Problem A. LaLa and Magic Circle (LiLi Version)

Input file:	`standard input`
Output file:	`standard output`
Time limit:	10 seconds
Memory limit:	1024 megabytes

This is an output-only problem.

LaLa has a pile of magic circles in her laboratory.

A magic circle can be represented as a simple polygon drawn with special ink, and is **usable** if it is convex. i.e. all of its internal angles are equal or less than π.

LaLa plans to turn every magic circle into a usable one. However, they may lose all their magical power if done incorrectly. Thankfully, LaLa has the perfect magical tool for that.

The tool works as follows. When you toss in a magic circle, if the circle is usable, the tool reports that it is. Otherwise, it takes two distinct points u and v such that

- u and v lie on the boundary of the convex hull of the magic circle, and
- none of the points on the path from u to v through the boundary of the magic circle in counterclockwise order lie on the boundary of the convex hull of the magic circle, except for u and v.

Then, it rotates the u-v path by π around the midpoint of u and v. In other words, for each point w on the u-v path, w becomes $u + v - w$ where the addition is done coordinate-wise over the two dimensional coordinate system over the paper the magic circle is drawn on. Note that the result of this modification is also a simple polygon.

Little did LaLa know, LaLa's sister, LiLi, overheard LaLa's plan. Knowing how lazy LaLa is, as a prank, LiLi will sneak in a magic circle that takes large amount of applications of the tool to make it usable. More specifically, LiLi will add a magic circle to the pile which is a union of equal or less than 1 000 line segments and the tool can perform some sequence of modifications with between 120 000 and 1 000 000 steps that turns it circle into a usable magic circle.

Write a program to help LiLi compute one such magic circle.

Output

The output should be in the following format:

$$
\begin{array}{cccc}
N & & & \\
x_0 & y_0 & & \\
x_1 & y_1 & & \\
 & \vdots & & \\
x_{N-1} & y_{N-1} & & \\
Q & & & \\
a_0 & b_0 & c_0 & d_0 \\
a_1 & b_1 & c_1 & d_1 \\
 & & \vdots & \\
a_{Q-1} & b_{Q-1} & c_{Q-1} & d_{Q-1}
\end{array}
$$

where the initial magic circle is the union of N line segments connecting points (x_i, y_i) and $(x_{(i+1 \bmod N)}, y_{(i+1 \bmod N)})$ for all integers $0 \le i < N$, and it has a sequence of modifications by the tool of length Q such that i-th modification choose the counterclockwise path from point (a_i, b_i) to (c_i, d_i).

The output should satisfy the following constraints:

- All the numbers in the output are integers.
- $3 \le N \le 1\,000$
- $120\,000 \le Q \le 1\,000\,000$
- $0 \le x_i, y_i, a_j, b_j, c_j, d_j \le 1\,000\,000\,000$ for all integers $0 \le i < N$ and $0 \le j < Q$.
- $x_i \ne x_j$ or $y_i \ne y_j$ for all integers $0 \le i < j < N$.
- $a_i \ne c_i$ or $b_i \ne d_i$ for all integers $0 \le i < Q$.
- The sequence of points $(x_0, y_0), (x_1, y_1), \cdots, (x_{N-1}, y_{N-1})$ defines a counterclockwise traversal of the boundary of a simple polygon.
- Choosing the counterclockwise path from (a_i, b_i) to (c_i, d_i) for the modification on the magic circle after the first i modifications is valid for all integers $0 \le i < Q$.
- The final magic circle is usable.

Note that it's allowed to have two consecutive segments meeting at the angle π. Also note that it is not required to minimize any number, your program just have to satisfy all the output constraints.

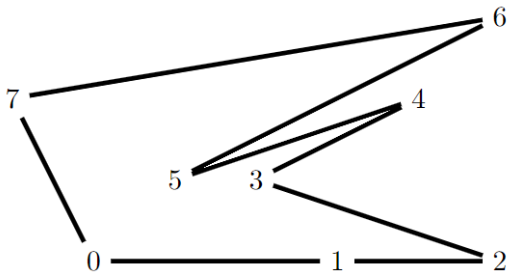
Example

standard input	standard output
	8 1 0 4 0 6 0 3 1 5 2 2 1 6 3 0 2 4 6 0 6 3 10 2 6 3 10 2 9 4 9 4 0 2

Note

Please note that the sample output above does not satisfy the condition $120\,000 \le Q \le 1\,000\,000$, thus it will give Wrong Answer verdict upon submission. It is there only to present the output format.

The following illustrates the initial magic circle for the sample output.



The following illustrates the sequence of usage of the tool to make it usable. The dotted part of the boundary is the path modified by the tool, which becomes the red part after the modification.