随机串生成器 解题报告

绍兴市第一中学 任之洲

1 试题来源

2016年集训队互测

2 试题大意

设字符集大小为c，$S_c(L)$为所有长度为L的串的集合。
对于一组固定的c, L，随机了n次，每次随机返回一个集合$S_c(L)$中的串。
设得到的n个串为$s_1 \sim s_n$，并利用以下方式评估这一次随机：

- 定义$a + b$为两个串a, b拼接后得到的串，如‘sin’ + ‘on’ = ‘sinon’。
- 定义两个串a, b相似，需要满足a是$b + b$的子串且b是$a + a$的子串。
- 设$count(s)$为满足$1 \le i < j \le n$且s_i和s_j相似的数对(i, j)数量。
- 评估结果为$G(count(s))$，其中$G(x)$为一个给定的函数。

函数$G(x)$以其在$x \in \{0, 1, 2, \ldots, \frac{n(n-1)}{2}\}$的点值形式给出。
求所有c^{nL}种生成情况的$G(count(s))$之和，答案对$10^9 + 7$取模。

2.1 数据规模与约定

对于10%的数据，$G(x) = 1$。
对于另20%的数据，$G(x) = x$。
对于50%的数据，$n \le 10$。
对于70%的数据，$n \le 20$。
对于80%的数据，$n \le 30$。

对于90%的数据，$n \le 40$。

对于100%的数据，$n \le 50$，$1 \le L \le 10^6$，$1 \le c \le 10^9$，$0 \le$ 输入点值 $\le 10^4$。

每一档数据中n均有一定梯度。

总共有50组测试点，在后35组中均匀分布着17个测试点满足输入的点值中只有$G(0) \neq 0$。

3 算法介绍

为了方便后文描述，设d表示L的约数个数，当$1 \le L \le 10^6$时，d最大为240。

3.1 算法一

对于10%的数据，$G(x) = 1$。

在这种情况中，每一种生成情况对答案的贡献都是一样的。

直接用快速幂计算c^{nL}输出。

期望得分10。

3.2 “相似串”的计数

首先需要解决一个重要的子问题：在确定的c, L下，有多少对串a, b相似。

容易发现，对于一个串a，与它相似的串个数等于它的循环节长度，例如：串‘abaaba’的循环节为3，与它相似的串为‘aabaab’、‘abaaba’、‘baabaa’。

根据这个性质，考虑计算循环节长度恰好为k的串个数，设为g_k。

循环节长度一定是L的约数，所以只需要计算d组g_k的值，并且有

$$g_k = c^k - \sum_{j|k} g_j$$

根据这个式子可以$O(d \log L + d^2)$递推计算所有需要被用到的g_k。

对于固定的循环节长度k，每一组两两相似的串集大小均为k，所以互不相似的串有$\frac{g_k}{k}$种。

考虑计算n个串两两相似的方案数，设为$f(n)$，有

$$f(n) = \sum_{k|L} k^{n-1} g_k$$

找到这d个约数以及计算g_k和$f(n)$的复杂度为$O(\sqrt{L}+d\log L+d^2)$，后文讨论时将忽略这一部分的复杂度。

3.3 算法二

对于另20%的数据，$G(x)=x$。

每一对相似的串都会对答案有1的贡献，考虑强制某一对串相似，剩下$n-2$个串的状态可以随意分配，即$\frac{n(n-1)}{2}f(2)c^{(n-2)L}$。

期望得分20。

通过特判输入的$G(x)$，和算法一结合可以得到30分。

3.4 算法三

在后35组中均匀分布着17个测试点满足输入的点值中只有$G(0)\neq 0$。

也就是说，不允许存在任何一对相似的串。

如果两个串的循环节长度不同，它们一定不可能相似，所以可以将每种循环节长度分开考虑。

先计算出u个循环节长度为k的互不相似的串的方案数

$$k^u\prod_{i=0}^{u-1}\left(\frac{g_k}{k}-i\right)$$

最后用类似背包的算法就可以求出n个串互不相似的方案数，注意合并两个串集时需要用组合数分配标号。

时间复杂度$O(dn^2)$，期望得分34。

和算法一结合实际可以得到66分（可以意外地多通过一个$n=1$的点）。

3.4.1 容斥算法

经典的容斥算法也可以解决这一个subtask。

问题可以转化为$\frac{n(n-1)}{2}$对限制关系，每对限制关系为某两个串不能相似。

容斥时需要枚举其中$k(0\leq k\leq\frac{n(n-1)}{2})$对限制关系强制违反，即这几对串强制相似，并将对应的方案数乘上系数$(-1)^k$累计入答案。

把限制关系看作一张无向完全图，直接枚举边是$O(2^{\frac{n(n-1)}{2}})$的，而注意到我们其实只关心连通块关系以及边数的奇偶性。

设$F_0(n), F_1(n)$为n个点边数为偶数和奇数的连通图数量，$G_0(n), G_1(n)$为n个点边数为偶数和奇数的图数量，对于$n > 1$的情况有

$$G_0(n) = G_1(n) = 2^{\frac{n(n-1)}{2}-1}$$

$$\begin{aligned} F_0(n) &= G_0(n) - \sum_{i=1}^{n-1} \binom{n-1}{i-1} (F_0(i)G_0(n-i) + F_1(i)G_1(n-i)) \\ F_1(n) &= G_1(n) - \sum_{i=1}^{n-1} \binom{n-1}{i-1} (F_0(i)G_1(n-i) + F_1(i)G_0(n-i)) \end{aligned}$$

以上的递推关系可以理解为枚举1号点所在连通块大小，将不连通的图从方案中去掉。

完成这一部分递推后将连通块背包起来，同时再乘上相关的容斥系数及整个连通块相似的方案数就可以完成计算，时间复杂度$O(n^2)$。

3.5 算法四

考虑将算法三一般化，我们只需要计算出u个循环节长度为k的串，相似对数为v的方案数，用$O(n^6)$的二维卷积可以完成背包计算。

仍然每种循环节长度分开考虑，影响方案数计算的主要有下面几个量：

- 串个数u，相似对数v，这两个量直接影响到该状态对答案的贡献系数。
- 按照相似关系将u个点划分为一些块，设块的个数为l，为了使得这些块两两不相似，需要计算系数$\prod_{i=0}^{l-1} \left(\frac{g_k}{k} - i\right)$。

设$t[l][u][v]$表示已经确定了l个块，u个串，相似对数为v的方案数，转移时可以枚举下一个块的大小c。

为了防止重复计算，强制标号最小的串一定在新加的块中，转移系数为

$$k^c \binom{u+c-1}{c-1} \left(\frac{g_k}{k} - l\right)$$

这一部分的计算是$O(n^5)$的，瓶颈在于前面的二维卷积。

时间复杂度$O(dn^6)$，期望得分50~70。

3.6 算法五

算法四的瓶颈在于$O(n^6)$的二维卷积，虽然模数10^9+7并不适合使用快速傅立叶变换，但可以使用其他插值算法，例如拉格朗日插值和牛顿插值。

由于有n个$O(n^2)$阶的多项式需要被插值，所以这部分的复杂度为$O(n^5)$。

时间复杂度$O(dn^5)$，期望得分60~70。

虽然降低了复杂度，但实际运行效率并没有显著的提升。

和算法三结合可以得到80~90分。

3.7 算法六

对于一种串的生成情况，它对答案的贡献只与相似对数相关，而相似具有传递性，所以可以将n个串划分为一些集合，每个集合中的串两两相似。

由于不需要区分串的顺序，所以枚举集合划分情况的状态数为拆分数，即枚举一个正整数序列a，设长度为m，满足$a_i \le a_{i+1}$，且$\sum_{i=1}^{m} a_i = n$，按照序列a将n个串分为m块。

$$a = \{a_1, a_2, \ldots, a_m\}$$

对于一种拆分状态a，先考虑它可以对应到多少种原序列中的相似情况，设b_j为大小为j的块的个数，我们认为a_i相同的块是有序的。

首先需要将每一个块对应到原序列中，这一部分的系数为

$$\frac{n!}{\prod_{i=1}^{m} a_i!}$$

因为a_i相同的块是有序的，所以还要考虑这些块的顺序问题，故这一种拆分状态a的计算次数为

$$\frac{n!}{\left(\prod_{i=1}^{m} a_i!\right)\left(\prod_{j=1}^{n} b_j!\right)}$$

考虑如何计算一种拆分状态a对应的串相似情况的方案数。

- 先计算出$a' = \{a_1, a_2, \ldots, a_{m-1}\}$的方案数，乘上最后一个块的贡献$f(a_m)$。
- 上面这样会计算到一些不合法的情况，即第m块可能会与其他第$i(i \ne m)$块相似。由于已经保证了前$m-1$块互不相似，所以可以枚举第i块和第m块相似，计算$a'' = \{a_1, a_2, \ldots, a_{i-1}, a_{i+1}, \ldots, a_i + a_m\}$的方案，从答案中减掉。

上面这个流程中，每一步都会减少m或减少a_i的和，所以可以通过记忆化完成计算，状态的检索可以用字母树$O(n)$完成。

状态数为$1 \sim n$的拆分数之和，设为S，当$n \le 50$时S不超过1300000。

时间复杂度$O(n^2S)$，空间复杂度$O(nS)$，期望得分100。

4 总结

这道题主要分为以下几个subtask：

- $G(x) = 1$（10分）
  所有状态对答案的贡献系数相同，只需要输出整个集合的大小的n次幂。
- $G(x) = x$（20分）
  每一对相似的串会对答案造成一个单位的贡献，可以对每一对串计算，只要解决了相似串的计数问题就容易完成这个subtask。
- 输入的点值中只有$G(0) \neq 0$（34分）
  不允许出现任何一对相似的串，可以通过简单的DP计数解决，也可以使用经典的容斥算法。
- 没有特殊的限制与约定（36分）
  需要解决一般化的情况，可以从前一个subtask的算法推广但复杂度较高，通过所有测试点需要基于拆分数进行计算。

5 得分估计

- 预计所有选手都能拿到10分以上。
- 8 ~ 12名选手得到30分以上。
- 4 ~ 8名选手得到60分以上。
- 1 ~ 3名选手得到100分。