# **8-Puzzle Problem Report**

## **1. Problem Formulation**

The 8-puzzle consists of a 3x3 grid with 8 numbered tiles and one empty space. The objective is to move the tiles from an initial board state into a predefined goal state using valid moves (up, down, left, right). In this implementation, the problem will be solved using the A\* search algorithm alongside the heuristic functions of Manhattan Distance and Misplaced Tiles.

## **2. Program Structure**

### **2.1 Global Variables**

* **initial\_state:** User input for the starting configuration of the puzzle.
* **goal\_state:** User input for the goal configuration of the puzzle.
* **nodes\_generated:** Counts the number of nodes generated during the solution of the puzzle.
* **nodes\_expanded:** Counts the number of nodes expanded upon during the solution of the puzzle.
* **open\_list:** A heap-style priority queue that contains the states to be exploring with A\*
* **closed\_list:** A set storing explored states to prevent redundancy
* **MOVES:** Python dictionary containing a mapping of the tile changes to their board index changes.
* **Priority Queue:** Stores nodes during the search.

### **2.2 Functions and Procedures**

* **get\_user\_input():** Prompts the user to enter both an initial board state and a goal state then stores them in their respective lists.
* **print\_board(board):** Displays the board in a 3x3 grid.
* **move\_tile(board, move, blank\_index):** Moves the blank tile in the specified direction and returns the new state of the board.
* **manhattan\_distance(board, goal\_state):** Calculates the manhattan distance of the board with the current state of the board and the goal state.
* **misplaced\_tile(board, goal\_state):** Calculates the misplaced tiles on the board with the current state of the board and the goal state.
* **a\_star(initial\_state, goal\_state, heuristic):** Implementation of the A\* search algorithm that explores different states based on their heuristic value and cost.
* **print\_solution(solution):** Traces back through the solution and prints the solution path from the initial state to the goal state.

## **3. Analysis of Input/Output Cases**

Eight different test cases were analyzed using Manhattan and Misplaced Tiles heuristics.

### **3.1 Test Cases**

| **Test Case** | **Initial State** | **Goal State** |
| --- | --- | --- |
| 1 | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | 4 | | 5 |  +---+---+---+  | 6 | 7 | 8 |  +---+---+---+ | +---+---+---+  | | 1 | 2 |  +---+---+---+  | 3 | 4 | 5 |  +---+---+---+  | 6 | 7 | 8 |  +---+---+---+ |
| 2 | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | 4 | 5 | 6 |  +---+---+---+  | 7 | 8 | |  +---+---+---+ | +---+---+---+  | | 8 | 7 |  +---+---+---+  | 6 | 5 | 4 |  +---+---+---+  | 3 | 2 | 1 |  +---+---+---+ |
| 3 | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | 4 | | 5 |  +---+---+---+  | 6 | 7 | 8 |  +---+---+---+ | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | | 4 | 5 |  +---+---+---+  | 6 | 7 | 8 |  +---+---+---+ |
| 4 | +---+---+---+  | 4 | 1 | 3 |  +---+---+---+  | 2 | 5 | |  +---+---+---+  | 7 | 6 | 8 |  +---+---+---+ | +---+---+---+  | 8 | 6 | 7 |  +---+---+---+  | 3 | 4 | 1 |  +---+---+---+  | 5 | | 2 |  +---+---+---+ |
| 5 | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | 4 | 8 | |  +---+---+---+  | 7 | 6 | 5 |  +---+---+---+ | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | 4 | 5 | 6 |  +---+---+---+  | 7 | 8 | |  +---+---+---+ |
| 6 | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | | 5 | 4 |  +---+---+---+  | 6 | 8 | 7 |  +---+---+---+ | +---+---+---+  | 1 | 5 | 4 |  +---+---+---+  | | 2 | 3 |  +---+---+---+  | 6 | 8 | 7 |  +---+---+---+ |
| 7 | +---+---+---+  | 1 | 7 | 5 |  +---+---+---+  | 6 | | 4 |  +---+---+---+  | 8 | 2 | 3 |  +---+---+---+ | +---+---+---+  | 7 | 6 | 5 |  +---+---+---+  | | 1 | 4 |  +---+---+---+  | 8 | 2 | 3 |  +---+---+---+ |
| 8 | +---+---+---+  | 1 | 2 | 3 |  +---+---+---+  | 4 | | 5 |  +---+---+---+  | 6 | 7 | 8 |  +---+---+---+ | +---+---+---+  | 3 | 2 | 1 |  +---+---+---+  | 4 | | 5 |  +---+---+---+  | 8 | 7 | 6 |  +---+---+---+ |

### **3.2 Test Case Results**

| **Test Case** | **Heuristic** | **Solution Path** | **Nodes Generated** | **Nodes Expanded** |
| --- | --- | --- | --- | --- |
| 1 | Manhattan Distance | LEFT → UP → RIGHT → RIGHT → DOWN → LEFT → UP → LEFT → DOWN → RIGHT → RIGHT → UP → LEFT → LEFT | 163 | 96 |
| 1 | Misplaced Tiles | LEFT → UP → RIGHT → RIGHT → DOWN → LEFT → UP → LEFT → DOWN → RIGHT → RIGHT → UP → LEFT → LEFT | 597 | 361 |
| 2 | Manhattan Distance | UP → UP → LEFT → LEFT → DOWN → DOWN → RIGHT → RIGHT → UP → UP → LEFT → LEFT → DOWN → DOWN → RIGHT → RIGHT → UP → UP → LEFT → LEFT → DOWN → DOWN → RIGHT → RIGHT → UP → UP → UP → LEFT → LEFT | 974 | 644 |
| 2 | Misplaced Tiles | UP → UP → LEFT → LEFT → DOWN → DOWN → RIGHT → RIGHT → UP → UP → LEFT → LEFT → DOWN → DOWN → RIGHT → RIGHT → UP → UP → LEFT → LEFT → DOWN → DOWN → RIGHT → RIGHT → UP → UP → UP → LEFT → LEFT | 95974 | 68108 |
| 3 | Manhattan Distance | LEFT | 5 | 2 |
| 3 | Misplaced Tiles | LEFT | 5 | 2 |
| 4 | Manhattan Distance | NO SOLUTION | 258978 | 258978 |
| 4 | Misplaced Tiles | NO SOLUTION | 295294 | 295294 |
| 5 | Manhattan Distance | DOWN → LEFT → UP → RIGHT → DOWN | 12 | 6 |
| 5 | Misplaced Tiles | DOWN → LEFT → UP → RIGHT → DOWN | 19 | 10 |
| 6 | Manhattan Distance | DOWN → RIGHT → RIGHT → UP → LEFT → UP → RIGHT → DOWN → DOWN → LEFT → LEFT → UP → RIGHT → UP → RIGHT → DOWN → LEFT → LEFT → | 969 | 591 |
| 6 | Misplaced Tiles | RIGHT → RIGHT → UP → LEFT → DOWN → LEFT → DOWN → RIGHT → RIGHT → UP → UP → LEFT → DOWN → RIGHT → DOWN → LEFT → LEFT → UP → | 2460 | 1479 |
| 7 | Manhattan Distance | LEFT → UP → RIGHT → DOWN → LEFT | 13 | 6 |
| 7 | Misplaced Tiles | LEFT → UP → RIGHT → DOWN → LEFT | 22 | 11 |
| 8 | Manhattan Distance | DOWN → RIGHT → UP → UP → LEFT → LEFT → DOWN → RIGHT → RIGHT → UP → LEFT → DOWN → DOWN → LEFT → UP → RIGHT → RIGHT → DOWN → LEFT → LEFT → UP → UP → RIGHT → DOWN | 2413 | 1487 |
| 8 | Misplaced Tiles | DOWN → RIGHT → UP → UP → LEFT → LEFT → DOWN → RIGHT → RIGHT → UP → LEFT → DOWN → DOWN → LEFT → UP → RIGHT → RIGHT → DOWN → LEFT → LEFT → UP → UP → RIGHT → DOWN | 33433 | 21021 |