Problem 2: Galactic Vacation (10 points) [CodeForces]

By Pedro Paredes

Problem Statement

As a Princeton student in the year 3000, you and your friends decide to spend your winter break traveling among the galaxy in a rental spaceship. There are so many places you could travel between, and so before you set off on your journey, you want to do some accounting for fuel to ensure you can perform all legs of your vacation.

Your current vacation plan has many legs, each of which consist of a beginning and ending point in space. You can view the galaxy as an N by N grid of tiles with $1 \leq N \leq 40$ in which your ship can move between vertically or horizontally adjacent tiles with 1 unit of fuel. Each tile is either occupied by a planet (blocked), free for your ship to move into (open), or part of a wormhole. Each wormhole tile has a sister wormhole tile such that entering the wormhole from one tile will immediately transport you to the other tile. You may not use the same wormhole in two consecutive moves (i.e., you may not travel forwards then backwards).

Your vacation has L legs with $1 \leq L \leq 10^5$. For each leg of the journey, compute the minimum amount of fuel units required to complete this leg of the journey. Due to occupied tiles, many journeys may not be possible, if so return -1.

It is guaranteed there is at least one free tile.

Input/Output Details

Input

The first line contains positive integers N, B, W, and L. The next B lines have two space-separated integers, a and b, with $1 \leq a, b \leq N$, which denote the coordinates of an occupied tile. The following W lines consist of four space-separated integers, x_1, y_1, x_2, y_2 with $1 \leq x_1, y_1, x_2, y_2 \leq N$, denoting tiles (x_1, y_1) and (x_2, y_2) as sister wormhole tiles.

The following L lines consist of four space-separated integers s_x, s_y, e_x, e_y with $1 \leq s_x, s_y, e_x, e_y \leq N$ that correspond to a leg of the trip starting at tile (s_x, s_y) and ending at tile (e_x, e_y).

Output
The output should consist of L lines each with one integer, with the integer on the ith line corresponding to the minimum fuel needed to fly the ith leg of the journey or -1 if that journey is not possible.

Example 1

Input:

```
3 3 0 2
2 2
3 2
3 3
3 1 2 3
1 3 1 1
```

Output:

```
5
2
```

Explanation: For both test cases, there is a unique path from start to end vertex and this provides the minimal distance, which is 5 and 2 respectively. The below diagram showcases the results for the first leg, with the yellow squares indicates blocked tiles, the green and red circles indicating the begin and end tiles, and the blue line showing the minimum-fuel path.

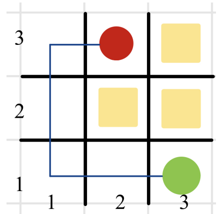

Figure 1: Leg 1 for Test Case 1

Example 2

Input:

```
5 7 1 3
1 1
2 2
3 3
3 4
3 5
4 4
5 5
2 1 2 4
5 4 1 2
2 3 4 5
4 2 4 2
```

Output:

```
9
-1
0
```

Explanation: For the first leg, the beginning and end tiles are separated by occupied tiles and thus we must take the portal. It takes 6 movements to reach the portal at $(2,1)$. From the second portal it takes 3 movements to reach the end, leading to a total fuel count of $6+3=9$. For the second leg, the tile $(4,5)$ is entirely blocked in and so unreachable.