Two Characters

String t always consists of two distinct alternating characters. For example, if string t's two distinct characters are `x` and `y`, then `t` could be `xyxyx` or `yxyxy` but *not* `xxyy` or `xyyx`.

You can convert some string s to string t by deleting characters from s. When you delete a character from s, you must delete *all* occurrences of it in s. For example, if $s =$ `abaacdabd` and you delete the character `a`, then the string becomes `bcdbd`.

Given s, convert it to the longest possible string t. Then print the length of string t on a new line; if no string t can be formed from s, print 0 instead.

Input Format

The first line contains a single integer denoting the length of s.
The second line contains string s.

Constraints

- $1 \leq |s| \leq 1000$
- s only contains lowercase English alphabetic letters (i.e., `a` to `z`).

Output Format

Print a single integer denoting the maximum length of t for the given s; if it is not possible to form string t, print 0 instead.

Sample Input

```
10
beabeefeab
```

Sample Output

```
5
```

Explanation

The characters present in s are `a`, `b`, `e`, and `f`. This means that t must consist of *two* of those characters.

If we delete `e` and `f`, the resulting string is `babab`. This is a valid t as there are only two distinct characters (`a` and `b`), and they are alternating within the string.

If we delete `a` and `f`, the resulting string is `bebeeeb`. This is not a valid string t because there are *three* consecutive `e`'s present.

If we delete only `e`, the resulting string is `babfab`. This is not a valid string t because it contains *three* distinct characters.

Thus, we print the length of `babab`, which is 5, as our answer.