NOI 2019天津市代表队选拔赛 day1题解

A 甲苯先生的字符串

$dp[i][j]$表示长度为i的字符串以j结尾时的合法方案数，$w[i][j]$表示字符串ij能否出现。
dp状态转移方程表示:
$dp[i][j] = \Sigma w[k][j] * dp[i-1][k]$
对于100%的数据，考虑到转移方程是个矩阵，可以使用矩阵快速幂加速运算。

B 甲苯先生的滚榜

简单来说类似于桶排序。 在题面中说罚时不超过1500000其实已经给了提示这个题可以按照罚时来建立线段树进行排序。

对于每个user来说，比他排名高的人来源于两个地方：第一是题目数更多，第二是罚时更少。所以我们可以根据题目数量建立树状数组，去维护ac题目数量的前缀和。对于题数相同的同学，我们就可以用线段树去维护罚时。但是我们发现 1500000∗ 平均ac数量 ∗4 是会MLE的，所以我们这时需要动态建立线段树（类似于主席树的方法）

所以对于每一次询问，时间复杂度为约 $log(1500000) + log(n/m)$ 。 空间复杂度为 $log(1500000) + (n/m)$. 注: n/m代表近似于ac题目数量，直接开一个比较大的数组也可以做。

常数比较小的splay也可以过这道题的绝大部分样例点。

C 唱、跳、rap和篮球

由于这是一个多重集的排列问题，在不考虑非法排列的情况下，我们可以利用指数生成函数来解决。
现在考虑如何从结果中去掉非法的排列:
这里我们考虑容斥，
我们可以把最喜欢唱的人当作a，最喜欢跳的人当作b，最喜欢rap的人当作c，最喜欢篮球的人当作d，
这个题就变成了，用给定数量的a, b, c, d来填充一个给定长度的序列，要求序列中不得出现子串abcd 我们可以先计算不考虑序列中有子串abcd的结

果，假设这个结果是f_0, 然后我们可以依次计算序列中有至少1个字串$abcd$，至少2个子串$abcd$,……一直到有至少$k(k = n/4)$个子串$abcd$的结果，并分别假设它们为f_1, f_2,..., f_k 由于$m0$中含有非法的序列，所以我们应该减去含有至少一个子串abcd的数量f_1，但因为对于每一个含有至少两个子串的abcd的排列，我们在f_1 中计算了两遍，所以还应该加上f_2， 这样我们就形成了一个容斥，最终的结果就是$f_0 - f_1$+...+$(-1)^k f_k$。 由于a, b, c, d的数量在题目中是给定的，因此我们不能直接推出公式，只能直接通过生成函数相乘，这样的复杂度是$O(n^{2k})$，k为容斥的次数， 由于是计算的过程是多项式相乘，我们可以使用fft加速，不过本题的数据范围并没有要求完成这一步。 题目中有20%的数据是$n = a = b = c = d$的，这部分数据我们可以把它看作是a, b, c, d均有无限多个的情况，这种情况下我们很容易用指数生成函数推出公式进行容斥，复杂度可以降到$O(k)$，k为容斥的次数。