

Assignment: 1

General Guidelines

1. This assignment is of 35 Marks and carries weightage for FINAL evaluation of ME 620.
2. Please **upload screenshots / PDF of handwritten pages as a single file to MS Teams under Assignment Tab**. Submission as posts on MS Teams or email submission would not be entertained.
3. You may discuss with other students about this assignment. Ask TAs for clarifications. Consult outside sources such as the Internet and take help to learn the material. Finally, **the solutions you submit should be your own work, not copied** from a peer or an abridged outline from any solution manual.
4. **Note that the assignment has two sections. You are required to do ONLY one section.**

Section A: For People Interested in Pen-Paper Assignment

A1. Finding if the 8-puzzle instance is solvable

Note that the 8-puzzle states are divided into two disjoint sets, such that any state is reachable from any other state in the same set, which no state is reachable from any state in the other set. Before you proceed with solving the 8-puzzle instance using the A* algorithm, it is mandatory to you check (either manually or as part of your 8-puzzle solver implementation) that the instance is solvable.

Definition: For any other configuration besides the goal, whenever a tile with a greater number on it precedes a tile with a smaller number, the two tiles are said to be inverted.

Proposition: For a given puzzle configuration, let N denote the sum of the total number of inversions. Then $(N \bmod 2)$ is invariant under any legal move. In other words, after a legal move an odd N remains odd whereas an even N remains even.

Example

Start State

7	2	4
5	0	6
8	3	1

Goal State

	1	2
3	4	5
6	7	8

Linear Representation 7; 2; 4; 5; 6; 8; 3; 1 - Ignore the 'blank'

Inversions	7	6 Inversions
	2	1 Inversion
	4	2 Inversions
	5	2 Inversions
	6	2 Inversions
	8	2 Inversions
	3	1 Inversion
	1	0 Inversions

Total Inversions 16 – Even

The goal state described above, with no inversions, has $N = 0$, and can only be reached from starting states with even N , not from starting states with odd N . Puzzle is solvable.

Generate random start and goal instances for the 8-puzzle and check for solvability.

A2. Start from the board position for 8-Puzzle as shown below.

1	2	3
6		4
8	7	5

Start

1	2	3
8		4
7	6	5

Goal

Draw the search tree resulting from using A* exploiting

- Misplaced tile heuristic.
- Manhattan distance heuristic.
- Number of tiles not in the correct row plus tiles not in the correct column.

Label the nodes in the order in they are expanded. Mark the f value of each node.

Is the heuristic at no. c above admissible? Why or why not?

Section B: For People Interested in Programming / AI Tools

Experimenting with Heuristic Search using 8-puzzle

- This assignment is based on *experimenting* with some heuristic search algorithms. We shall use the 8-puzzle game.
- For the experimentation, you would need to create 10 random 8-puzzle instances (i.e., define start and *reachable* goal states). Each of these are to be solved using variants of A* algorithm.
- You may write your own code or tailor code available from other sources for the 8-Puzzle; the A* algorithm is to be used with different heuristics and need to have possibility of recording the experimental parameters of interest as detailed below.
- You must cite any kind of help from elsewhere. Each algorithm should be run on the same set of problems for the comparison to be fair.

Once a start and goal is specified, check for solvability as detailed in A1 above.

Implement A* algorithm. The algorithm is expected to record some vital parameters for each of the solved problem. You are to *at least* record a. the total running time in seconds b. the length of the solution and c. the total number of nodes that are *removed* from the frontier.

The variants of A* algorithms you should test over *at least* 10 solvable instances are:

- A* using the misplaced tile heuristic.
- A* using the Manhattan distance heuristic.
- Any other heuristic of your choice – either *novel* or not discussed in the class!

Submission Guidelines

- You need to submit **PROGRAM OUTPUT** for two **SAMPLE SOLUTIONS** with depth of at least 10. For this your program, should output each step from the initial to the final state. This is part of the REPORT below.
- A **short REPORT** detailing discussion on **EVALUATION OF THE HEURISTICS** is to be submitted. The analysis together with justification is to be presented for figuring out the better heuristic for the 8-Puzzle.