

# NOI 模拟赛解题报告

by samjia2000

**say**

**“对于 30%的数据，字符串的总长度 $\leq 5000$ ”**

DP，枚举两个串 A 和 B，设  $f[i][j]$  表示 A 中以第 i 位开始的字符串变成 B 中以第 j 为开始的字符串的最小次数

**“对于 40%的数据，字符串的总长度 $\leq 12000$ ”**

跟 30%的一样，数组滚动就好了。

**“对于 60%的数据，字符串的总长度 $\leq 100000$ ”**

所以 DP 时很多状态都是无用的，因为如果两个串的长度差大于 8 那么对答案没有贡献

**“对于另外 20%的数据， $1 \leq n \leq 70$ ”**

其实这档跟正解没有什么区别，好像也是可以过掉 100 分的，没有实测，但是分分钟跑的比正解快？！

由于只要求相似度小于等于 8 的字符串对数，那么考虑递归的计算两个串的相似度，对于串 A 和 B，假设当前我们已经匹配到 x 和 y (x 是 A 中的位置, y 是 B 中的位置)，那么首先跳过以这两个位置开头的 LCS，然后三种操作继续递归下去，如果答案大于 8 就退出。

求 LCS 的时候就用 hash+二分

**“对于 100%的数据， $1 \leq n \leq 200$ ，字符串的总长度 $\leq 1000000$ ”**

其实我也不是很懂，怎么玄学就跑过去了呢？std 的做法是...跳 LCS 的时候一个个跳...详情见标程...不要打我...wo shi la ji



see

关键字：常系数齐次线性递推，分治，FFT

**“对于 20% 的数据,  $1 \leq n \leq 20$ ”**

直接暴力就好了。

**“对于另外 10% 的数据满足  $f(0)=f(1)=1$ ”**

这档是给 OEIS 和找到某种神奇规律的选手的，然而我不会

**“对于另外 20% 的数据满足所有  $a[i]$  的和  $\leq 1000, 1 \leq n \leq 100$ ”**

DP，设  $f[i][j][x]$  表示前  $i$  个数里面选了  $j$  个和为  $x$  的方案数，直接算。

**“对于另外 20% 的数据满足  $1 \leq n \leq 5000$ ”**

$$f(i) = 2 * f(i-1) + 3 * f(i-2)$$

显然  $f$  是一个常系数齐次线性递推的形式，考虑求第  $n$  项大概是这样一个过程：

首先，求出  $x^n$  对  $M(x) = x^2 - 2x - 3$  取模的结果，设为  $\text{Res}(n) = k[n] * x + b[n]$ （注： $\text{Res}(n)$  为多项式）

然后  $f(n)$  即为  $k[n] * f[1] + b[n] * f[0]$

考虑计算  $f[a+c] + f[b+c]$

即求出  $\text{Res}(a+c) + \text{Res}(b+c)$

那么由于  $\text{Res}(a+c) = \text{Res}(a) * \text{Res}(c) \pmod{M(x)}$

得到  $\text{Res}(a+c) + \text{Res}(b+c) = (\text{Res}(a) + \text{Res}(b)) * \text{Res}(c) \pmod{M(x)}$

得到这个结论后我们可以有一个  $O(n^2)$  的 DP：

设  $f[i][j]$  表示前  $i$  个数里面选了  $j$  个的所有方案的 **Res 的和**，也就是说  $f[i][j]$  是一个一次多项式，那么转移就是  $g[i][j] = g[i-1][j] + g[i-1][j-1] * \text{Res}(a[i]) \pmod{M(x)}$

最后利用  $g[n][k]$  得到答案

## 进阶满分

上面比较麻烦的是有一个一次多项式，那么如何去掉这个一次多项式的影响呢？

考虑  $ax+b$  和  $cx+d$  对  $M(x)$  取模的答案

$$(ax+b)*(cx+d)=(ad+bc+2ac)x+(bd+3ac) \pmod{M(x)}$$

于是定义一类新的数，称之为 One-Degree Number（当然这是我瞎起的名字，下文称之为 ODN）

定义 ODN  $t=(a,b)$  表示  $t=ax+b$ ，定义乘法  $(a,b)*(c,d)=(ad+bc+2ac, bd+3ac)$ ，加法与减法显然（没有必要使用除法所以不讨论 ODN 的除法）

那么  $\text{Res}(a[i])$  也可以看做是一个 ODN

设  $G_i(y)$  表示  $g[i][j]$  的生成函数，那么  $G_i(y)=G_{i-1}(y)*(1+\text{Res}(a[i])*y)$

这个像什么？分治 FFT!!!

如果我们可以做 ODN 的 FFT，那么我们就可以利用分治 FFT 算出  $G_n(y)$ ，利用其第  $k$  项算出答案。

怎么做 ODN 的 FFT？

这个不是传统的实数，但是有一个东西是和实数域相同的，单位元都是 1！！

这就说明我们可以直接用主  $n$  次单位复数根来进行运算，即使进行 FFT 的数是 ODN！！

但是要注意的是进行 FFT 时要重定义数的类型，详情见标程。

时间复杂度  $O(n (\log n)^2)$

其实本题可以用二维的 FFT 来实现，而常系数线性递推的式子也可以不止 2

**do**

**“对于 20%的数据， $1 \leq n, q, S \leq 3000$ ”**

直接暴力就好了。

**“对于另外 30%的数据， $1 \leq n, q, S \leq 100000$  且每组数据的  $k \leq 5$ ”**

这档其实我不会，但是可以参考这道题：

<https://www.hackerearth.com/practice/algorithms/graphs/depth-first-search/practice-problems/algorithm/minimum-distance-1/>

PS.事实上，我就是在想 Minimum distance 这题的时候想到 do 这题的做法，但是发现自己并没有用它给出  $k \leq 5$  的特殊条件就决定出出来，然后把原题出成部分分，考虑到这一档也曾是一道题，所以就给了 30 分...

**“对于另外 10%的数据，给出的树是一条链”**

直接瞎搞搞

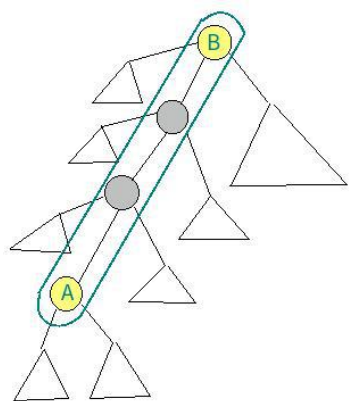
**“对于另外 20%的数据， $1 \leq q \leq 10$ ”**

对于每次询问把整棵树都扫一次就好了。

**“对于 100%的数据， $1 \leq n, q, S \leq 100000$ ”**

很容易想到虚树的套路，但是问题在于如何处理虚树上两个相邻关键点之间的那一段的贡献。

就是说，要考虑虚树上类似这样的段的贡献：



关键点是 A 和 B，对每个出 A,B 之间的点  $x$ ，我们要求出以  $x$  为起点，不经过 A、B 之间的点所能到达的最远的点的距离。

那么这个如何处理呢？

考虑使用 LCT，对于每个点  $x$  维护一个权值表示不经过与  $x$  同处一条重链的点所能到达的最远的点的距离，再对每个点用 set 维护虚边的信息。

每次 link 和 cut 的时候都要注意更改一下 set 和点的值。

询问两个关键点的时候就先对 A makeroot 然后对 B access，接着可以在这一段上二分出一个分界点，分别计算贡献就好了。

时间复杂度  $O(n(\log n)^2)$

所以其实这题还可以动态，做法类似，要打多一个 splay 来维护括号序。

而静态，即本题，可以用树链剖分做到  $O(n \log n)$