String-o-Permute

Kazama gave Shaun a string of even length, and asked him to swap the characters at the even positions with the next character. Indexing starts at 0.

Formally, given a string *str* of length L where L is even, Shaun has to swap the characters at position i and $i+1$, where $i \in \{0, 2, .., L-2\}$.

For example, if *str = "abcdpqrs"*, $L = 8$. We have to swap the characters at positions:
$\{(0,1), (2,3), (4,5), (6,7)\}$

So, answer will be *"badcqpsr"*.

Input Format

The first line contains an integer, T, the number of test cases.
T lines follow, each containing some string *str*.

Output Format

For each test case, print the new string as explained in the problem statement.

Constraints

$1 \leq T \leq 10$
$1 < L \leq 10^5$
L is even
str consists of lowercase English characters, $\{a - z\}$.

Sample Input

```
2
abcdpqrs
az
```

Sample Output

```
badcqpsr
za
```

Explanation

Test case #00: This is the same example as mentioned in the problem statement.
Test case #01: Here L is 2, so we have to swap the characters at position $(0,1)$ only.