

Inconstant Ordering

Problem

We want to build a string with English alphabet uppercase letters in sorted order. However, we want the order to be sometimes strictly increasing and sometimes strictly decreasing.

The first letter of the string must be `A`. After that, the string must contain one or more blocks of letters. The i -th block must contain exactly L_i letters. Each letter in the i -th block must be later in the alphabet than its preceding letter in the string if i is odd and earlier in the alphabet than its preceding letter if i is even. Notice that for the first letter of a block, its preceding letter exists, even though it is not in the block. Strings that follow all of these rules are called valid. There can be multiple valid strings, and we want to find the alphabetically first one.

For example, if there are 2 blocks of sizes $L_1 = 2$ and $L_2 = 3$, the string must have exactly $1 + L_1 + L_2 = 1 + 2 + 3 = 6$ letters (the 1 is for the initial `A`). The strings `XYZYBA`, `AZYCBA` and `AYZYBB` are not valid for this case because they violate the required starting letter condition, and the ordering conditions in the first and second block, respectively. The string `AYZYBA` is valid. The string `ABDCBA` is also valid and, moreover, it is the alphabetically first valid string.

Given the sizes of the blocks, output the valid string that comes first in alphabetical order in the list of all valid strings. It can be shown that, for all inputs within the given limits, at least one valid string exists.

Input

The first line of the input gives the number of test cases, T . T test cases follow. Each test case is described with two lines. The first line of a test case contains a single integer N , the number of blocks. The second line contains N integers L_1, L_2, \dots, L_N , the number of letters each block must have, in order.

Output

For each test case, output one line containing `Case #x: y`, where x is the test case number (starting from 1) and y is the valid string that comes first in alphabetical order. It is guaranteed that at least one valid string exists.

Limits

Time limit: 10 seconds.

Memory limit: 1 GB.

$1 \leq T \leq 100$.

$1 \leq L_i \leq 25$, for all i .

Test Set 1 (Visible Verdict)

$1 \leq N \leq 2$.

Test Set 2 (Visible Verdict)

$1 \leq N \leq 100$.

Sample

Sample Input

```
3
2
2 3
2
5 1
1
2
```

Sample Output

```
Case #1: ABDCBA
Case #2: ABCDEFA
Case #3: ABC
```