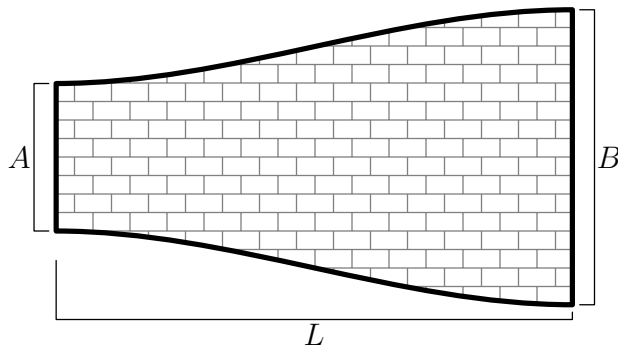


Chimney Service (chimneys)

Giorgio is always on the lookout for new business opportunities. Today, he acquired a bankrupt company that used to deal with industrial construction material. As soon as he made the purchase, he went into the warehouse and noticed that there were N perfectly preserved “chimney parts”. A chimney part is, essentially, a pipe (or tube) made of concrete that is characterized by two diameters, A and B , one for each open side of the pipe, and a length L .



By combining these chimney parts, Giorgio can build and sell chimneys to nearby industries. As you might know, in order to build a functional chimney, the structure should be getting progressively narrower towards the top. This is essential in order for the steam or smoke to naturally flow upward, instead of creating air pockets and stagnating. Furthermore, two chimney parts can be combined together only if they perfectly match: the larger diameter of a part should be **exactly equal** to the smaller diameter of the next part, otherwise they would leave openings for the smoke. The total length of a chimney is given by the sum of the L values of its parts.

Giorgio wants to get rid of **all** the chimney parts. Thus, he decided to build several different chimneys, while ensuring that **all of them are high enough** so that they can all be of interest to some possible buyers. In more practical terms, Giorgio wants to combine the chimney parts into a set of chimneys (no matter how many) such that *the shortest chimney is as long as possible*.

Among the attachments of this task you may find a template file `chimneys.*` with a sample incomplete implementation.

Input

The first line contains the only integer N , the number of chimney parts. The next N lines contain three integers each that describe the i -th chimney part: A_i , B_i , and L_i .

Output







The first line of the output should contain an integer C : the number of chimneys you have assembled. Then, $2C$ lines should follow. Two consecutive lines describe how a chimney is formed: the first one should contain an integer k representing the number of chimney parts used for assembling that chimney, and the second one should contain k integers, the indexes of the parts used.

Constraints

- $1 \leq N \leq 10\,000$.
- $1 \leq A_i < B_i \leq 100\,000$ for each $i = 0 \dots N - 1$.
- $1 \leq L_i \leq 10^9$ for each $i = 0 \dots N - 1$.
- Note that a single chimney part is already by itself a valid chimney.
- The description of chimneys in the output should follow the assembling order (**increasing order of diameters**): the first index refers to the part on top of the chimney, and so on.

Scoring

Your program will be tested against several test cases grouped in subtasks. Your score on a subtask will be equal to the minimum score for one of its testcases multiplied by the value of the subtask. Your score on a test case will be zero if the output does not represent a valid set of chimneys, using exactly once every chimney part in the input. If the output is a valid set of chimneys, let S_{out} be the minimum length of a chimney among them, and S_{ref} be the minimum chimney length in a reference solution (not necessarily optimal). Your score on a test case will be 1 if $S_{\text{out}} \geq S_{\text{ref}}$, or $\frac{S_{\text{out}}}{S_{\text{ref}}}$ otherwise.

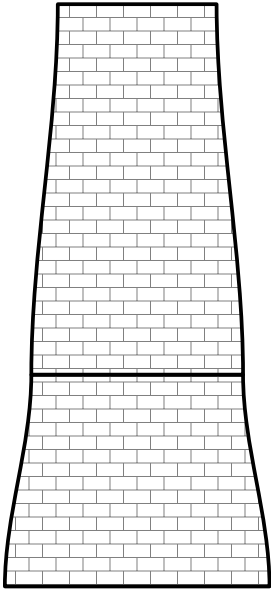
- **Subtask 1** (0 points) Examples.

- **Subtask 2** (7 points) All the values of A are distinct. All the values of B are distinct.

- **Subtask 3** (18 points) All the values of B are distinct.

- **Subtask 4** (27 points) $N \leq 15$.

- **Subtask 5** (16 points) $L_i = 1$ for each $i = 0 \dots N - 1$.

- **Subtask 6** (32 points) No additional limitations.


Examples

input	output
2 4 5 4 3 4 7	1 2 1 0
3 4 5 4 3 4 7 1 4 10	2 2 1 0 1 2

Explanation

In the **first sample case**, represented in the following picture, we can assemble the two parts to form one chimney placing the one at index 1 at the bottom, and the one at index 0 at the top.



In the **second sample case** we have the same two parts as in the first example and an extra one. Unfortunately, the last one cannot be combined with the others, which means that we need to create a smaller chimney just with it.

