Problem 2018.1.5 - Casino

A famous casino from Bucharest wishes to detect card cheaters faster. Using an image recogntion software one can detect the cards played by each player at a certain table and one wishes to find if any player used cards extracted from its pockets. The card game that is monitored uses **two** classic card decks (52 cards each deck, without the Joker).

Requirement

Write a software that can help detect cheaters. If one detect that one card appears too often, the software will display the card and will stop the game.

Input data

One will read from the keyboard (the *stdin* stream) the following data:

- On the first line is the number **n**, representing the maximal number of hands to be played;
- On the following **n** lines is the played card, in the format:
  `<card_number> <card_sign>`

Output data

If no one tries to cheat, the text "**JOC OK**" will be displayed. If someone tries to cheat, the software will display the cheated card in the same format as the one for the input (the card number and the card sign, separated by space).

ATTENTION to the compliance to the problem requirements: the display of results must be done EXACTLY as required! In other words, on the standard output stream there will be nothing displayed in addition to the problem requirements; following the automatic evaluation, any supplemental character displayed, or any display different than the requirements, will produce an eroneous result and will lead to the „Reject" of the solution.

Restrictions and remarks

1. 2<**n**<100
2. The sign of the cards can be: **trefla**, **caro**, **cupa**, **pica**.
3. The number of the cards is an integer number within [2; 14] (ace has value 11).
4. **Warning**: According to the chosen programming language, the file containing the code must have one of the extensions .c, .cpp, .java, or .m. The web editor does not add automatically these extensions and the lack of the extensions leads to the impossibility of program compilation!
5. **Warning:** The source file must be named by the candidate as: <name>.<ext> where name is the family name (last name) of the candidate and the extension is the one chosen according to the previous warning. Attention to the restrictions imposed by the Java language regarding the class name and the file name!

Examples

Input	Output	Explanation
5 2 trefla 11 caro 14 cupa 14 pica 6 caro	JOC OK	No card has been played 3 times => the game is normal.

7 2 trefla 11 caro 11 caro 11 caro 14 cupa 14 pica 6 caro	11 caro	The card 11 caro has been played 3 times.
7 2 trefla 11 caro 11 caro 6 caro 11 caro 14 cupa 14 pica	11 caro	The card 11 caro has been played 3 times, eventhough not consecutively.
7 11 caro 11 pica 11 caro 11 pica 11 pica 11 caro 14 pica 6 caro	11 pica	The card 11 pica is the first card that has been played 3 times.

Work time: 120 minutes