

## 团队

2020.05.03 [2019-2020 ICPC Asia Taipei-Hsinchu Regional Contest](#) pro: 11/11/13 rk: 9/488

## 个人

**zzh**

2020.05.06 [Codeforces Round 639 \(Div. 1\)](#) pro:6/6 **DONE**

2020.04.15 [Codeforces Round 635 \(Div. 1\)](#) pro:6/7

**pmxm**

**jsh**

个人周报：[2020.05.01-2020.05.07\\_周报](#)

## 本周推荐

**zzh**

**pmxm**

**jsh**

[Codeforces Round #393 \(Div. 1\): D. Bacterial Melee](#)

题意：

给一个字符串，字符可以把相邻的字符变得和自己一样，随便进行这样的操作若干次，问最后有多少种不同的字符串。

题解：

会发现最终的字符串中，字符的顺序应当和给定的字符串中字符的顺序一样，只不过给定的字符串有的字符跳过去了（吞没了）。问题就可以视为，每次插一个字符，该字符可以作为后缀出现  $0, 1, \dots$  若干次，维护方案数。

那我们可以记  $f(c, i)$  为以字符  $c$  作为最后一个字符的情况下，长度为  $i$  的字符串有多少种。记当前插入的字符是  $u$  那么  $f$  只有  $f(u, *)$  被改变了，值变为  $f(u, i) \gets \sum_{v \neq u} \{ \sum_{j < i} f(v, j) \}$  再记一下  $f(i, *)$  的前缀和，就可以每次以  $\mathcal{O}(n)$  的时间复杂度更新  $f$

答案是  $\sum_u \{ f(u, n) \}$  总时间复杂度  $\mathcal{O}(n^2)$

From: <https://wiki.cvbbacm.com/> - **CVBB ACM Team**

Permanent link: [https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:intrepidword:2020.05.01-2020.05.07\\_%E5%91%A8%E6%8A%A5&rev=1588955961](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:intrepidword:2020.05.01-2020.05.07_%E5%91%A8%E6%8A%A5&rev=1588955961) 

Last update: **2020/05/09 00:39**