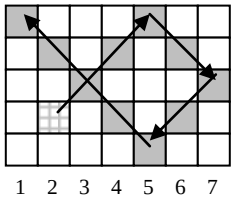
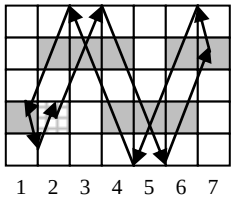
Reflection
Input File: ReflectioinIn.txt

A video game is being played on a monitor that contains **`w`** pixels in the horizontal (x) direction and **`h`** pixels in the vertical (y) direction. When the game begins, a ball is launched from a given pixel location at a given velocity. When the ball reaches an edge of the screen, it is reflected in a perfect elastic collision, causing its x (horizontal) velocity component to change sign if it hits a vertical edge, or its y (vertical) velocity component to change sign if it hits a horizontal edge..

For example, if the ball was traveling from the lower left portion of the monitor to the upper right and its x and y component velocities were +1 and +3 pixels per second respectively, after it collided with the top of the monitor its new x and y velocities would be +1 and -3.

At each tick of the system clock, the ball is relocated to its new pixel location. For example, if the monitor was comprised of 7 x 5 pixels, and the ball was launched from (2,2) at an x and y velocity of +1 and +1 respectively, then the black squares in the monitor depicted on the left would be the pixel positions of the ball after the first 11 ticks of the clock. However, if the ball's initial y velocity was increased to +2, its path would be as depicted on the monitor on the right.





Your task is to determine the number of ticks of the clock from launch to the first time the ball returns to its launch position *and* launch direction. You may assume the ball is never launched from a pixel on the edge of the screen (i.e., leftmost or rightmost column, or top or bottom row) and that a collision with a corner changes the sign of both velocity components..

Inputs:
The first line of input contains the number of games to consider. This will be followed by two lines of input per game. The first of these lines will contain two integers that are the pixel width, **`w,`** and pixel height, **`h,`** of the monitor. The second line will contain four integers: the pixel location from which the ball will be launched, x followed by y, followed by the x and y launch velocity of the ball. All numbers will be separated by a single space.

Output:
There will be one line of output per game that contains the number of ticks of the clock from launch to the first time the ball returns to its launch position and launch direction.

Sample Input

```
3
100 8
10 5 0 1
7 5
2 2 1 1
7 5
2 2 1 2
```

Sample output

```
14
24
12
```