depth          : 16
Primary   Cost : 16
Secondary Cost : 47
```

```
----------------------
Current Node:

5       0       1
3       4       2
6       7       8

Direction Moved: Left
Depth          : 17
Primary   Cost : 17
Secondary Cost : 48
----------------------
Current Node:

0       5       1
3       4       2
6       7       8

Direction Moved: Left
Depth          : 18
Primary   Cost : 18
Secondary Cost : 53
----------------------
Current Node:

3       5       1
0       4       2
6       7       8

Direction Moved: Down
Depth          : 19
Primary   Cost : 19
Secondary Cost : 56
----------------------
Current Node:

3       5       1
4       0       2
6       7       8

Direction Moved: Right
Depth          : 20
Primary   Cost : 20
Secondary Cost : 60
----------------------
Current Node:

3       0       1
```

```
4       5       2
6       7       8

Direction Moved: Up
Depth          : 21
Primary   Cost : 21
Secondary Cost : 65
-----------------------
Current Node:

3       1       0
4       5       2
6       7       8

Direction Moved: Right
Depth          : 22
Primary   Cost : 22
Secondary Cost : 66
-----------------------
Current Node:

3       1       2
4       5       0
6       7       8

Direction Moved: Down
Depth          : 23
Primary   Cost : 23
Secondary Cost : 68
-----------------------
Current Node:

3       1       2
4       0       5
6       7       8

Direction Moved: Left
Depth          : 24
Primary   Cost : 24
Secondary Cost : 73
-----------------------
Current Node:

3       1       2
0       4       5
6       7       8

Direction Moved: Left
```

```
Depth          : 25
Primary   Cost : 25
Secondary Cost : 77
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Up
Depth          : 26
Primary   Cost : 26
Secondary Cost : 80
-----------------------
Time: 1.0 millie seconds
Space: 144
```

7.3 Misplaced Tiles

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 4
Choose the Heuristic function

1. Manhattan Distance
2. Euclidean Distance
3. Misplaced Tiles

Enter your choice: 3
The root node
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       0       3
4       2       5
6       7       8
```

```
Direction Moved: Up
Depth          : 1
Primary   Cost : 1
Secondary Cost : 2
-----------------------
Current Node:

1       3       0
4       2       5
6       7       8

Direction Moved: Right
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
-----------------------
Current Node:

1       3       5
4       2       0
6       7       8

Direction Moved: Down
Depth          : 3
Primary   Cost : 3
Secondary Cost : 10
-----------------------
Current Node:

1       3       5
4       0       2
6       7       8

Direction Moved: Left
Depth          : 4
Primary   Cost : 4
Secondary Cost : 12
-----------------------
Current Node:

1       3       5
0       4       2
6       7       8

Direction Moved: Left
Depth          : 5
Primary   Cost : 5
```

```
Secondary Cost : 16
-----------------------
Current Node:

0       3       5
1       4       2
6       7       8

Direction Moved: Up
Depth          : 6
Primary   Cost : 6
Secondary Cost : 17
-----------------------
Current Node:

3       0       5
1       4       2
6       7       8

Direction Moved: Right
Depth          : 7
Primary   Cost : 7
Secondary Cost : 20
-----------------------
Current Node:

3       5       0
1       4       2
6       7       8

Direction Moved: Right
Depth          : 8
Primary   Cost : 8
Secondary Cost : 25
-----------------------
Current Node:

3       5       2
1       4       0
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 27
-----------------------
Current Node:
```

```
3       5       2
1       0       4
6       7       8

Direction Moved: Left
Depth          : 10
Primary   Cost : 10
Secondary Cost : 31
-----------------------
Current Node:

3       5       2
0       1       4
6       7       8

Direction Moved: Left
Depth          : 11
Primary   Cost : 11
Secondary Cost : 32
-----------------------
Current Node:

0       5       2
3       1       4
6       7       8

Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 35
-----------------------
Current Node:

5       0       2
3       1       4
6       7       8

Direction Moved: Right
Depth          : 13
Primary   Cost : 13
Secondary Cost : 40
-----------------------
Current Node:

5       1       2
3       0       4
6       7       8
```

```
Direction Moved: Down
Depth          : 14
Primary   Cost : 14
Secondary Cost : 41
-----------------------
Current Node:

5       1       2
3       4       0
6       7       8

Direction Moved: Right
Depth          : 15
Primary   Cost : 15
Secondary Cost : 45
-----------------------
Current Node:

5       1       0
3       4       2
6       7       8

Direction Moved: Up
Depth          : 16
Primary   Cost : 16
Secondary Cost : 47
-----------------------
Current Node:

5       0       1
3       4       2
6       7       8

Direction Moved: Left
Depth          : 17
Primary   Cost : 17
Secondary Cost : 48
-----------------------
Current Node:

0       5       1
3       4       2
6       7       8

Direction Moved: Left
Depth          : 18
Primary   Cost : 18
Secondary Cost : 53
```

```
----------------------
Current Node:

3       5       1
0       4       2
6       7       8

Direction Moved: Down
Depth          : 19
Primary   Cost : 19
Secondary Cost : 56
----------------------
Current Node:

3       5       1
4       0       2
6       7       8

Direction Moved: Right
Depth          : 20
Primary   Cost : 20
Secondary Cost : 60
----------------------
Current Node:

3       0       1
4       5       2
6       7       8

Direction Moved: Up
Depth          : 21
Primary   Cost : 21
Secondary Cost : 65
----------------------
Current Node:

3       1       0
4       5       2
6       7       8

Direction Moved: Right
Depth          : 22
Primary   Cost : 22
Secondary Cost : 66
----------------------
Current Node:

3       1       2
```

```
4       5       0
6       7       8

Direction Moved: Down
Depth          : 23
Primary   Cost : 23
Secondary Cost : 68
-----------------------
Current Node:

3       1       2
4       0       5
6       7       8

Direction Moved: Left
Depth          : 24
Primary   Cost : 24
Secondary Cost : 73
-----------------------
Current Node:

3       1       2
0       4       5
6       7       8

Direction Moved: Left
Depth          : 25
Primary   Cost : 25
Secondary Cost : 77
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Up
Depth          : 26
Primary   Cost : 26
Secondary Cost : 80
-----------------------
Time: 1.0 millie seconds
Space: 186
```

8 A* search : 1 2 3 4 0 5 6 7 8

8.1 Manhattan distance

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 5
Choose the Heuristic function

1. Manhattan Distance
2. Euclidean Distance
3. Misplaced Tiles

Enter your choice: 1
The root node
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       2       3
0       4       5
6       7       8

Direction Moved: Left
Depth          : 1
Primary   Cost : 1
Secondary Cost : 4
-----------------------
Current Node:

0       2       3
1       4       5
6       7       8

Direction Moved: Up
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
```

```
----------------------
Current Node:

2       0       3
1       4       5
6       7       8

Direction Moved: Right
Depth          : 3
Primary   Cost : 3
Secondary Cost : 7
----------------------
Current Node:

2       3       0
1       4       5
6       7       8

Direction Moved: Right
Depth          : 4
Primary   Cost : 4
Secondary Cost : 10
----------------------
Current Node:

2       3       5
1       4       0
6       7       8

Direction Moved: Down
Depth          : 5
Primary   Cost : 5
Secondary Cost : 15
----------------------
Current Node:

2       3       5
1       0       4
6       7       8

Direction Moved: Left
Depth          : 6
Primary   Cost : 6
Secondary Cost : 19
----------------------
Current Node:

2       0       5
```

```
1       3       4
6       7       8

Direction Moved: Up
Depth          : 7
Primary   Cost : 7
Secondary Cost : 22
-----------------------
Current Node:

0       2       5
1       3       4
6       7       8

Direction Moved: Left
Depth          : 8
Primary   Cost : 8
Secondary Cost : 24
-----------------------
Current Node:

1       2       5
0       3       4
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 25
-----------------------
Current Node:

1       2       5
3       0       4
6       7       8

Direction Moved: Right
Depth          : 10
Primary   Cost : 10
Secondary Cost : 28
-----------------------
Current Node:

1       2       5
3       4       0
6       7       8

Direction Moved: Right
```

```
Depth          : 11
Primary   Cost : 11
Secondary Cost : 32
-----------------------
Current Node:

1       2       0
3       4       5
6       7       8

Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 37
-----------------------
Current Node:

1       0       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 13
Primary   Cost : 13
Secondary Cost : 39
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 14
Primary   Cost : 14
Secondary Cost : 40
-----------------------
Time: 2.0 millie seconds
Space: 371
```

8.2 Euclidean distance

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
```

```
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 5
Choose the Heuristic function

1. Manhattan Distance
2. Euclidean Distance
3. Misplaced Tiles

Enter your choice: 2
The root node
1       2       3
4       0       5
6       7       8
-----------------------
Current Node:

1       2       3
0       4       5
6       7       8

Direction Moved: Left
Depth          : 1
Primary   Cost : 1
Secondary Cost : 4
-----------------------
Current Node:

0       2       3
1       4       5
6       7       8

Direction Moved: Up
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
-----------------------
Current Node:

2       0       3
1       4       5
6       7       8

Direction Moved: Right
Depth          : 3
Primary   Cost : 3
```

```
Secondary Cost : 7
----------------------
Current Node:

2       3       0
1       4       5
6       7       8

Direction Moved: Right
Depth          : 4
Primary   Cost : 4
Secondary Cost : 10
----------------------
Current Node:

2       3       5
1       4       0
6       7       8

Direction Moved: Down
Depth          : 5
Primary   Cost : 5
Secondary Cost : 15
----------------------
Current Node:

2       3       5
1       0       4
6       7       8

Direction Moved: Left
Depth          : 6
Primary   Cost : 6
Secondary Cost : 19
----------------------
Current Node:

2       0       5
1       3       4
6       7       8

Direction Moved: Up
Depth          : 7
Primary   Cost : 7
Secondary Cost : 22
----------------------
Current Node:
```

```
0       2       5
1       3       4
6       7       8

Direction Moved: Left
Depth          : 8
Primary   Cost : 8
Secondary Cost : 24
-----------------------
Current Node:

1       2       5
0       3       4
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 25
-----------------------
Current Node:

1       2       5
3       0       4
6       7       8

Direction Moved: Right
Depth          : 10
Primary   Cost : 10
Secondary Cost : 28
-----------------------
Current Node:

1       2       5
3       4       0
6       7       8

Direction Moved: Right
Depth          : 11
Primary   Cost : 11
Secondary Cost : 32
-----------------------
Current Node:

1       2       0
3       4       5
6       7       8
```

```
Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 37
-----------------------
Current Node:

1       0       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 13
Primary   Cost : 13
Secondary Cost : 39
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 14
Primary   Cost : 14
Secondary Cost : 40
-----------------------
Time: 1.0 millie seconds
Space: 494
```

8.3 Misplaced Tiles

```
Welcome to 8 puzzle Solver
Enter the puzzle : 1 2 3 4 0 5 6 7 8
Choose the Algorithm

1. Breadth First Search
2. Depth First Search
3. Uniform Cost  Search
4. Best First Search
5. A*

Enter your choice: 5
Choose the Heuristic function

1. Manhattan Distance
2. Euclidean Distance
3. Misplaced Tiles
```

```
Enter your choice: 3
The root node
1       2       3
4       0       5
6       7       8
----------------------
Current Node:

1       2       3
0       4       5
6       7       8

Direction Moved: Left
Depth          : 1
Primary   Cost : 1
Secondary Cost : 4
----------------------
Current Node:

0       2       3
1       4       5
6       7       8

Direction Moved: Up
Depth          : 2
Primary   Cost : 2
Secondary Cost : 5
----------------------
Current Node:

2       0       3
1       4       5
6       7       8

Direction Moved: Right
Depth          : 3
Primary   Cost : 3
Secondary Cost : 7
----------------------
Current Node:

2       3       0
1       4       5
6       7       8

Direction Moved: Right
Depth          : 4
```

```
Primary   Cost : 4
Secondary Cost : 10
-----------------------
Current Node:

2       3       5
1       4       0
6       7       8

Direction Moved: Down
Depth          : 5
Primary   Cost : 5
Secondary Cost : 15
-----------------------
Current Node:

2       3       5
1       0       4
6       7       8

Direction Moved: Left
Depth          : 6
Primary   Cost : 6
Secondary Cost : 19
-----------------------
Current Node:

2       0       5
1       3       4
6       7       8

Direction Moved: Up
Depth          : 7
Primary   Cost : 7
Secondary Cost : 22
-----------------------
Current Node:

0       2       5
1       3       4
6       7       8

Direction Moved: Left
Depth          : 8
Primary   Cost : 8
Secondary Cost : 24
-----------------------
Current Node:
```

```
1       2       5
0       3       4
6       7       8

Direction Moved: Down
Depth          : 9
Primary   Cost : 9
Secondary Cost : 25
-----------------------
Current Node:

1       2       5
3       0       4
6       7       8

Direction Moved: Right
Depth          : 10
Primary   Cost : 10
Secondary Cost : 28
-----------------------
Current Node:

1       2       5
3       4       0
6       7       8

Direction Moved: Right
Depth          : 11
Primary   Cost : 11
Secondary Cost : 32
-----------------------
Current Node:

1       2       0
3       4       5
6       7       8

Direction Moved: Up
Depth          : 12
Primary   Cost : 12
Secondary Cost : 37
-----------------------
Current Node:

1       0       2
3       4       5
6       7       8
```

```
Direction Moved: Left
Depth          : 13
Primary   Cost : 13
Secondary Cost : 39
-----------------------
Current Node:

0       1       2
3       4       5
6       7       8

Direction Moved: Left
Depth          : 14
Primary   Cost : 14
Secondary Cost : 40
-----------------------
Time: 1.0 millie seconds
Space: 530
```