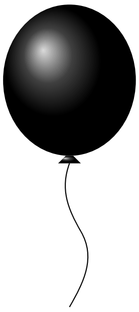
F Blackjohn



Time limit: 2.0s
Memory limit: 512MB

Lately, Blackjack has not been bringing John the same pleasure as before. All probabilities have been calculated long ago, and the optimal game strategy has been developed. Therefore, John decided to create his own modification of this game — Blackjohn.

John took a deck of cards and wrote a fraction $\frac{p_i}{q_i}$ on the i-th of them. The fractions do not exceed 1 in their absolute value. The fraction is called the value of the card. The rules are as follows: players take turns opening cards from the deck and either keep them in their hand or discard them. To win, it is necessary to achieve a sum of the card values in the hand equal to one.

John wants to show the new game to his friends, but he has a serious problem. The game turned out to be so complex that even knowing the deck of cards in advance, he cannot determine which cards to keep.

John asked you to write a program that, given a deck of cards, determines which cards should be kept in order to win.

Input

The first line contains an integer n ($1 \le n \le 100$) — the number of cards in the deck.

Each of the next n lines describes one of the cards and contains two integers p_i and q_i separated by a space ($1 \le q_i \le 21$, $|p_i| \le q_i$).

Output

If winning is impossible, output a single line `NO`.

Otherwise, output `YES` on the first line. On the second line, output an integer m — the number of cards in the hand at the end of the game. On the third line, output m numbers $k_1, k_2, \ldots, k_m$ — the numbers of the cards whose sum of values is equal to one.

The cards are numbered starting from one in the order they are listed in the input.

Samples

Sample input 1	Sample output 1

4
1 2
-1 6
1 5
2 3
	YES
	3
	1 2 4

Sample input 2	Sample output 2

`4`
`1 2`
`1 4`
`1 8`
`1 16`
	`NO`