Lab Report

Group Members: Armando Hernandez, Michael Zheng, Gabriel Serrano, and Thian Pham.
Contributions:

- **Armando**: Foundation code, A* with euclidean distance, report, and graphs.
- **Michael**: Work on the Uniform Cost search function and A star with the Misplaced Tile Heuristic.
- **Gabriel**: Work on the UI system, debug the functions and testing phase, applied fixes and modifications as to the final code.
- **Thien**: Help Michael with the A star Heuristic distance algorithm. He joined our group late.

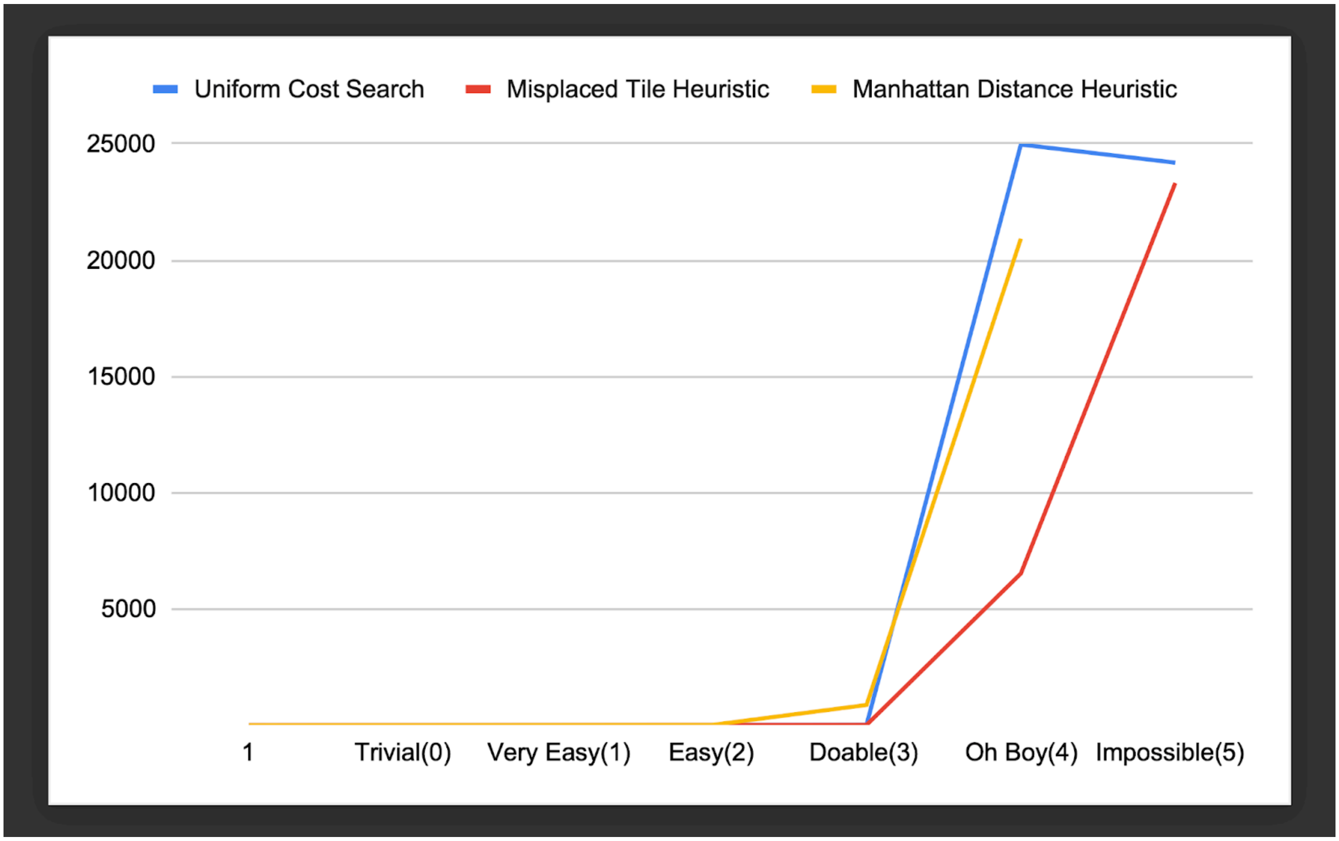
The challenges our group encountered while working on this program were

- Spacing and syntax issues (e.g. Spacing was off);
- Merge conflicts;
- Implementing the programs in Python;
- What data structures to use;
- Language issues (e.g. Different versions of Python);
- Getting the right maximum queue sizes; expanded nodes; etc
- Setting flags for each method so that it uses the correct cost for the comparator function _lt_ (e.g. for misplaced tile: f(n) = g(n) + h(n) and for UCS: f(n) = g(n))

Our design: We used two classes

1) In the problem class we defined the initial and goal states with their corresponding operators;
2) For the puzzle class, we initialized the parameters that are required for all three algorithms;
3) Optimization: We used a set called visited_costs that
    a) Had tracking for the visited states: Kept record of all the states that have been visited during the search;
    b) Had storing minimum costs: visited_costs stores the minimum cost at which this state has been reached. This is crucial because, in problems where paths can have different costs, a state might be worth revisiting if a cheaper path to it is discovered later in the search.

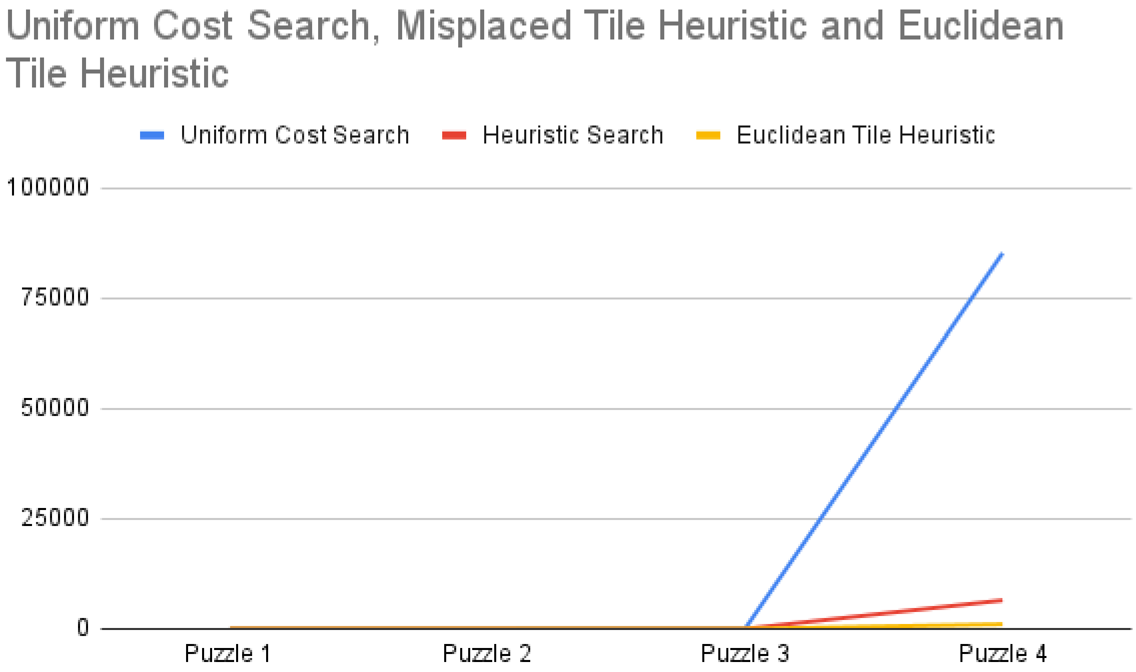
Number of Nodes Expanded Per Puzzle



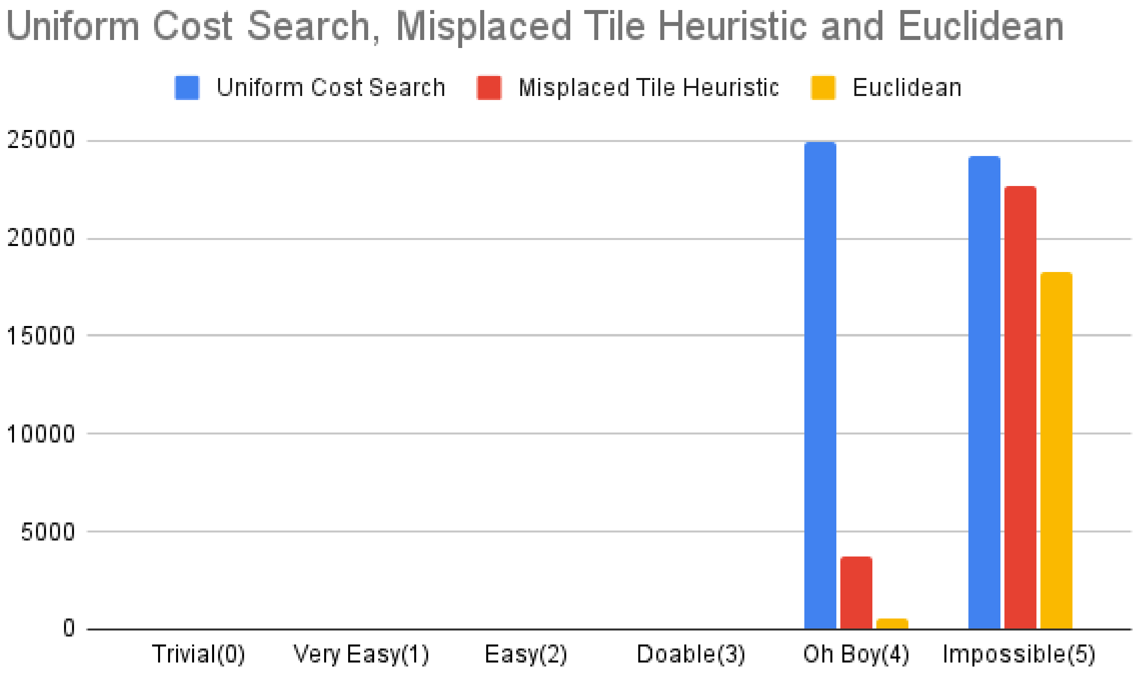
Maximum Queue Size

	Uniform Cost Search	Misplace Tile Heuristic	Euclidean Distance
Trivial(0)	1	1	1
Very Easy(1)	5	3	3
Easy(2)	4	3	3
Doable(3)	18	4	4
Oh Boy(4)	24969	3737	589
Impossible(5)	24175	22691	18224

Maximum Queue Size Per Puzzle



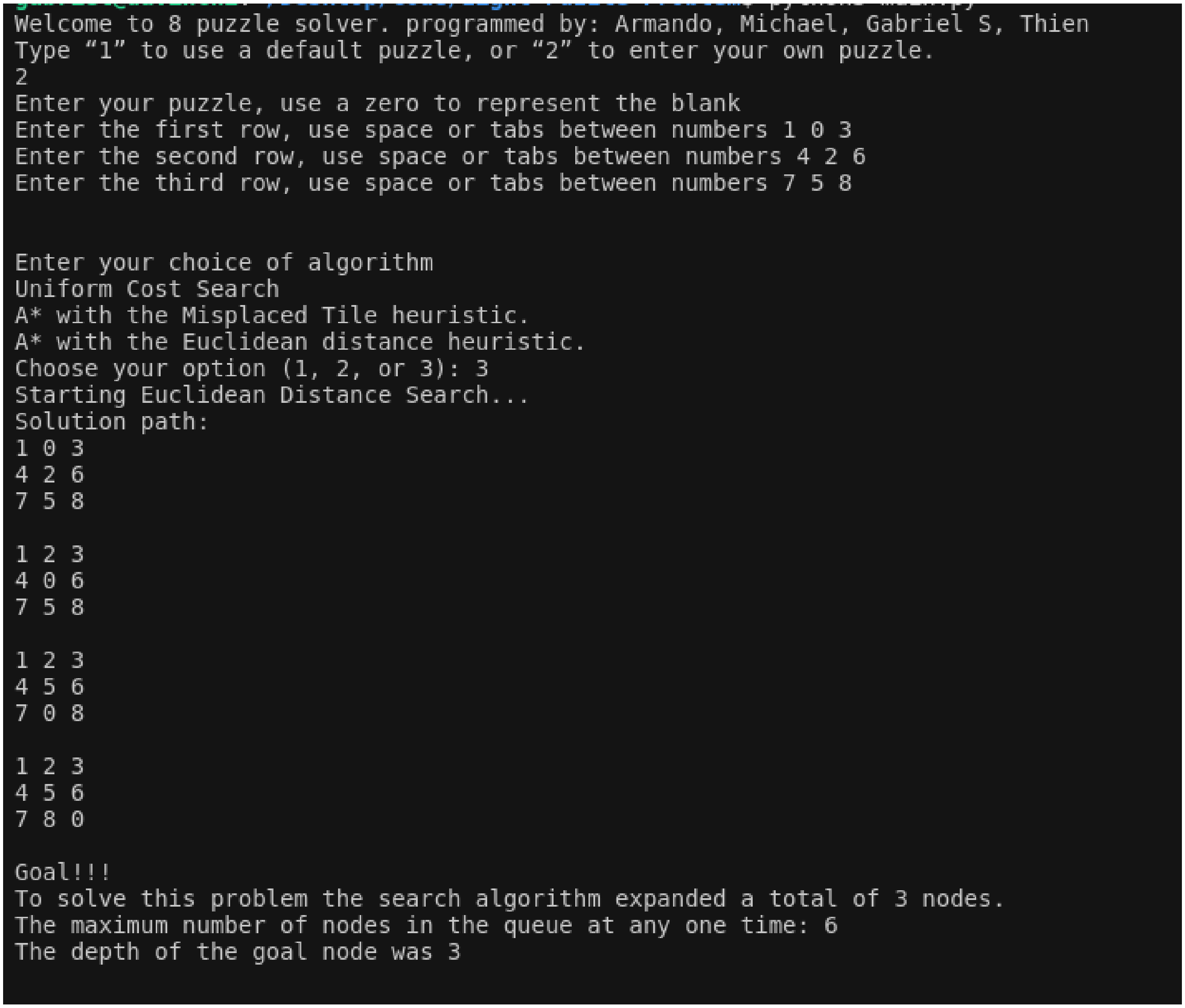
Number of Nodes Expanded



Number of Nodes Expanded

	Uniform Cost Search	Misplaced Tile Heuristic	Euclidean Distance
Puzzle 1	3	1	1
Puzzle 2	3	2	2
Puzzle 3	28	4	4
Puzzle 4	85224	6488	1051
Puzzle 5 (Impossible)	181440	22691	182050

Trace of Euclidean A* algorithm



```
Welcome to 8 puzzle solver. programmed by: Armando, Michael, Gabriel S, Thien
Type “1” to use a default puzzle, or “2” to enter your own puzzle.
2
Enter your puzzle, use a zero to represent the blank
Enter the first row, use space or tabs between numbers 1 0 3
Enter the second row, use space or tabs between numbers 4 2 6
Enter the third row, use space or tabs between numbers 7 5 8


Enter your choice of algorithm
Uniform Cost Search
A* with the Misplaced Tile heuristic.
A* with the Euclidean distance heuristic.
Choose your option (1, 2, or 3): 3
Starting Euclidean Distance Search...
Solution path:
1 0 3
4 2 6
7 5 8

1 2 3
4 0 6
7 5 8

1 2 3
4 5 6
7 0 8

1 2 3
4 5 6
7 8 0

Goal!!!
To solve this problem the search algorithm expanded a total of 3 nodes.
The maximum number of nodes in the queue at any one time: 6
The depth of the goal node was 3
```