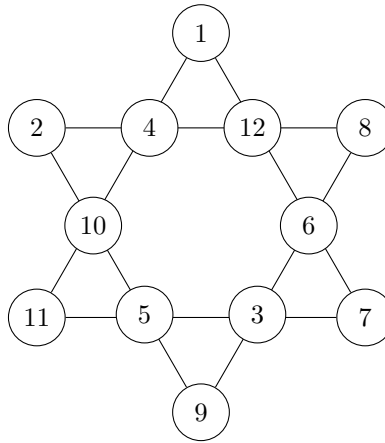


# Magic Star

A magic star consists of all the numbers from 1 to 12 arranged in the shape of a hexagram:



The magic comes from the fact that in each line of 4 numbers, the sum of the numbers is 26. In the example given above, the six lines consist of the following numbers:

- $1 + 4 + 10 + 11$
- $11 + 5 + 3 + 7$
- $7 + 6 + 12 + 1$
- $2 + 10 + 5 + 9$
- $9 + 3 + 6 + 8$
- $8 + 12 + 4 + 2$

There are several possible ways to arrange the numbers to get a magic star. Given a partially labelled star, your task is to extend the solution such that a magic star is formed.

## Input

The first line of the input contains an integer  $t$ .  $t$  test cases follow, each of them separated by a blank line.

Each test case consists of five lines containing a visualization of the star; the unlabelled fields of the star will be represented by an “x” character, and labelled fields will contain a letter between “A” and “L”, where the  $i$ -th letter in the alphabet represents number  $i$ . The character “.” is used to align the fields of the star in the shape of a hexagram. You may assume that each input will use the same alignment of the fields as the one in the sample input.

## Output

For each test case, print a line containing “Case # $i$ :” where  $i$  is its number, starting at 1. Print five more lines containing the visual representation of the lexicographically smallest extension of the given partial solution which is a magic star (lexicographically smallest means that the concatenation of the rows should result in a string which is lexicographically smaller than other potential solutions). You may assume that there is always a solution for the given input. Each line of the output should end with a line break.

## Constraints

- $1 \leq t \leq 2$

**Sample Input 1**

```
1
....X....
.A.I.D.x.
..X...x..
.x.x.x.x.
....X....
```

**Sample Output 1**

```
Case #1:
....F....
.A.I.D.L.
..H...E..
.C.J.B.K.
....G....
```