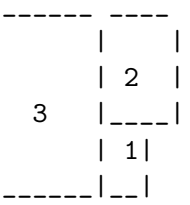
C - Magic Boxes

Input: `standard input`
Output: `standard output`

A certain company has recently introduced a whole line of "magic" boxes. The smallest of these boxes is a 1×1×1 hollow cube. The next size up is a 2×2×2 hollow cube, and so on, each box 1 unit bigger than the previous in all dimensions. However, because the boxes are so fragile, they can be packed neither inside of each other, nor on top of each other.

You are to determine how large a set can fit in a certain sized crate. You are told that there is no restriction on the height of the crate, and the boxes may be packed tightly, with no space between them. Your task is, given x and y, representing the dimensions of the floor of the crate, determine the maximum number of boxes that can be safely packed, given that the sizes of the boxes start at 1×1×1 and increase by 1 with each box (no skipping boxes, if you ship a 3×3×3 box, you must also ship a 2×2×2 and a 1×1×1 box).

For example, a 3×5 crate could hold a 3×3×3, a 2×2×2 and a 1×1×1 box with this configuration (as viewed from above):



Input

The problem input consists of several cases, each one defined in a line that contains two integer values x, and y ($1 \leq x, y \leq 30$), representing the dimensions of the floor of the crate.

The end of the input is specified by a line with the string "`0 0`".

Output

For each case in the input, print one line with the the maximum number of boxes that can be safely packed, given that the sizes of the boxes start at 1×1×1 and increase by 1 with each box (no skipping boxes, if you ship a 3×3×3 box, you must also ship a 2×2×2 and a 1×1×1 box).

Sample Input

```
1 1
2 2
10 10
26 26
0 0
```

Sample Output

```
1
1
5
11
```