
MUV LUV ALTERNATIVE

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

In order to cover *the Tactical Armored Squadron* withdrawing, *the Combined Fleet's 2nd Squadron* is attracting attacks from *Lux*. After a while, the fleet is heavily damaged and some ships have been on fire. The soldiers on one fired ship are trying escaping from the cabin and reaching the board.

The cabin can be abstracted into a grid of $n \times m$, where rows are numbered $1, 2, \dots, n$ from bottom to top and columns are numbered $1, 2, \dots, m$ from left to right. The cabin is divided into 3 seated zones by 2 vertical corridors of width 1, where the left zone is of width l_1 , the middle zone is of width l_2 and the right zone is of width l_3 . Also, for the two corridors, one is between the left zone and the middle zone and the other one is between the middle zone and the right zone so that $l_1 + l_2 + l_3 + 2 = m$ always holds. Let's use (x, y) to denote the location of the cell in the x -th row and the y -th column. Below the two corridors are the left exit and the right exit, which can be considered to be positioned at $(0, l_1 + 1), (0, l_1 + l_2 + 2)$ respectively. Following is the illustration when $n = 4, l_1 = l_2 = l_3 = 2$.

column row	1	2	3	4	5	6	7	8
4	seated zone 1 (left zone)		left corridor	seated zone 2 (middle zone)		right corridor	seated zone 3 (right zone)	
3								
2								
1								
0			left exit			right exit		

There are k soldiers inside the cabin, each soldier is in a unique cell (x_i, y_i) in one of the three seated zones initially. A soldier in seated zones can only move leftward or rightward, while one in corridors can move not only leftward or rightward, but also upward or downward. And one soldier can stay still or move to an adjacent cell in one of the allowed directions according to its current position at every moment. Two cells are adjacent if and only if they share an edge. Each cell mustn't contain more than one soldier after each moment's movement.

For maintaining the order of escape, the soldiers who are initially in the left zone can only go to the left exit, while ones who are initially in the right zone can only go to the right exit. For ones who are initially in the middle zone, they can go to either the left exit or the right exit.

You want to know the minimum possible time that all the soldiers have reached one of the two exits.

Input

The first line contains five positive integers n, l_1, l_2, l_3, k ($1 \leq k \leq 100\,000, 1 \leq n, l_1, l_2, l_3 \leq 10^9$), denoting the number of rows, the three widths of the left zone, the middle zone, the right zone, and the number of soldiers respectively.

Next k lines each contains three positive integers a_i, x_i, y_i ($a_i \in \{1, 2, 3\}, 1 \leq x_i \leq n, 1 \leq y_i \leq l_1 + l_2 + l_3 + 2, y_i \neq l_1 + 1, y_i \neq l_1 + l_2 + 2$), denoting the initial position of a soldier, where a_i denotes the zone index and x_i, y_i denotes the row index and the column index respectively.

It is guaranteed that input k positions are pairwise distinct and that (x_i, y_i) is indeed in the zone a_i .

Output

Output a single line containing a positive integer, denoting the minimum possible time.

Example

standard input	standard output
4 2 2 2 3 1 2 1 1 2 2 2 2 4	4