CCC '05 S2 - Mouse Move

Most likely, you will notice that you have a mouse attached to your computer, which lets you move the cursor around the screen. Your job is to get between the mouse and the cursor.

Suppose that the bottom left-hand corner of your screen is $(0, 0)$, and all points on the screen are given by integer co-ordinates (x, y) where $0 \leq x \leq c$ and $0 \leq y \leq r$. Thus, the top-right corner of the screen is at position (c, r), bottom-right corner is $(c, 0)$, and top-left corner is $(0, r)$.

When a mouse is moved, it sends a pair of integers (a, b), indicating that the cursor should be moved a units in the x-direction and b units in the y-direction. It is worth noting that this is relative motion (i.e., how far to move) rather than absolute motion (i.e., where to move). It is also worth noting that a and b may be positive, negative or zero.

You can assume the mouse starts at position $(0, 0)$. Your job is to read input messages (i.e., relative motion positions sent by the mouse) and update the cursor to the new position on the screen. Your output (to the screen) will be the position of the mouse after each move.

If the mouse hits the screen boundary, it stops moving in that direction. For example, if the mouse is supposed to move $(-100, -10)$ from its current position $(30, 40)$, the final positions will be $(0, 30)$: the mouse will hit the left-hand side boundary, but still manages to move down.

Input is listed as pairs, the first pair being (c, r), followed by the relative motion pairs (x, y). The input is terminated when the mouse moves $(0, 0)$.

Sample Input 1

```
100 200
10 40
-5 15
30 -30
0 0
```

Sample Output 1

```
10 40
5 55
35 25
```

Sample Input 2

```
30 40
30 40
-100 -10
0 0
```

Sample Output 2

```
30 40
0 30
```