Merge the Tools!

Consider the following:

- A string, s, of length n where $s = c_0c_1 \ldots c_{n-1}$.
- An integer, k, where k is a factor of n.

We can split s into $\frac{n}{k}$ substrings where each subtring, t_i, consists of a contiguous block of k characters in s. Then, use each t_i to create string u_i such that:

- The characters in u_i are a subsequence of the characters in t_i.
- Any repeat occurrence of a character is removed from the string such that each character in u_i occurs exactly once. In other words, if the character at some index j in t_i occurs at a previous index $< j$ in t_i, then do not include the character in string u_i.

Given s and k, print $\frac{n}{k}$ lines where each line i denotes string u_i.

Example

$s = \text{'AAABCADDE'}$

$k = 3$

There are three substrings of length 3 to consider: 'AAA', 'BCA' and 'DDE'. The first substring is all 'A' characters, so $u_1 = \text{'A'}$. The second substring has all distinct characters, so $u_2 = \text{'BCA'}$. The third substring has 2 different characters, so $u_3 = \text{'DE'}$. Note that a subsequence maintains the original order of characters encountered. The order of characters in each subsequence shown is important.

Function Description

Complete the merge_the_tools function in the editor below.

merge_the_tools has the following parameters:

- string s: the string to analyze
- int k: the size of substrings to analyze

Prints

Print each subsequence on a new line. There will be $\frac{n}{k}$ of them. No return value is expected.

Input Format

The first line contains a single string, s.
The second line contains an integer, k, the length of each substring.

Constraints

- $1 \leq n \leq 10^4$, where n is the length of s
- $1 \leq k \leq n$
- It is guaranteed that n is a multiple of k.

Sample Input

```
STDIN       Function
-----       --------
AABCAAADA   s = 'AABCAAADA'
3           k = 3
```

Sample Output

```
AB
CA
AD
```

Explanation

Split s into $\frac{n}{k} = \frac{9}{3} = 3$ equal parts of length $k = 3$. Convert each t_i to u_i by removing any subsequent occurrences of non-distinct characters in t_i:

1. $t_0 = \text{"AAB"} \rightarrow u_0 = \text{"AB"}$
2. $t_1 = \text{"CAA"} \rightarrow u_1 = \text{"CA"}$
3. $t_2 = \text{"ADA"} \rightarrow u_2 = \text{"AD"}$

Author anuj_95
Difficulty Medium
Max Score 40
Submitted By 236556

NEED HELP?

View discussions
View editorial
View top submissions

RATE THIS CHALLENGE

MORE DETAILS

Download problem statement
Download sample test cases
Suggest Edits

Print each u_i on a new line.

Change Theme Language Pypy 3

```
def merge_the_tools(string, k):
    # your code goes here

if __name__ == '__main__':
    string, k = input(), int(input())
    merge_the_tools(string, k)
```

Line: 6 Col: 31

Upload Code as File Test against custom input Run Code Submit Code