

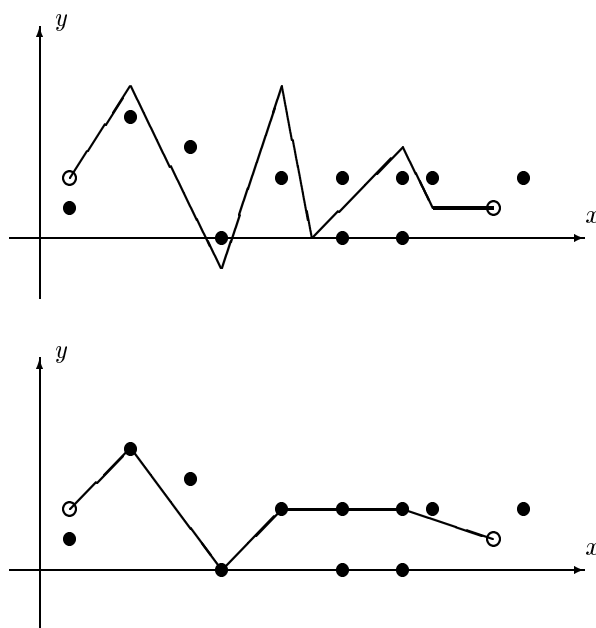


Pull the String

Time limit for each test: 2000 milliseconds
Memory limit: 40 megabytes

n nails are driven halfway through the surface of a wooden table. We will refer to points on the table by specifying their (x, y) coordinates. A thin string is placed on the table as a broken line such that it is x -monotone along its path. More specifically for each x there is at most a single y such that (x, y) is on the string's path. The two ends of the string go into a tiny hole drilled in the table. We know that if we pull the ends of the string from under the table, the length of the string will decrease until it can't decrease anymore without passing through any of the nails. For example, if in the initial state, the string has passed from above of a certain nail (by a greater y coordinate), pulling the string can't cause the final state to be in such a way that the string would pass from below the same nail. Our goal is to find the final state of the string, after we pull it as much as possible.

In the figures below, nails are shown with black circles and the holes are shown by white circles. The top figure shows the initial state of a string and the bottom figure shows the final state.



Problem

Write a program that

- Reads the initial state of the string and the positions of the nails from the *Standard Input*,
- Computes the final state of the string after it is completely pulled.
- Describes the final state of the string in the *Standard Output*.

Input Sepcification

In the first line of the input a single integer n is given which is the number of nails. In each of the next n lines two integers x and y are given which means that a nail is on point (x, y) . No two nails are placed on the same point.

In the next line (the $(n + 2)_{th}$ line) a single integer m is given which means that the string's path breaks at exactly $m - 2$ points. In each of the next m lines two integers x and y are given, which specify points on the path of the string, beginning with the left-most point and ending with the right-most point; The first of these m points is the left hole (the one with a smaller x) and the last is the right hole. The points are given in order of their x coordinates. No three subsequent points in this part of the input are in a straight line. At the initial state, the string doesn't touch any of the nails. Additionally, no nail is driven in the two tiny holes.

Output Specification

In the first line of the output, a single integer k should be written, meaning that apart from the two endpoints of the string, there are $k - 2$ points where the path of the string breaks. In each of the next k lines you should specify a breaking point on the path of the string in the final state, by writing two integers x and y . These points should be ordered according to their x coordinates. The first of these k points should be the tiny hole on the left and the last should be the one on the right. Make sure that the points written are strictly "breaking points" meaning that the line should break on those points; No three subsequent points of the output should be on a straight line.

Restrictions

- The thickness of the string and the diameter of the nails are negligible.
- The absolute value of coordinates in the input doesn't exceed 10^9
- $0 \leq n \leq 100,000$ and $2 \leq m \leq 100,000$
- In 40% of the grade $m, n \leq 200$.
- In 92% of the grade $m, n \leq 30,000$.

Standard Input	Standard Output
11	6
3 4	1 2
1 1	3 4
6 0	6 0
10 0	8 2
8 2	12 2
5 3	15 1
12 2	
13 2	
12 0	
10 2	
16 2	
8	
1 2	
3 5	
6 -1	
8 5	
9 0	
12 3	
13 1	
15 1	