ACE Coursework 2

This is the second assessed coursework. It is worth 20% of the final mark. Please submit Java code, a `read me' file (optional) and a 2-3 pages report describing the problem-solving process, including selection of data structures, algorithm design, algorithm correctness justification and efficiency analysis. The submission deadline is **4 pm, 12th of December 2023**.

Question 1 [10 marks]

Suppose there is a maze in the form of an $n \times m$ grid (n columns and m rows). Every cell contains a positive integer. Two cells are *adjacent* if they share a common wall. From a cell c_1, a person spends a time t walking to a cell c_2 that is *adjacent* to c_1, where t is equal to one plus the cardinality of the difference between the integers contained in c_1 and c_2. A person wants to walk from a `start cell' $(s_1, s_2)$ to a `destination cell' (f_1, f_2), where $1 \leq s_1 \leq n$, $1 \leq s_2 \leq m$, $1 \leq f_1 \leq n$, $1 \leq f_2 \leq m$, and both cells are on the boundary of the grid. Write a program which calculates the shortest time that a person needs to spend to walk from a start cell to a destination cell.

Input

The first line includes two positive integers n and m indicating the number of columns and the number of rows of the grid, respectively. The integers n and m are separated by a single space. The second line includes two pairs of integers (s_1, s_2) and (f_1, f_2) indicating the positions of the start and the finish, respectively, where $s_1, f_1 \in [1, n]$, $s_2, f_2 \in [1, m]$, and the positions are on the boundary of the grid. The pairs (s_1, s_2) and (f_1, f_2) are separated by a single space.
The $i + 2$-th line $(1 \leq i \leq m)$ contains n positive integers separated by single spaces.

Output

The output is the shortest time that a person needs to spend to walk from the start cell to the destination cell.

Example

Standard input	Standard output
6 3 (1, 1) (6, 3) 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	7
6 3 (1, 1) (6, 3) 4 1 2 3 1 1 5 3 1 1 4 1 3 2 2 4 1 2	15

Question 2 [10 marks]

Suppose there is an area in the form of an $n \times m$ grid (n columns and m rows). Within this area, a person starts from a `start cell' $(s_1, s_2)$ and walks to a `destination cell' (e_1, e_2). The person walks following the rules below, where k is a non-negative integer.

1. At each step, the person can walk from one space cell to another space cell such that they have a common wall.
2. At each step, the person can walk from one space cell to another space cell such that they have a common point.
3. If k is greater than zero, then the person can jump horizontally or vertically from one space cell to another space cell, such that there are at most k block cells between them.
4. If k is greater than zero, then the person can jump diagonally from one space cell to another space cell, such that there are at most k block cells between them.

Write a Java program which takes an $n \times m$ grid, a start cell (s_1, s_2), a destination cell (e_1, e_2), and a non-negative integer k as input, outputs the minimal number of steps taken from the start cell to the end cell. A jump is counted as a single step. Once the value of k is specified in the input, it is fixed in the whole walking process of the person.

Input

The first line includes two positive integers n and m indicating the number of columns and the number of rows of the grid, respectively. The integers n and m are separated by a single space.
The second line includes two pairs of integers (s_1, s_2) and (e_1, e_2) indicating the positions of the start and destination cells, respectively, where $s_1, e_1 \in [1, n], s_2, e_2 \in [1, m]$. The pairs (s_1, s_2) and (e_1, e_2) are separated by a single space.
The third line contains a non-negative integer k.
The $i + 3$-th line $(1 \leq i \leq m)$ contains n integers separated by single spaces. Each integer is one of 0 and 1, indicating a space cell and a block cell, respectively.

Output

Output the minimal number of steps taken from the start cell to the end cell, if the end cell is reachable from the start cell; otherwise, return -1.

Example

Standard input	Standard output
6 3 (3, 1) (5, 1) 0 0 1 0 1 0 1 0 0 0 0 0 1 0 1 0 0 0 0	2
6 3 (3, 3) (5, 1) 1 0 1 0 1 0 1 0 0 0 1 0 1 0 1 0 0 0 0	1

Reference and Plagiarism

If you use code you found in a textbook or on the web, you must acknowledge it. We will run the plagiarism detector tools to check for similarities between submissions and web-based material. You are reminded of the School's Policy on Plagiarism. **Never do "copy and paste" in your coding, that will be easily detected by tools as plagiarism. Similar to academic writing, in addition to providing a citation, you still need to rephrase it**.

How to submit

Online submission via Moodle. Please make sure that all the java files needed to compile your program are included. Please note that every next submission overwrites all the files in the previous one, so if you submit several times, make sure that your last submission includes all the necessary files.