# Packing

SEASON 9 – FIRST ROUND

We will call *figure* a set of connected convex polygons, where it is possible to reach any polygon from any other by traveling through the common sides of the polygons and there are no two polygons with a common interior point

You are given N figures that you can move and rotate separately, without changing the relative position of the polygons in any of them. You could also choose to not use some of the figures. After the transformations, none of the used figures should overlap with each other (they should have no common interior point).

Your task is to minimize the $score = S_{hull} + 1.5^{Nrem} * S_{rem}$, where:

- $S_{hull}$ is the area of the convex hull of the used figures (that is to say the smallest convex polygon, that contains all of them).
- $S_{rem}$ is the sum of the areas of the unused figures
- $N_{rem}$ is the number of unused figures

## Input

Read the input from the file `packing.in`

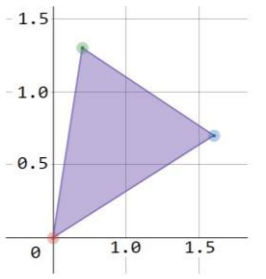
The first line of the input contains one integer **F** – the number of figures. **F** descriptions of these figures follow, each one consisting of:

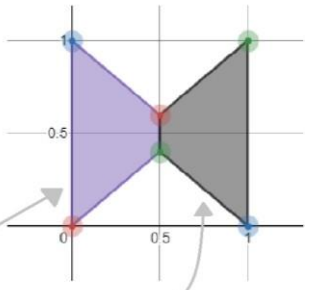
The number of points **P**, followed by P pairs of rational numbers $X_i$ $Y_i$
($0 \le i < P$)

The number of convex polygons **M**, followed by their descriptions.
The description of each of the polygons consists of the number of vertices **K**, followed by **K** indices of the points that represent its vertices. (In counterclockwise order).

See the image for clarification.

```
2
3
0.0 0.0
1.1 0.7
0.2 1.3
1
3 2 1 0
6
0.0 0.0
0.0 1.0
0.5 0.4
0.5 0.6
1.0 0.0
1.0 1.0
2
4 0 2 3 1
4 4 5 3 2
```



# Packing

SEASON 9 – FIRST ROUND

## Output

Write your output in the file `packing.out`

Your program should print N lines describing where to place each of the figures.

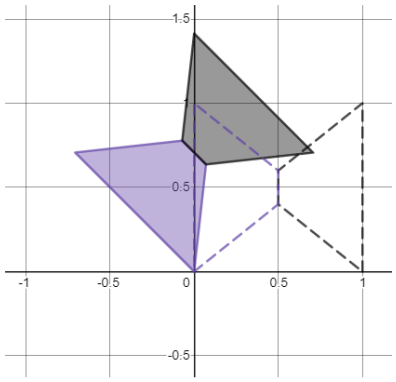
For each of the figures first print 1 or 0 – respectively, whether or not you want to use the figure. **If you want to use it**, print three real numbers $A_i$ $X_i$ $Y_i$ – the angle at which it should be rotated and the coordinates, with which it should be translated afterwards. The angle is measured in degrees and denotes a counterclockwise rotation about the origin of the coordinate system.

Each of the numbers should be printed with up to 6 digits after the decimal point.
Additionally $0 \le A_i \le 360$ and $-10^6 \le X_i, Y_i \le 10^6$

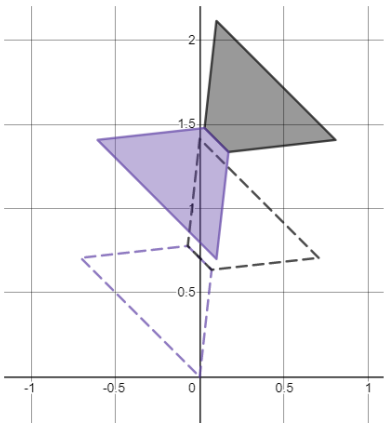
Note that the constrains for the coordinates in the output are different from these in the input.

```
1 0 0 0
1 45 0.1 0.7
```

1) A = 45°




2) X = 0.1
Y = 0.7




## Constrains

$2 \le F \le 500$
For each figure $3 \le P \le 100$, $1 \le M \le 100$, and for each polygon $3 \le K \le P$
The sum of P for all figures is no more than 5000
$-10^3 \le X_i, Y_i \le 10^3$
Each of the coordinatres has up to 6 digits after the decimal point.

**Time limit: 5 sec**
**Memory limit: 256 MB**

# Packing

SEASON 9 – FIRST ROUND

## Grading

You will receive 0 points for a test, if:

1. The output does not follow the described constrains
2. After the transformations there is at least one pair of overlapping figures

If your output is valid, you will receive$100 * \left(\frac{minScore}{yourScore}\right)^4$ % of the points for that test. We define yourScore as the score your program got and minScore as the minimum score, that some of the contestants' programs received.

## Subtasks

| Number of test cases | Constrain |
|---|---|
| 40% | Every figure consists of exactly one convex polygon |
| 20% | Every figure is a tetromino, i.e.it is made out of 4 squares with side 1 |
| 40% | No further constrains |

## Example

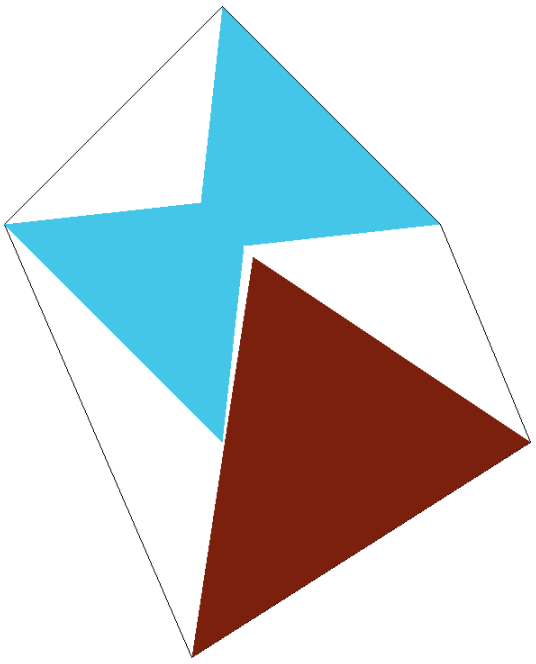
| Input (packing.in) | output (packing.out) |
|---|---|
| 2<br>3<br>0.0 0.0<br>1.1 0.7<br>0.2 1.3<br>1<br>3 2 1 0<br>6<br>0.0 0.0<br>0.0 1.0<br>0.5 0.4<br>0.5 0.6<br>1.0 0.0<br>1.0 1.0<br>2<br>4 0 2 3 1<br>4 4 5 3 2 | 1 0 0 0<br>1 45 0.1 0.7 |

**Notes**

This is the example shown above. It looks like this after the transformations:

# Packing

SEASON 9 – FIRST ROUND



The area of the convex hull is ~1.99 and because all of the figures are used, this is also the **score**.

## Visualizer

You are provided with a visualizer. Check the site for additional information.