

4. Catapult-Carousel (kk)

1 sec / 3 sec

50 points

Catapult-carousel is a new experimental fairground attraction that John invented: a rotating wheel where N seats are spaced at equal distances. The seats are numbered $1 \dots N$ in clockwise order. Each seat is equipped with a mechanism that can launch the rider up the air in a controlled manner—so that they land on a predetermined seat.

An (X, Y) -jump is the mode of operation of the carousel where all riders will be launched simultaneously in such a way that the rider launched from seat 1 will land on seat X and each subsequent rider (from seats $2 \dots N$) will land on a seat Y positions clockwise from the previous landing position. John has designed the carousel in a way that no two riders will ever land on the same seat (for any allowed jump mode).

One ride on the carousel comprises of several jumps. The number and parameters of jumps are the same in all rides. The riders do not change seats between jumps, even when a rider takes several rides in a sequence. John is selling multiple-ride tickets that specify the initial seat and the number of rides. Now he wants to check that the rider is in the expected seat after completing all the rides.

Write a program that gets the seat number U of a rider and computes their seat after K rides (or the seat they were at $|K|$ rides ago, in case K is negative).

Input. The first line contains N , the number of seats on the carousel ($2 \leq N < 2^{63}$), and S , the number of jumps in a ride ($1 \leq S \leq 10^6$).

Each of the following S lines specifies the parameters X_i and Y_i of one jump ($0 < X_i \leq N$, $0 < Y_i < N$, where $1 \leq i \leq S$).

The last line contains U , the number of the seat in question ($1 \leq U \leq N$), and K , the number of rides ($-2^{63} \leq K < 2^{63}$).

Output. The only line should contain one integer: the number of the seat the rider will be on after K rides (or was on $|K|$ rides ago, if K is negative).

Example.	Input	Output
	11 2	1
	3 2	
	5 7	
	4 2	

Let's number the riders according to their initial seats:

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11].

After the first jump of the first ride (where $X = 3$, $Y = 2$), they land as follows:

[11, 6, 1, 7, 2, 8, 3, 9, 4, 10, 5].

After the second jump of the first ride ($X = 5$, $Y = 7$):

[6, 10, 3, 7, 11, 4, 8, 1, 5, 9, 2].

After the first jump of the second ride ($X = 3$, $Y = 2$):

[2, 4, 6, 8, 10, 1, 3, 5, 7, 9, 11].

After the second jump of the second ride ($X = 5$, $Y = 7$):

[4, 9, 3, 8, 2, 7, 1, 6, 11, 5, 10].

The answer is 1, as the rider who started from seat 4 ends up on seat 1 after 2 rides.

Example.	Input	Output
	911 4	825
	3 2	
	5 7	
	133 22	
	11 12	
	1 -1	

Grading. In this task, test cases are divided into groups. A solution scores points for a group only if it solves correctly all test cases in the group. The following additional constraints apply in groups:

1. (10 points) $N \leq 100$, $1 \leq K \leq 20$, $S \leq 100$.
2. (10 points) $1 \leq K \cdot S \leq 10^6$.
3. (10 points) $1 \leq K \leq 10^7$.
4. (20 points) No additional constraints.