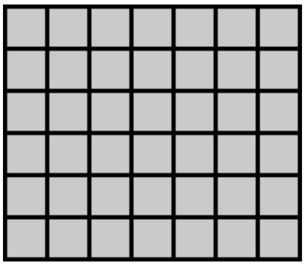
Figures in an Infinite Grid (Approximate)

You have an infinite grid made of numerous 1×1 cells. The following figure shows a small subsection of the grid:



Also You have n figures indexed from 1 to n. A figure is connected set of 1×1 cells. The diagram below shows some examples:

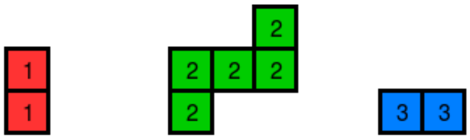
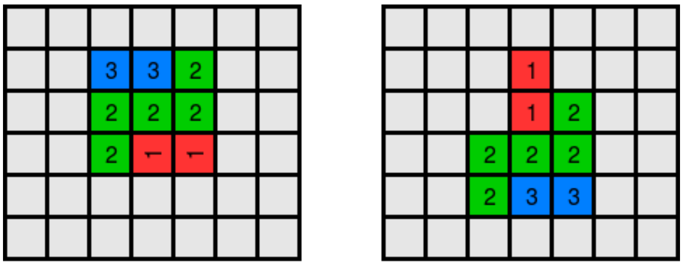


Figure 1 Figure 2 Figure 3

Your task is to put figures inside the grid. The Figures must not overlap. It's allowed rotate a figure only by 0^o or 90^o or 180^o or 270^o.

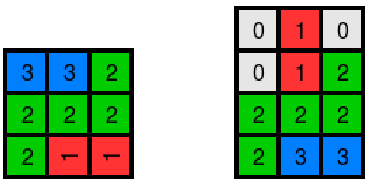
The following figure shows two different ways to arrange the figures:



Arrangement 1 Arrangement 2

Now we have to cut the smallest rectangle that contains all the figure. Your task is to print positions of the figures in this rectangle. You score for this problem depends on the size of this rectangle, smaller rectangle will give higher scores. (See scoring section for details).

For the examples above, arrangement 1 will produce smaller rectangle as shown in the following diagram:



(We put 0 in the cells that don't contain any figures)

Scoring

Let (w, h) be the height and width of the rectangle you printed and let s_{min} be the minimum possible area of the rectangle that can contain all figures.

$$\frac{s_{min}}{w \times h}\%$$

In the examples above, $s_{min} = 9$. If your solution is similar to arrangement 1, you will get $\frac{9}{3 \cdot 3}\% = 100\%$ score. But If your solution is similar to arrangement 2, you will get $\frac{9}{3 \cdot 4}\% = 75\%$ score.

Test case generation

- At first we create the rectangle of the size $w \times h$ with each cell having a different color.
- Let $figureSize$ be the largest possible size of the figure. Now we will take two random neighbouring cells, let's define their colors as $color_1$ and $color_2$. If $color_1 != color_2$ and $componentSize[color_1] + componentSize[color_2] <= figureSize$, we merge components with colors $color_1$ and $color_2$. We will do this fixed number of times.
- After that we rotate each of the received figures some random number of times. Let's define 1×1 cells of the figure of size sz as $[(x_1, y_1), \ldots, (x_{sz}, y_{sz})]$. We will move each figure to the center of coordinates in such way that $min(x_1, \ldots, x_{sz}) = min(y_1, \ldots, y_{sz}) = 1$

Input Format

The first line contains n, the number of figures. For each figure:

- The first line contains s_i denoting the number of cells the figure contains.
- s_i subsequent lines will contain two space-separated integers x_j and y_j. This means if you put a block in these cells, you will get the shape and size of the figure.

It's guaranteed that the figures will be connected. A cell is connected to another cell if they share an edge.

Constraints

- $1 \leq n \leq 1500$
- $1 \leq \sum s_i \leq 25000$
- $1 \leq x_j, y_j \leq 150$

Subtasks

- $\sum s_i \leq 100$ for 20% of the dataset.
- $\sum s_i \leq 1600$ for 30% of the dataset.

Output Format

On the first line, print two space-separated integers h and w denoting the height and width of the rectangle.

Each of the next h lines should contain w space-separated integers describing the rectangle. Let $c_{i,j}$ be the j^{th} integer of the i^{th} row. If there are no figure in cell (i, j) then $c_{i,j} = 0$. Otherwise, $c_{i,j}$ should denote the index of the figure that is present in cell (i, j).

Sample Input 0

```
3
2
1 1
2 1
5
1 3
2 1
2 2
2 3
3 1
2
```

```
1 1
1 2
```

Sample Output 0

```
3 3
3 3 2
2 2 2
2 1 1
```

Explanation 0

This sample input corresponds to the example shown in the problem statement. The figures were placed in the similar way of arrangement 1. This will give 100% score (see scoring section).