Geometry Hacking

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

Jeroen made a new algorithm for checking if a point is inside a polygon:

Given a simple polygon $\mathbf{p_0}, \mathbf{p_1} \ldots \mathbf{p_{n-1}}$ and a query point $\mathbf{q}$, his algorithm does the following:

- Initialize ans := 0
- Draw a ray from $\mathbf{q}$ in the positive x-direction.
- For all $0 \le i < n$: Check if the closed segment $\mathbf{p_i p_{(i+1) \bmod n}}$ intersects the ray, if so, increment ans.
- For all $0 \le i < n$ if $\mathbf{p_i}$ lies on the ray, add 1 to ans.
- If ans is odd, the algorithm says inside, otherwise outside.

He is convinced his algorithm is correct, but you have to convince him otherwise. To humiliate him, you are looking for polygons where the point $(0, 0)$ is strictly inside the polygon (so not on the boundary), but his algorithm will say that the point is outside. Consider all simple polygons with vertices on integer coordinates for which this is true, and sort this infinite list by area, breaking ties arbitrarily. Given an integer $k \le 10^3$ in the input, output the k first polygons in this list. If multiple possible answers exist, you can print any of them.

Note that two polygons $\mathbf{a}$ and $\mathbf{b}$ are considered the same polygon if the point list of $\mathbf{a}$ is a cyclic rotation of the points of $\mathbf{b}$. Two polygons are also considered the same, if the reversed point list of $\mathbf{a}$ is a cyclic rotation of $\mathbf{b}$. So in these cases, only one of $\mathbf{a}$ and $\mathbf{b}$ is included in the infinite list. Note that although placing an extra integer point on one of the edges of a polygon produces the same shape, this will be counted as a different polygon.

Input

The only line of input contains the integer k $(1 \le k \le 1000)$ — the number of polygons you should output.

Output

For the first k polygons for which Jeroen's algorithm fails, in sorted order of area, output a description:

On the first line of the description, output n, the number of vertices of the polygon.

On each of the next n lines, output 2 integers, x and y ($|x|, |y| \le 10^9$), the coordinates of the vertices of the polygon.

Under the constraints of the problem, it can be proven that if the smallest areas of the first k valid polygons are $A_1 \le A_2 \le \cdots \le A_k$, then ensuring the additional constraint that the coordinates do not exceed 10^9 in absolute value, does not change this list of areas.

Note that polygons in the output have to be simple polygons.

Example

standard input	standard output
2	4 -1 0 0 1 1 0 0 -1 3 0 -4 3 5 -3 5

Note

The output for the sample input is actually wrong. It is only to show the output format. These two polygons both have the point $(0, 0)$ inside, but Jeroen's algorithm does not fail, and their areas are not guaranteed to be the smallest possible 2 areas.