

The forest of El-Dorado has been enchanted by dark magic. Rabbit wizards live in the forest but all the paths among different wizards have been blocked by the magic. Zao, the lord of wizard rabbits, wants to lift up the curse and clean up all the dark magic from the forest, so that all the rabbits can reach each other's houses again like they used to do in the past.

Therefore, he asked for your help. You will be given an adjacency matrix that represents whether there is a bidirectional path between two rabbits or not. Determine the number of different sets of paths that can be cleaned.

## Input

First line of the input is an integer  $T$  ( $1 \leq T \leq 100$ ), where  $T$  is the number of test cases. For each test case, first you will be given an integer  $n$  ( $1 \leq n \leq 15$ ), the number of rabbits, followed by  $n$  strings containing  $n$  characters each, where  $j$ -th character of  $i$ -th element is 'Y' if there's a path between the  $i$ -th rabbit and the  $j$ -th rabbit, or 'N' otherwise. All paths are bidirectional.

## Output

For each test case, print a line in the following format: 'Case  $X$ :  $Y$ ', where  $X$  is the number of test case starting from 1 and  $Y$  is the number of different sets of paths that can be cleaned to achieve Zao's goal. Since the number can be very big, output it in *modulo* 10000.

## Sample Input

```
4
3
NYY
YNY
YYN
4
NYNN
YNY
NYNN
NYNN
4
NYYY
YNY
YYNY
YYYN
2
NN
NN
```

## Sample Output

```
Case 1: 4
Case 2: 1
Case 3: 38
Case 4: 0
```