

# **CE213 Artificial Intelligence Assignment Report (2020-2021)**

Student ID: 1906027

The purpose of this program is solving the 8-puzzle problem with a given initial state using uniform cost search and A\* search with a heuristic of our choosing.

My solution to this lab involves a Solver class which implements a breadth first search as the basis for uniform cost and A\* search in order to make all search functions as general as they can be. Furthermore, in the case of this puzzle, uniform cost search equals breadth first search, where the cost is considered at each node with the heuristic function  $h_n$  equating to 0 at every move. While uniform cost takes the shortest amount of possible moves to reach the goal state (30), the speed at which it does is not very efficient, taking around 8 minutes to complete.

In relation to A\* search, the implemented heuristic is the Manhattan distance heuristic, which is essentially the sum of absolute values of differences in the goal state's x and y coordinates and the current state's x and y coordinates, for which every tile will have to be moved at least the number of spots in between itself and its correct position. In addition, because of the priority queue used for unexpanded nodes, this search function not only gets the shortest path to the goal state with 30 moves, but also takes a short amount of time at just around two minutes.

Possible tile moves in this program are calculated by acquiring the position of 0 (or empty tile) at each step. Each if statement equates to a possible up, down, right or left move from the position, and for each possible move we are establishing the current state of the grid, and executing a nested for loop to get the position value, and in the end incrementing the movement cost function  $g_n$  by 1 and storing this in an array of lists called the moveArray.

In comparison, while both search functions are able to return the goal state with the least possible number of moves, the optimal algorithm in this case is A\* search with Manhattan distance heuristic because of its time optimality.