Problem A
AB to C

Time limit: 1 second
Memory limit: 1 GB

Problem Description

You are given a string consisting only of characters A, B, or C. You can do the following operation as many times as you like:

- Choose $1 \leq i < j \leq |S|$ such that $S_i \neq S_j$.
  - Delete S_i and S_j.
  - Insert the complementary character of (S_i, S_j) back into the string at **any** position of your choice.

The complementary character of X is defined as follows:

- If $X = (A, B)$ or (B, A), the complementary character is C.
- If $X = (B, C)$ or (C, B), the complementary character is A.
- If $X = (A, C)$ or (C, A), the complementary character is B.

You are allowed to do as many operations on S as you want. Find the **lexicographically minimal** final string possible.

Input Format

- The first line of input will contain a single integer T, denoting the number of test cases.
- Each test case consists of multiple lines of input.
  - The first line of each test case contains a single integer N - the length of the string.
  - The second line of each test case contains a string S.

Output Format

For each test case, output on a new line the lexicographically minimal string possible after applying operations on S.

Constraints

- $1 \leq T \leq 10^5$
- $1 \leq N \leq 2 \cdot 10^5$
- $S_i \in \{A, B, C\}$
- $|S| = N$
- The sum of N over all test cases does not exceed 10^6.

Samples

Sample Input 1

```
3
2
BC
3
AAA
3
CAA
```

Sample Output 1

```
A
AAA
AB
```

Sample Explanation

Test Case 1 : We use one operation with $i = 1, j = 2$. We delete S_1 and S_2, making the string empty temporarily. The complementary character is A, and we insert that into the string's 1st position. Thus, we end up with A.

Test Case 2 : No operations are possible.