Problem B: Rolling Encryption

You have a sequence of lowercase characters that you want to encrypt.

The first *k* characters will be encoded as plain-text. All characters after the first *k* characters will be shifted by the most frequently occuring character that appeared in the previous *k* characters, with ties broken by the character which occurs first in the alphabet. By "shifted by", we mean that if c was the most frequently occuring character, the character would be shifted ahead by 3 positions (since c is the third letter of the alphabet), modulo 26 (e.g., b becomes e, and z becomes c).

Input Specification

On the first line of input contains *k* ($1 \leq k \leq 10\ 000$). The next line contains *c* characters ($1 \leq c \leq 100\ 000$).

Output Specification

One line, containing the encrypted version of the *c* characters from the input.

Sample Input

```
5
abbaabbacdecde
```

Output for Sample Input

```
abbaacdcdegdgh
```

Troy Vasiga