12453 Simulating Billiard Balls

In an $(s \times s)$ shaped billiard board there is a ball at location A (x_1, y_1). There is another point on the board B (x_2, y_2). The location of A and B are never same. The ball is hit with a strictly positive velocity v in such a way so that it passes through B before hitting any wall. The ball then hits the walls several times (it can be even millions) and again reach point A. Minimum how many times will the ball have to hit the walls (horizontal or vertical or both at the same time) before it comes back to the initial location A? Assume that the ball is a point, and the collisions with a side are elastic (no energy loss) and thus the absolute value of a velocity component of the ball parallel to each side remains unchanged after each bounce and the ball rolls along forever! Also assume that the coordinate of the lower left corner of the billiard board is $(0, 0)$ and the coordinate of the upper right corner of the billiard board is (s, s). When the ball hits a corner it actually touches both the horizontal and vertical wall at the same time (So it is considered as two hits) and the balls direction is rotated by 180 degree.

For example in the figure below you can see a (5×5) billiard table. A ball is at location $(2, 3)$. It is hit in such a way that passes through the location $(4, 4)$ and then starts hitting the walls. After hitting the walls total 6 times the ball again comes back to point $(2, 3)$.

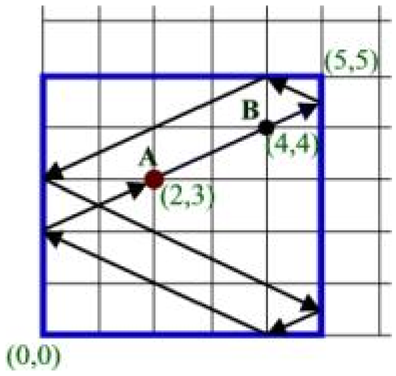

Fig: A billiard ball in a 5×5 board

Input

Input file contains around 50000 lines of inputs. Each line contains five integers s $(1 < s < 1000000000)$, x_1 $(0 < x_1 < s)$, y_1 $(0 < y_1 < s)$, x_2 $(0 < x_2 < s)$, y_2 $(0 < y_2 < s)$. Input is terminated by a line containing five zeroes. This case should not be processed.

Output

For each line of input produce one line of output. This line contains the serial of output followed by an integer. This integer denotes minimum number of times the ball must hit the wall (horizontal or vertical) before reaching point A again.

Sample Input

```
5 2 3 4 4
4 2 2 3 3
0 0 0 0 0
```

Sample Output

```
Case 1: 6
Case 2: 2
```