# Problem A. Point on the Line segment

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

## Input

Single line contains six integers — coordinates of the point and ends of the line segment.

## Output

Output `YES`, if point lies on the line segment, and `NO` — otherwise.

## Examples

| standard input | standard output |
|---|---|
| 3 3 1 2 5 4 | YES |
| 4 2 4 2 4 5 | YES |

# Problem B. Distance to line segment

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

## Input

Single line contains six integers — coordinates of the point and ends of the line segment.

## Output

Output a number — distance to line segment from a given point precise up to $10^{-6}$.

## Examples

| standard input | standard output |
|---|---|
| 0 4 2 3 2 5 | 2.00000000000000000000 |
| 4 0 0 0 4 0 | 0.00000000000000000000 |

# Problem C. Distance between line segments

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

Find the distance between given line segments.

## Input

Two lines contain four integers each — coordinates of the ends of the line segments.

## Output

Output the distance between line segments with accuracy no less than $10^{-6}$.

## Examples

| standard input | standard output |
|---|---|
| 1 1 2 2<br>2 1 3 0 | 0.707106781 |
| 1 1 2 2<br>1 2 2 1 | 0.000000000 |

# Problem D. Two Circles

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

You are given two circles. Intersect given circles and find the points of their intersection.

## Input

First line of input contains one positive integer up to 10 000 — the number of testcases.

Then follow the testcases. Every testcase consists of two lines. Each of these lines describe one of two circles containing three integers: coordinates of center and radius of the circle.

All numbers are integers not exceeding 10 000 by their absolute value. Radius is positive.

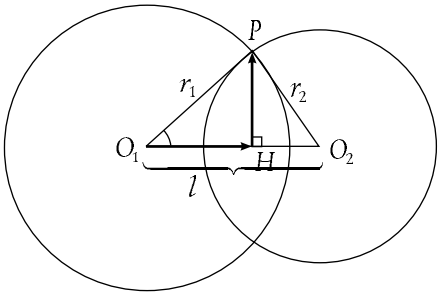
## Output

For each testcase output several lines. The first line should contain the number of points of intersection: 0, 1 or 2, if it's finite, otherwise output 3. If there is only one point of intersection, output its coordinates on the second line. If there are two points of intersection, output coordinates of points $H$ on the second line, third line should contain two numbers: the lengths of vectors $\overrightarrow{O_1H}$ and $\overrightarrow{HP}$, and the next two lines — the coordinates of points of intersection. You can output these two points in arbitrary order.

## Example

| standard input | standard output |
|---|---|
| 4<br>3 4 5<br>11 4 2<br>3 4 5<br>11 4 3<br>3 4 5<br>11 4 4<br>3 4 5<br>3 4 5 | 0<br>1<br>8.0000000000 4.0000000000<br>2<br>7.5625000000 4.0000000000<br>4.5625000000 2.0453835215<br>7.5625000000 6.0453835215<br>7.5625000000 1.9546164785<br>3 |

## Note




The output is not ambiguous. Coordinates and lengths should be output with no less that six correct digits after the decimal point. Note, that the input consists of integers, so the value "number of points of intersection" can be determined absolutely precisely.

# Problem E. Area

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

## Input

First line contains single integer $n$ — number of vertices of the polygon ($3 \le n \le 100\,000$).

Next $n$ lines contain coordinates of the vertices in a clockwise or counterclockwise direction.

All coordinates are integers with absolute values not exceeding $10^4$.

## Output

Output a single number — area of the polygon.

## Example

| standard input | standard output |
|---|---|
| 3<br>1 0<br>0 1<br>1 1 | 0.5 |

# Problem F. Perimeter

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

For given points on a plane find the perimeter of the convex hull.

## Input

First line contains a single integer $n$ — number of the points ($3 \le n \le 10^5$).

Next $n$ lines contain coordinates of the points — a pair $(x, y)$ one per line ($-10^9 \le x, y \le 10^9$).

Note that, given points can coincide and lay on the same line.

## Output

Output the perimeter of the convex hull as accurate as possible.

## Example

| standard input | standard output |
|---|---|
| 5<br>0 0<br>2 0<br>0 2<br>1 1<br>2 2 | 8 |

# Problem G. Halfplanes Intersection

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

You are given $n$ halfplanes. Find the area of their intersection. The area of the intersection is guaranteed to be finite.

## Input

The first line of the input contains an integer $n$, the number of halfplanes ($3 \le n \le 2000$). Next $n$ lines describe halfplanes: each line contains three integers $a_i$, $b_i$, and $c_i$, coefficients of the $i$-th halfplane $a_i \cdot x + b_i \cdot y + c_i \ge 0$ ($|a_i|, |b_i| \le 2 \cdot 10^4$, $|c_i| \le 10^9$).

## Output

Print one real number, the area of the intersection. The absolute or relative error must be below $10^{-6}$.

## Example

| standard input | standard output |
|---|---|
| 3<br>1 0 0<br>0 1 0<br>-1 -1 1 | 0.5000000000 |

# Problem H. Areas

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

You're given $n$ straight lines on 2D plane. They divide the plane into parts. Some of them are finite and others are infinite. Find areas of all finite parts.

## Input

First line contains integer $n$ — the number of lines ($1 \le n \le 80$).

Each of next $n$ lines contain four integers $x_1$, $y_1$, $x_2$ and $y_2$ — coordinates of two distinct points of $i$-th line.

Coordinates don't exceed 100 by absolute value. All lines are pairwise distinct.

## Output

In the first line output $k$ — number of finite parts.

In next $k$ lines output areas of these parts in non-decreasing order. The error shouldn't exceed $10^{-4}$.

Ignore and don't output parts with an area less than $10^{-8}$.

## Example

| standard input | standard output |
|---|---|
| 5<br>0 0 1 0<br>1 0 1 1<br>1 1 0 1<br>0 1 0 0<br>0 0 1 1 | 2<br>0.5000000000<br>0.5000000000 |