At the largest conference on coding and cryptography the following theorem needed a proof or a counterexample: Suppose you are given a set of words of equal length; each word consisting of '0's, '1's and/or '*'s. Furthermore suppose the pattern of '*'s is different for all words in the set. By this we mean: if you replace all '0's and '1's by say '$' you obtain different words.

The claim is: if you replace the '*'s by '0's and '1's in all possible ways, then you obtain a set that is at least as big as the set you started with.

Example:
`{ 10*, *0*, *00 }` produces `{ 100, 101, 000, 001 }`
`{ 100, 101, 10* }` produces `{ 100, 101 }`

Notice that the set in the latter example does not satisfy the condidtion mentioned above, so it does not provide a counterexample.

You program has to check for a number of cases:

1. Whether the pattern of '*'s is different for all words in the set and:
2. Compute the number of words obtained by replacing the '*'s by '0's and '1's.

The words will not be longer than 15 symbols.

Input

The input is a text-file that presents a sequence of sets. Each set is described as follows. The first line gives two integers: the length of the words and the number of the words. Then follow the words, each on a separate line. The end of the sequence of sets is indicated by a set with wordlength 0 and number of words equal to 0.

Output

The output is a textfile that contains one line for each set. if the pattern of '*'s is different for all the words in this set this line should contain 'YES' (in uppercase), followed by a space and the number of obtained words, otherwise it should contain 'NO' (uppercase) only.

Sample Input

```
3 3
10*
*0*
*00
4 3
1100
1101
110*
0 0
```

Sample Output

```
YES 4
NO
YES 0
```