The 43rd Annual ACM

International Collegiate Programming Contest Asia Regional – Seoul Nationwide Internet Competition



Problem J Sliding Blocks

Time Limit: 1 Second

When rectangular blocks, which are all parallel to the xy-axis, are given one by one in the plane, each block moves along the specified direction from the specified starting location. The movement of a block consists of two phases. For the first phase, each block moves from the specified location along the given direction which is always left-downward. While moving if it hits either the x-axis, the y-axis, or another block it changes its direction and enters the second phase. More specifically, if the bottom part of the moving block hits the x-axis or another block its moving direction changes leftward and it enters the second phase. Similarly, if the left part of the moving block hits the y-axis or another block its moving direction changes downward and it enters the second phase.

The second phase movement consists of repeating 'move-left' and 'move-down' (or 'move-down' and 'move-left') as long as the block can move. Once the block enters the second phase, it keeps moving in the same direction as farther as possible. If it is blocked either by the xy-axis or by another block it changes its moving direction and continues moving.

A block's location is denoted by the (x, y) coordinate of its lower-left corner point. Figure J.1 shows an example how a block moves. The size of each block is shown in the figure. Suppose seven blocks (A to G, all are colored green) are already placed and block H has the turn to move. The dotted rectangle indicates the starting and the intermediate locations of block H (orange block at final position). The red lines indicate the moving path. As Figure J.1 shows it first moves left-downward (in the first phase) until it hits block C. Then the direction changes leftward, which means it enters the second phase of movement, and continues moving as farther as possible until it hits block F. Then it continues moving in the second phase by changing directions until it can no longer move.

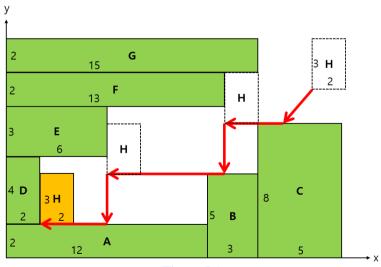
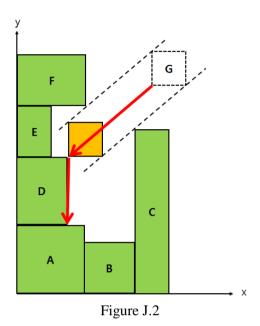


Figure J.1

How to treat some special cases will be explained with an example as shown in Figure J.2. While a block, say G, is moving left-downward in the first phase, if the lower left corner point of it hits another block, say block D, at the upper right corner point of D, G may have two choices of changing direction either downward or

leftward. In such case, moving downward has higher priority than moving leftward. If G, in such case, is unable to move downward due to some other blocks, then its direction changes leftward. While block G is moving left-downward its upper-left corner point (or its lower-right corner point) may hit another block's corner points as shown in Figure J.2 (Note that block F and block C hit block G at their corner points during the first phase movement.) In such case, block G continues moving in the same direction since it has not been blocked by the corner points.



The movement rules can be summarized as follows:

- 1. Each block begins move at the given location along the given direction. The direction is always left-downward.
- 2. A moving block changes its direction when it is blocked either by the x-axis, by the y-axis, or by another block.
- 3. The moving block keeps moving by changing directions until it can no longer move.

For each block, following information is given as shown in <Fig. 3>.

- 1. The width and height w, h of the block.
- 2. The starting location (x, y) at which the block begins move.
- 3. $(\Delta x, \Delta y)$ to indicate the direction of the first movement (see the red line in Figure J.3). Each block begins move from location (x, y) toward $(x \Delta x, y \Delta y)$.

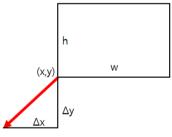


Figure J.3

Given information on n rectangular blocks, write a program which finds the location of the last block after moving every block one by one according to the movement rules.

You may assume that the $i^{th}(2 \le i \le n)$ block at its the starting location is overlapped with none of the previous blocks (i.e., blocks of indices between 1 and (i-1)) and that some block may not move at the very starting location since it is blocked by other blocks.

Input

Your program is to read from standard input. The input starts with a line containing an integer, $n \ (1 \le n \le 1,000)$, where n is the number of blocks. Each of the following n line contains information on each block. Each block information consists of 6 integers. The first two integers w, $h \ (1 \le w, h \le 100,000)$ indicate the size of it. Next two integers $(x, y) \ (0 \le x, y \le 10^8)$ indicate the starting location of it. Next two integers $(\Delta x, \Delta y) \ (1 \le \Delta x, \Delta y \le 10^5)$ represent the direction for the first movement as explained above.

Output

Your program is to write to standard output. Print exactly one line which contains two integers x and y, where (x, y) is the location of the last block after moving every block one by one according to the movement rules.

The following shows sample input and output for two test cases. (Note that the first sample corresponds to Figure J.1 and the second to Figure J.2)

Sample Input 1

Output for the S	Sample Input 1
------------------	----------------

8	2 2
12 2 100 100 1 1	
3 5 100 1 1 1	
5 8 100 1 1 1	
2 4 0 100 1 1	
6 3 0 100 1 1	
13 2 1 100 100 1	
15 2 1 100 100 1	
2 3 18 10 1 1	

Sample Input 2

Output for the Sample Input 2

7	3 4
4 4 1000 1 1 1	
3 3 1000 1 1 1	
2 10 1000 1 1 1	
3 4 1 100 1 1	
2 3 0 100 1 1	
4 3 0 100 1 1	
2 2 10 15 1 1	