5342 Bingo

The game of BINGO is another of the great American pastimes. Our version of BINGO is to be played on a square card containing a 5×5 matrix.

For this problem, you are to write a program that accepts the integers that are to be placed in the cells of a "BINGO" card. Then you are to read in the "called" values, which may or may not be the same as the numbers on your BINGO card, one at a time until you either find a BINGO, or you run out of called numbers.

A BINGO occurs when one of the following rules has been fulfilled:

1. Each of the five elements in a row have either been matched by a called number, or are FREE,

2. Each of the five elements in a column have either been matched by a called number, or are FREE,

3. Each of the four corner elements have either been matched by a called number, or are FREE.

4. Each of the five elements in either prime diagonal have been matched by a called number or are FREE.

Any element in the matrix containing a zero (0) after the original load is FREE, and is considered to have been matched.

Input

There are several test cases.

The first five lines of every test case will each contain five integer values separated by spaces in the range $0 \leq V \leq 99$; these values are to be placed in the corresponding row of the matrix from left to right. Immediately following the values for the card will be a series of lines each containing one "called" number such that $1 \leq C \leq 99$; the last line will contain an invalid "called" number of zero (0).

Output

If you run out of called numbers before you find a BINGO, you are to print out the following message immediately following the last valid "called" number:

```
No BINGO on this card.
```

If you find a BINGO at some point in the input, you are to ignore the rest input values, and print out a notification of the BINGO followed by a list of comma separated triples in ROW ASCENDING order (and in COLUMN ASCENDING order if tie) that represent the Row, Column, and Value of all of the elements making up the BINGO. If there is more than a kind of BINGO at the same time, you should output all of them ordered by the kind of BINGO, with no blank lines between them.

The output should have the following format:

```
BINGO #N
R,C,V
R,C,V
...
R,C,V
```

Where:

N = kind of BINGO as specified above

R = row subscript from 1 to 5, and

C = column subscript from 1 to 5, and

V = called number value that matched the cell contents, or the word 'FREE'.

Print a blank line after each data set.

Sample Input

```
 1  2  3  4  5
11 12 13 14 15
21 22  0 24 25
31 32 33 34 35
91 92 93 94 95
21
22
24
25
99
0
 1  2  3  4  5
11 12 13 14 15
21 22  0 24 25
31 32 33 34 35
91 92 93 94 95
99
98
97
96
0
```

Sample Output

```
BINGO #1
3,1,21
3,2,22
3,3,FREE
3,4,24
3,5,25

No BINGO on this card.
```