Assignment 4

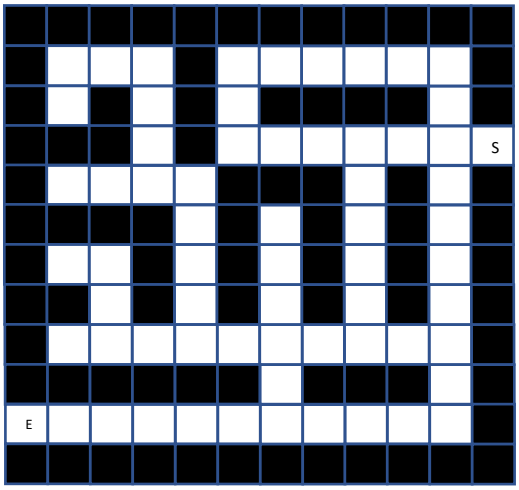
Question (10points). Using DFS and BFS for solving a maze problem. A robot needs to find a path given a start position S and an end position E.

Descriptions:

1. The maze can be represented using an 2d array, e.g.,

```
maze = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
        [0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0],
        [0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0],
        [0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1],
        [0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0],
        [0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0],
        [0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0],
        [0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0],
        [0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0],
        [0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0],
        [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0],
        [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]
```

   0 means the wall, 1 means an empty black can be visited by the robot.



2. In the main.py file, you are required to implement BFS and DFS functions. Please see the requirement for input and output.
3. We will check the results for the given start and end positions by looking at the output.