

B. Game with Strings

time limit per test: 1 second

memory limit per test: 512 megabytes

input: standard input

output: standard output

Given an $n \times n$ table T consisting of lowercase English letters. We'll consider some strings *good* if the table contains a correct path corresponding to the given string. In other words, good strings are all strings we can obtain by moving from the left upper cell of the table only to the right and down. Here's the formal definition of correct paths:

Consider rows of the table are numbered from 1 to n from top to bottom, and columns of the table are numbered from 1 to n from left to the right. Cell (r, c) is a cell of table T on the r -th row and in the c -th column. This cell corresponds to letter $T_{r, c}$.

A path of length k is a sequence of table cells $[(r_1, c_1), (r_2, c_2), \dots, (r_k, c_k)]$. The following paths are correct:

1. There is only one correct path of length 1, that is, consisting of a single cell: $[(1, 1)]$;
2. Let's assume that $[(r_1, c_1), \dots, (r_m, c_m)]$ is a correct path of length m , then paths $[(r_1, c_1), \dots, (r_m, c_m), (r_m + 1, c_m)]$ and $[(r_1, c_1), \dots, (r_m, c_m), (r_m, c_m + 1)]$ are correct paths of length $m + 1$.

We should assume that a path $[(r_1, c_1), (r_2, c_2), \dots, (r_k, c_k)]$ corresponds to a string of length k :

$$T_{r_1, c_1} + T_{r_2, c_2} + \dots + T_{r_k, c_k}.$$

Two players play the following game: initially they have an empty string. Then the players take turns to add a letter to the end of the string. After each move (adding a new letter) the resulting string must be good. The game ends after $2n - 1$ turns. A player wins by the following scenario:

1. If the resulting string has strictly more letters "a" than letters "b", then the first player wins;
2. If the resulting string has strictly more letters "b" than letters "a", then the second player wins;
3. If the resulting string has the same number of letters "a" and "b", then the players end the game with a draw.

Your task is to determine the result of the game provided that both players played optimally well.

Input

The first line contains a single number n ($1 \leq n \leq 20$).

Next n lines contain n lowercase English letters each — table T .

Output

In a single line print string "FIRST", if the first player wins, "SECOND", if the second player wins and "DRAW", if the game ends with a draw.

Examples

input
2 ab cd
output
DRAW

input
2 xa ay

output
FIRST

input
3 aab bcb bac
output
DRAW

Note
Consider the first sample:

Good strings are strings: a, ab, ac, abd, acd.

The first player moves first and adds letter a to the string, as there is only one good string of length 1. Then the second player can add b or c and the game will end with strings abd or acd, correspondingly. In the first case it will be a draw (the string has one a and one b), in the second case the first player wins. Naturally, in this case the second player prefers to choose letter b and end the game with a draw.

Consider the second sample:

Good strings are: x, xa, xay.

We can see that the game will end with string xay and the first player wins.