
Interactive Smiley Face

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Igor likes smiley faces a lot. He wrote a program that generates really large pictures of white and black pixels with smiley faces. Depending on Igor's mood, a picture can be a happy or a sad smiley. In the sequel white pixels are represented with 0's, and black pixels are represented with 1's.

Igor's program behaves as follows. First, a basic 41×41 rectangular pattern is fixed (see the pattern below). Then, an odd integer $k \geq 1$ and the pattern is magnified k times. That is, the result will be a $(41 \cdot k) \times (41 \cdot k)$ rectangle, with each $k \times k$ block having the same color as the corresponding pixel of the basic pattern.

After that, a happy smile or a sad smile consisting of white pixels is put on the face (see basic patterns for smiles below). The program can choose an integer $m \geq 1$ and magnify the smile m times (magnification behaves the same way as with the face pattern). The magnified smile is then superimposed over the face in such a way that:

- All pixels of the smile should lie strictly below than the center pixel of the face (note that since $41 \cdot k$ is odd, center is defined unambiguously).
- No white pixel of the smile is put over a previously white pixel of the face pattern.
- Moreover, the border of the face should stay untouched by pixels of the smile. By border of the face we mean extreme black pixels in a row a column.

After this step, the resulting picture is placed on a $10^9 \times 10^9$ rectangular grid of initially white pixels. All pixels of the smiley face should be inside the grid. Pixels of the grid have usual integer Cartesian coordinates (x, y) ranging from 1 to 10^9 . Note that the $0x$ axis is pointed rightwards, and $0y$ axis is pointed upwards, that is, the bottom pixels of the grid (and, therefore, the smiley face) have lower y coordinate.

Igor challenged you to guess the mood of his smiley face. Since the grid is very large and cannot be sent to you directly, Igor implemented a system to answer your queries. A query is a rectangle with integer coordinates of corners; the answer to the query is the number of black pixels inside the rectangle.

Your task is to guess the mood of the smiley face by asking at most 500 queries.

Interaction Protocol

Initially your program receives no input.

To ask a query, your program should print four integers x_1, y_1, x_2, y_2 ($1 \leq x_1 \leq x_2 \leq 10^9$, $1 \leq y_1 \leq y_2 \leq 10^9$) on a separate line — coordinates of the bottom left and top right corners of the rectangle respectively. Do not forget to flush your output after each query.

After each query your program is fed one integer — the number of black pixels within the rectangle.

To ask a query, your program should print four integers x_1, y_1, x_2, y_2 ($1 \leq x_1 \leq x_2 \leq 10^9$, $1 \leq y_1 \leq y_2 \leq 10^9$) on a separate line — coordinates of the bottom left and top right corners of the rectangle respectively. Do not forget to flush your output after each query.

;

```

      0 0 0 0
    0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Page 3 of 7

[illegible]

;

;