

1. Write a C++ program that accepts a string from the user and then replaces all occurrences of the letter e with the letter x.

2. You've given strings jewels representing the types of stones that are jewels, and stones representing the stones you have. Each character in stones is a type of stone you have. You want to know how many of the stones you have are also jewels.

Letters are case sensitive, so "a" is considered a different type of stone from "A".

Example 1:

Input: aA aAAbbbb

Output: 3

Example 2:

Input: z ZZ

Output: 0

3. There is a programming language with only four operations and one variable X:

- ++X and X++ increments the value of the variable X by 1.
- --X and X-- decrements the value of the variable X by 1.
- Initially, the value of X is 0.

Given an array of strings operations containing a list of operations, return the final value of X after performing all the operations.

Example 1:

Input: operations = ["X++", "++X", "--X", "X--"]

Output: 0

Explanation: The operations are performed as follows:

Initially, X = 0.

X++: X is incremented by 1, X = 0 + 1 = 1.

++X: X is incremented by 1, X = 1 + 1 = 2.

--X: X is decremented by 1, X = 2 - 1 = 1.

X--: X is decremented by 1, X = 1 - 1 = 0.

4. Given a string s, remove duplicate letters so that every letter appears once and only once. You must make sure your result is in the order in which the letters appear in the string.

Example 1:

Input: s = "bcabc"

Output: "bca"

Example 2:

Input: s = "cbacdcbc"

Output: "cbad"

5. You are given an alphanumeric string *s*. (Alphanumeric string is a string consisting of lowercase English letters and digits).

You have to find a permutation of the string where no letter is followed by another letter and no digit is followed by another digit. That is, no two adjacent characters have the same type.

Return the reformatted string or return an empty string if it is impossible to reformat the string.

Example 1:

Input: *s* = "a0b1c2"

Output: "0a1b2c"

Explanation: No two adjacent characters have the same type in "0a1b2c". "a0b1c2", "0a1b2c", "0c2a1b" are also valid permutations.

Example 2:

Input: *s* = "leetcode"

Output: ""

Explanation: "leetcode" has only characters so we cannot separate them by digits.

Example 3:

Input: *s* = "1229857369"

Output: ""

Explanation: "1229857369" has only digits so we cannot separate them by characters.