

**CSC 225 SPRING 2023**  
**ALGORITHMS AND DATA STRUCTURES I**  
**ASSIGNMENT 6 – PROGRAMMING (OPTIONAL)**  
**UNIVERSITY OF VICTORIA**

For some unknown reasons some intersections of the city Richtopia have been blocked. Residents of Richtopia (known as Richtopians) can no longer simply pass the intersections to go to their desired destinations. In fact, it is now much more realistic to look at the city as a maze. Richtopians need an app that can tell them how fast they can get from point A to point B. That's the only way they can schedule at least a small portion of their days. Write the app and save the people!

The city is modeled as a two dimensional table. The objective is to know the minimum number of moves to get from cell A to cell B on the table. In each move, we can either go right, go left, go up, or go down one cell in the table. We cannot make a diagonal move. Also, we can only move to a cell if the cell **is not blocked**, and if we **do not leave** the table.

### Input Format

The first line of the input contains a positive integer  $n$ . In the next lines, the city is described as an  $n \times n$  table. There are  $n$  lines where each line has  $n$  characters. Each character is one of the following:

- 1) A '.' cell represents an intersection that is in good shape and can be used for the commute.
- 2) A '#' cell represents a blocked intersection that cannot be used.
- 3) An 'A' shows the starting point.
- 4) A 'B' shows the destination.

### Constraints

- 1)  $1 < n \leq 1000$
- 2) There is exactly one 'A' cell and one 'B' cell in the input.

### Output Format

If it is possible to reach from cell 'A' to cell 'B', print the minimum number of moves required to do so. If not, print the word "IMPOSSIBLE".

### Sample Input 0

```
5
.A...
.....
.....
.####
...B.
```

### Sample Output 0

8

### Sample Input 1

```
5
.A...
.....
.....
.####
#..B.
```

### Sample Output 1

IMPOSSIBLE

### Sample Input 2

```
7
.B.....
#####.
.....
.#####
.....
#####.
A.....
```

### Sample Output 2

29

### Sample Input 3

```
7
.....
.....
A.....
..#....
.#B#...
..#....
.....
```

### Sample Output 3

```
IMPOSSIBLE
```

### Submission

You must solve this problem using Java. You will read from standard input and print to standard output using the template provided, `wayFinder.java`.

### Evaluation Criteria

The programming assignment will be marked out of 20, based on a combination of automated testing and human inspection. I will only include your 5 best assignment grades in your final percentage. The following score ranges will apply to this assignment:

Score	Description
0 – 5	Submission does not compile.
5 – 10	Compiles but is incorrect in at least one test.
10 – 15	Correctly reports the solution but does inefficiently.
15 – 20	Correctly and efficiently reports the solution.