

题目描述

给你一个包含字母串 S，将其 N 个 1/分桶或 N+1 的子串，给定正整数 K，要求将第一个子串中，其余的串每 K 个用 1/分隔，并得小写字母替换为 1/。

输入描述

正整数 K 和 1/分隔的字符串，如：

2
2503C-abc-d

输出描述

转换后的字符串

用例

输入	4 2512-2a-6-w
输出	2512-2a-6-w
说明	字符串 S 被分成了两个部分，每部分 4 个字符； 注意，两个部分的数字符号需要删除。
输入	2 2503-3-1
输出	2503-3-1
说明	字符串 S 被分成了 3 个部分，每部分 3 个字符，第一部分的字符串以 1/子字符串的末尾，其余部分为 2 个字符。

题目解析

题解的 字符串操作

JavaScript 算法源码

```
1 // 题目描述: 给你一个包含字母串 S，将其 N 个 1/分桶或 N+1 的子串，给定正整数 K，要求将第一个子串中，其余的串每 K 个用 1/分隔，并得小写字母替换为 1/。
2 const readline = require("readline");
3 const rl = readline.createInterface({
4   input: process.stdin,
5   output: process.stdout,
6 });
7
8 const lines = [];
9 rl.on("line", (line) => {
10   lines.push(line);
11 });
12
13 if (lines.length === 1) {
14   const k = parseInt(lines[0]);
15   const arr = lines[1].split("");
16   console.log(convert(arr, k));
17   lines.length = 0;
18 }
19
20 function convert(arr, k) {
21   const first = arr.shift();
22   const ans = arr;
23   const res = [];
24   for (let i = 0; i < ans.length; i++) {
25     const v = ans[i];
26     if (i % k === 0) {
27       v = v.toUpperCase();
28     }
29     res.push(v);
30   }
31   return res;
32 }
33
34 ans.unshift(first);
35 return ans.join("");
36 }
```

Java 算法源码

```
1 import java.util.Scanner;
2
3 public class Main {
4   public static void main(String[] args) {
5     Scanner sc = new Scanner(System.in);
6
7     int k = sc.nextInt();
8     String str = sc.next();
9
10    System.out.println(getResult(k, str));
11  }
12
13   public static String getResult(int k, String str) {
14     String[] arr = str.split("");
15
16     String first = arr[0];
17
18     StringBuilder sb = new StringBuilder();
19     for (int i = 1; i < arr.length; i++) {
20       sb.append(arr[i]);
21     }
22     String[] res = sb.toString().split("");
23
24     StringBuilder upper = new StringBuilder();
25     upper.append(first);
26     for (int i = 0; i < res.length; i++) {
27       String v = res[i].toUpperCase();
28       if (i % k === 0) v = v.toUpperCase();
29       upper.append(v);
30     }
31     return upper.toString();
32   }
33 }
```

Python 算法源码

```
1 k = int(input())
2 s = input()
3
4
5
6 def getResult():
7     arr = s.split("")
8
9     ans = [arr.pop(0)]
10
11     res = list("")
12     for i in range(len(arr)):
13         v = arr[i].upper()
14         if i % k == 0:
15             v = v.upper()
16         ans.append(v)
17
18     return "".join(ans)
19
20 if __name__ == '__main__':
21     print(getResult())
```