DAILY DSA | DAY-17 | Queues - Practise|
-GOPALKRISHNA A

Let's solve the shortest path using a breadth-first search algorithm (Queues)!

Problem:
In a given grid of 0s and 1s, we have some starting rows and columns sr, sc and a target row and column tr, tc. Return the length of the shortest path from sr, sc to tr, tc that walks along 1 value only.

Each location in the path, including the start and the end, must be a 1. Each subsequent location in the path must be 4-directionally adjacent to the previous location.

It is guaranteed that grid[sr][sc] = grid[tr][tc] = 1, and the starting and target positions are different.

If the **task is impossible, return -1.**

Example:

```
input:
grid = [[1, 1, 1, 1], [0, 0, 0, 1], [1, 1, 1, 1]]
sr = 0, sc = 0, tr = 2, tc = 0
output: 8
(The lines below represent this grid:)
1111
0001
1111

grid = [[1, 1, 1, 1], [0, 0, 0, 1], [1, 0, 1, 1]]
sr = 0, sc = 0, tr = 2, tc = 0
output: -1
(The lines below represent this grid:)
1111
0001
1011
```