

Bracket Convex Hull

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 1024 megabytes

Given n points on a two-dimensional plane, each point is labeled with a bracket, which may be either a left bracket (or a right bracket).

We say that a string S is a valid bracket sequence if and only if it can be generated by the following recursive rules:

- The empty string is a valid bracket sequence;
- If A is a valid bracket sequence, then the string (A) is also a valid bracket sequence;
- If both A and B are valid bracket sequences, then AB is also a valid bracket sequence;
- Any string that cannot be generated by the above rules is not a valid bracket sequence.

For example, $()$, $(())$, and $()()$ are valid bracket sequences, while $)()$, $((()$, and $()()$ are not valid bracket sequences.

Now you need to choose some distinct points from the given points as vertices to form a convex polygon.

In this problem, a polygon formed by m points $p_{a_1}, p_{a_2}, \dots, p_{a_m}$ is called a convex polygon if and only if the following conditions are satisfied:

- $m \geq 3$;
- Connecting these points in the order a_1, a_2, \dots, a_m , and then connecting a_m and a_1 , forms a simple polygon;
- For every edge of the polygon, all other vertices lie strictly on the same side of the line containing that edge.

In other words, the selected points must appear exactly in the cyclic order of their own convex hull, and all interior angles must be strictly less than 180° ; that is, the convex polygon has no three collinear points.

For a convex polygon, choose any vertex on its boundary as the starting point, and traverse all vertices along the polygon boundary in either clockwise or counterclockwise direction, stopping just before returning to the starting point. This yields a bracket sequence of length m : if the current vertex is labeled with a left bracket, write (; if it is labeled with a right bracket, write).

Your task is to find a convex polygon containing at least one left bracket such that, for any vertex on the polygon labeled with a left bracket, the bracket sequence obtained by starting from that vertex and traversing the polygon boundary in either direction is a valid bracket sequence.

If such a convex polygon exists, output any one of them; otherwise, report no solution.

Input

The first line contains an integer T ($1 \leq T \leq 400$), the number of test cases.

For each test case, the first line contains an integer n ($1 \leq n \leq 2000$), the number of points.

The next n lines each contain three integers x_i, y_i, t_i ($0 \leq x_i, y_i \leq 10^9, t_i \in \{0, 1\}$), representing the coordinates and bracket type of the i -th point. Here, $t_i = 0$ denotes a left bracket, and $t_i = 1$ denotes a right bracket.

It is guaranteed that, within each test case, all points are distinct and no three points are collinear.

It is also guaranteed that the sum of n over all test cases does not exceed 2000.

Output

For each test case, if there is no solution, output a single integer -1 on one line.

Otherwise, output the number of vertices m of the selected convex polygon on the first line.

On the second line, output m pairwise distinct integers a_1, a_2, \dots, a_m , representing the indices of the selected points.

These points must be output in the order along the boundary of the convex polygon, either clockwise or counterclockwise.

Example

standard input	standard output
5	4
4	1 2 3 4
1 1 0	-1
2 4 1	4
3 9 0	1 3 2 4
4 16 1	-1
5	-1
1 1 0	
2 4 0	
3 9 0	
4 16 1	
5 25 1	
6	
47 58 0	
30 23 0	
27 34 1	
35 7 1	
10 30 1	
1 25 1	
8	
10 5 0	
10 16 0	
1 5 0	
24 9 0	
6 2 0	
6 12 0	
7 18 1	
3 13 1	
1	
0 0 0	