Problem B
k-Dominant Skyline

Input File: *testdata.in*
Time Limit: 10 seconds

Problem Description

Skyline queries are useful in many applications such as multicriteria decision-making, data mining, and user-preference queries. Give a set P of d-dimensional points, the skyline operator returns the points in P that are not dominated by other points. A point p_i *dominates* another point p_j if $p_i.d_x \geq p_j.d_x$ ($p_i.d_x$ denotes the x-th dimension value of the i-th points and $1 \leq x \leq d$) and is strictly larger in at least one dimension. However, the probability of a point dominating another one decreases significantly as the number of dimensions increases, which makes it possible that a large number of skyline points will be retrieved. To solve this problem, the idea of k-domination is introduced to reduce the number of significant points retrieved in a high-dimensional database. A point p is said to *k-dominate* another point q if there are $k(\leq d)$ dimensions in which p is better than or equal to q and is better in at least one of these k dimensions. And a point is in k-dominant skyline if it is not k-dominated by any other points. The following is the formal definition for k-dominant skyline.

Definition: A point p_i is said to k-dominate point p_j on D if and only if $\exists D' \subseteq D$, $|D'| = k$, $\forall d_i \in D'$, $p_i.d_i \geq p_j.d_i$, and $\exists d_j \in D'$, $p_i.d_j > p_j.d_j$. Moreover, p_j is said to be k-dominated by p_i. A point p_i is a k-dominant skyline point if and only if there does not exist any point p_j in the dataset that k-dominates p_i.

For example, if a person wants to purchase a notebook computer at a website. He may consider a large number of features—including the CPU type, the RAM and VRAM sizes, the HDD capacity and speed, and the overall quality—when selecting one that suits him. Without loss of generality,

we assume a larger value represents better performance. Referring to Table 1, p_2 is 4-dominated by p_1 in dimensions of CPU type, RAM size, VRAM size and HDD capacity, therefore, p_2 is not a 4-dominant skyline point. Similarly, p_3 and p_4 are also not 4-dominant skyline points because p_3 is 4-dominated by p_1 (in dimensions of CPU type, RAM size, VRAM size and HDD capacity) and p_4 is 4-dominated by p_2 (in dimensions of CPU type, VRAM size, HDD speed and overall quality). Moreover, p_1 is not 4-dominated by any other points in the dataset, p_1 is a 4-dominant skyline point.

Table 1: Dataset for selecting a notebook.

Notebook	CPU type	RAM size	VRAM size	HDD capacity	HDD speed	Overall quality
p_1	9	8	7	6	8	8
p_2	9	4	7	5	8	8
p_3	4	4	3	2	8	4
p_4	3	9	7	7	5	3

Assume a larger value represents better performance. Given dataset and k, can you determine the number of k-dominant skyline points in the dataset?

Technical Specifications

1. The number of test cases would be smaller than or equal to 10.
2. The number of points in a dataset (n) would satisfy $0 < n < 50$.
3. The range of each dimension value would satisfy $0 < p.dx < 20$
4. The number of dimensions is 4 and k is 3.

Input Format

The first line of the input file contains an integer indicating the number of test cases to follow. Each test case contains an integer n and a sequence of data points, separated by end of line. A data point is a single line containing 4 integers delimited by a space

Output Format

For each test case, output the number of k-dominant skyline points in a line.

Sample Input

```
3
2
5 3 1 7
4 2 2 8
3
9 4 5 3
7 8 9 7
2 9 6 5
3
8 10 4 6
1 2 3 8
6 4 8 7
```

Sample Output

```
2
1
2
```