ECOO '18 R1 P3 - Missing Art

The local museum had a possible theft in their displays recently. The thief would have replaced the original art piece with a similarly created replica. The museum has a security system in place that is invisible to the naked eye, and they want to ensure that the art that is in the case is the original art, and not the replica.

For security, the N display cases have a specific digital code overlaid on the glass case that is invisible to the naked eye. The digital code is only readable by a scanner to tell the curators whether or not an art piece is an original. If someone were to tamper with the glass in any way, there would be an issue with the digital code.

The codes that are on the display must be converted to the values that the scanner has in its database. Converting the digital codes to the scanner-readable codes is done as follows:

- Each even digit must be increased by X. Digits can increase above 9, in which case they are replaced with the two-digit result.
- Each odd digit must be decreased by Y. Digits cannot decrease below 0.
- Each digit that equals zero must be changed to Z

You will also be given the expected scanner-readable codes for each of the art display cases, so that you can check to see if they match.

Your task is to find out whether the piece of art is authentic, and if it isn't, then you must "show your work" so that the curators can see the proof of your work.

Input Specifications

The standard input will contain 10 datasets. Each dataset begins with four integers N, X, Y, Z $(5 \le N \le 20$, $1 \le X, Y, Z \le 10)$. The next N lines each contain a display's digital code (of length at most 100 digits). The next line contains the letter `A` signifying the start of the answer key. The N lines each contain the answer keys to the display cases' digital codes. Finally, each dataset ends with a line containing an asterisk (`*`).

Output Specifications

For each dataset, if all the digital codes match with the answer keys, output `MATCH`. Otherwise, output `FAIL: ` followed by a comma-separated list of which display cases had a mismatch. Display cases are numbered starting from 1.

Sample Input (Two Datasets Shown)

```
4 1 2 3
12345678
87654321
28374827
18723470
A
72384728
72374822
39155935
32874281
*
3 2 3 1
02934820
40294838
82934820
A
146061041
6146610010
1046061041
*
```

Sample Output

```
FAIL: 1,2,4
MATCH
```