



链滴

左旋转字符串

作者: [yudake](#)

原文链接: <https://ld246.com/article/1519021713058>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

题目描述

汇编语言中有一种移位指令叫做循环左移（ROL），现在有个简单的任务，就是用字符串模拟这个指的运算结果。对于一个给定的字符序列S，请你把其循环左移K位后的序列输出。例如，字符序列S="bcXYZdef"，要求输出循环左移3位后的结果，即"XYZdefabc"。是不是很简单？OK，搞定它！

解题思路

- 如果可以用substring()，直接用substring()；
- 如果不可以，用字符数组替换移位。原理： $YX = (X.T Y.T).T$

代码

代码1

```
public class Solution {
    public String LeftRotateString(String str,int n) {
        if (str == null || str.length() == 0)
            return "";
        int length = str.length();
        n = n % length;
        String right = str.substring(0, n);
        String left = str.substring(n, length);
        return left + right;
    }
}
```

代码2

```
public class Solution {
    public String LeftRotateString(String str,int n) {
        if (str == null || str.length() == 0)
            return "";
        char[] cs = str.toCharArray();
        n %= cs.length;
        swap(cs, 0, n);
        swap(cs, n, cs.length);
        swap(cs, 0, cs.length);
        return String.valueOf(cs);
    }

    private void swap(char[] cs, int start, int end) {
        for (int i = start; i < (start+end)/2; i++) {
            char tmp = cs[i];
            cs[i] = cs[end+start-1-i];
            cs[end+start-1-i] = tmp;
        }
    }
}
```