Problem G. Yandex Museum

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Yandex is a Russian technology company developing lots of products that make our everyday life easier: web search, e-mail, taxi service, maps, and so on. Many of Yandex products are running on the most modern intelligent devices varying from computers and mobile phones to self-driving cars and smart home assistants. But there were times when computers were less user-friendly, much less efficient and required true technical skills just to run some program. Does any of the names "Commodore 64", "Elektronika MK-90", "ZX Spectrum 128" and "Apple IIe" raise nostalgia feelings in you? For some Yandex employees, these words mean a lot. That's why Yandex has created a special Yandex Museum that is free and open for anybody to visit, and has an exhibition of old computers including all aforementioned legendary devices!

Arkady decided to bring to the museum his old computer which is not functioning any more. This computer has a rather primitive display, which was able to display only four colors: black, red, green and blue. Still, Arkady spent a lot of time with that computer back in the day, especially with the primitive image editing program.

Initially, when user starts working on an image, the drawing area is filled with red color. The favorite drawing tool of Arkady is a triangle tool, which works as follows: user chooses three distinct integer points A, B and C defining a non-degenerate triangle (i.e. of positive area), and then editor draws a triangle $\triangle ABC$ with the given points as its vertices. As you remember, the screen is able to display only few colors, so the coloring logic is following.

- All points belonging to the sides of the triangle (to at least one of the segments AB, BC and AC) are painted black. Once painted black, point never changes its color and remains black till the very end.
- Each non-black point in the triangle interior changes its color to the next one in order of "RGB", i.e. every point that is currently red is colored green, every green point is colored blue and every blue point becomes red.
- Points that are strictly outside of the triangle do not change their colors.

Arkady used to enjoy drawing triangles, but he never liked blue and green colors, so he was especially happy when after some drawing steps picture consisted of black and red colors only (by the way, same holds for Yandex logo!). He called such pictures *nice*.

While packing the computer to bring it to the museum, Arkady found an old piece of paper containing n sextuplets of integers. He supposed that these integers are the coordinates of vertices of n triangles he has once drawn during his younger days. Indeed, he noticed that all triangles are non-degenerate, but he was not able to figure out by simply looking at these numbers, whether the resulting picture is nice or not. Help him find the answer for that question!

Input

The first line of the input contains an integer n ($1 \le n \le 100\,000$), the number of triangles.

Each of the following n lines contains six integers which are the coordinates of the triangle. Each triangle is non-degenerate, and each coordinate is between 0 and 10^4, inclusive.

Output

If the resulting picture is nice, print a single word "`nice`".

Otherwise, on the first line print "`not nice`", and on the second line print two (not necessarily integer) numbers, which are coordinates of any point having blue or green color. The numbers should be printed

as integers or decimal fractions with at least 0 and **at most** 9 digits after decimal point. Coordinates should be between 0 and 10^4, inclusive. Note that usage of exponential format **is not allowed**. Please, also note that your answer will be verified with no absolute or relative tolerance.

Examples

standard input	standard output
5 0 0 0 2 1 0 0 0 2 0 0 1 0 0 0 2 2 0 2 0 0 1 0 2 0 2 1 0 2 0	nice
5 0 0 1 0 0 1 1 0 2 0 1 1 0 0 0 2 2 0 0 1 0 2 1 1 0 0 2 0 0 2	not nice 0.7 0.8

Note

Illustrations for sample cases are given below:

