

Adorable



SEASON 10 - SECOND ROUND

Let's call a string *adorable* if its letters can be realigned in such a way that they form two consequent groups of equal symbols (note that different groups must contain different symbols). For example, the string *ababa* is adorable (you can transform it to *aaabb*, where the first three letters form a group of *a*-s and others — a group of *b*-s), but *abacbb* and *cccc* are not adorable.

You are given a string *S*, consisting of lowercase latin letters. Write a program, which checks whether its letters can be split into two adorable strings. All letters must be used and each of the two strings must contain at least one letter. Each letter in *S* must be put into exactly one of the two strings.

Input (adorable.in)

You have to solve *T* independent testcases in a single test. On the first line of the input file *adorable.in* you will be given the number *T*. On each of the next *T* lines you will be given a single string, consisting of lowercase latin letters.

Output (adorable.out)

For each of the strings in the input, print on a separate line in the output file *adorable.out* "Yes" (without the quotes) if the string can be split into two adorable strings, and "No" (without the quotes) otherwise.

Constraints

$$1 \leq T \leq 10$$

$$1 \leq \text{Number of letters in each string} \leq 50\,000$$

Example

Input	Output	Explanation
3	Yes	<i>ababa</i> → <i>aba</i> , <i>ba</i> → <i>aab</i> , <i>ba</i>
ababa	Yes	<i>zxcxx</i> → <i>zc</i> , <i>zxx</i>
zxcxx	No	<i>yeee</i> can't be split into two adorable strings.
yeee		