Note:
1. Tutorial problems are provided to help you understand the materials discussed in the class, and to improve your skills in solving AI problems.
2. Tutorial problems will not be graded. However, you are highly encouraged to do them for your own learning. Moreover, we hope you get the satisfaction from solving the problems 😊
3. You'll get the best learning outcome when you try to solve these problems on your own first (before your tutorial session), and use your tutorial session to ask about the difficulties you face when trying to solve this problem set.

Fig. 1. Left. UQ map. Right. A simplified UQ map, where the vertices are labeled with building’s numbers/names, and edges represent the cost of moving from one building to another.

Navigation problem. Ever wonder how Google map or the navigation app in your (smart) mobile phone can provide you with a shortest path to go from one place to another? Well, this problem is about developing a core component for such map/navigation app. Suppose you are given a map in the form of a graph with the following format.
1. The file contains |V|+|E|+2 lines, where |V| is the number of vertices and |E| is the number of edges.
2. The first line is the number of vertices. Each line in the next |V| lines is the vertex’ label. The next line is the number of edges. Each line in the last |E| lines is the end-points of the edges and the cost associated with it.
3. Example of the content of the input file for Fig.1.Right.

7
78
50
Union
TaxiRank
Your task is to develop a program to find the shortest path in the graph from a given initial vertex to a given goal vertex. To this end,

1. Please define the action space, state space, and world dynamics of this problem.

2. Please use A* search algorithm to solve the problem.

3. Please write the program.

4. For a very basic test to check if your algorithm is correct or not, please use Fig.1.Right (the above input example) as your graph, and please find the shortest path from building 78 to building 82D.

**The 8-puzzle** (p. 70 RN, 3rd ed). Fig.2 shows an 8-puzzle. It consists of 3X3 board with 8 numbered tiles and a blank space. A tile adjacent to the blank space can slide into the space. The object is to reach a specified goal state, such as the one shown on the right of Fig. 2. Our goal is to develop a program to solve the 8-puzzle problem.

1. Please define the action space, state space, and world dynamics of this problem.

2. Please choose a search algorithm to solve the 8-puzzle program, and explain why you choose the particular algorithm.

3. Please write a program that solves 8-puzzle problem. It accepts a file of the initial state and goal state in the following format:
a. The file consists of two lines, the first line is the initial state while the second line is the goal state.

b. The initial & goal state are written as sequence of numbers on the tiles, from left to right and top to bottom. The numbers are separated by space. Blank is ‘0’

c. Example for the initial & goal as in Fig.2 is a file input.txt which contains:

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

d. Please output the steps to reach the goal from the initial state, and check if your program is correct or not.