Consider the following game played between two players **A** and **B**. A position in the game is defined by an array of N elements, where each element in the array is a point in the plane with non-negative integer coordinates. The allowed points are only those with x coordinate greater than or equal to the y-coordinate. In other words (x, y) such that $x \geq y \geq 0$. A move in the game consists of replacing some point (x, y) in the array by a point (p, q) such that $p \geq q \geq 0$, and one of the following holds:

1. $p = x$ and $0 \leq q < y$.

2. $q = y$ and $q \leq p < x$.

3. $0 \leq p = q \leq y < x$.

Informally, in each move, a point can be moved vertically downward, or horizontally to the left until it meets the line $y = x$, and then can be moved along this line downward and to the left, maintaining $y = x$. After any such move, the new coordinates will still satisfy $x \geq y \geq 0$. The player who cannot make a move loses the game. This happens when all points in the array are $(0, 0)$. Given the initial N points in the array, you have to determine whether the first player **A** has a winning strategy starting from the given position, and if so determine all possible moves that guarantee a win.

Input Format The first line of input contains the number N, where $1 \leq N \leq 100$. The next N lines contain the coordinates of the N points, one per line, starting from the 0th element of the array. Each line contains two non-negative integers x, y separated by a space, with $0 \leq y \leq x \leq 10^{15}$.

Output Format The first line of output should be the number of winning moves for the first player. If this number is k, the next k lines should specify the winning moves. (If there is no winning strategy, just output 0). Each move should be specified by three numbers, the index of the point in the array that is moved, and the x and y coordinates of the point that replaces it. The moves should be ordered in lexicographic order, with smallest index first, and for equal index, by smaller x-coordinate.

```
Sample Input          Sample Output
2                     2
5 3                   0 3 3
6 1                   0 5 0
```