Super League of Chinese College Students Algorithm Design 2021 Contest 10

August 19th, 2021

1 A. Pty loves sequence

如何不重不漏地统计好的序列？

设最大值 $=k$ ，我们可以取最前的一个 $[1,2,\&,k-1,k]$ 子序列。

再考虑往它们之间的空插数，不难发现，要使该子序列是最前的，唯一的限制条件是：

对于 $x-1$ 和 $x(1\le x\le k)$ 之前那个空，里面不能有 x 。

特别的，最后一个空没有限制。

那么只需要枚举最后一个空的填的数的个数 i ，由挡板法可得填数方案是 $k^i*(k-1)^{n-k-i}*\binom{n-k-i+k-1}{k-1}$

第一问的答案 $Ans=\sum_{k=1}^{n}\sum_{i=0}^{n-k}k^i*(k-1)^{n-k-i}*\binom{n-i-1}{k-1}$

第二问的做法可以由第一问得到。

比较显然的一个结论：

> 当 k 固定时， $1..k$ 在好的序列中的出现次数是相同的。

简略证明：

> 第一问的做法中，当 k 固定时，限制为第 i 个空不能填 i ，最后一个空随意。
> 因为对 $1..k$ 每个元素的限制是一样的，所以 $1..k$ 本质相同，因此出现次数应当相同。

所以枚举 k ，使 $ans[1..k]+=(\sum_{i=0}^{n-k}k^i*(k-1)^{n-k-i}*\binom{n-i-1}{k-1})*\frac{n}{k}$ 。

最后的问题在于本题的模数不是质数，所以没有逆元，因此需要把 $\frac{n}{k}$ 化进式子里。

观察式子，不难得到：

当 $i>0$ 时，把 k^i 变成 k^{i-1} 即可。

当 $i=0$ 时，把 $\binom{n-1}{k-1}*\frac{n}{k}$ 变成 $\binom{n}{k}$ 即可。

那么式子里只用到了幂和组合数，组合数可以杨辉三角预处理，所以模数不需要是质数。

时间复杂度： $O(n^2)$

事实上当 $p=998244353$ 时，这题可以做到： $O(n\sqrt{n\,log\,n})$ ，需要用到多项式多点求值。

因为该做法常数颇大，且代码较长，不适合于ACM比赛，所以善良的出题人并没有放这个版本。

2 B. Pty with card

我们约定一轮抽牌，指在场的所有人都顺次抽了一次牌；定义局面就是圈内所有人的手牌按顺序构成的一个数列。

我们从一个局面将会如何变化入手。那么一开始的局面是 $(1,1,1,\cdots,1,1)$ ，一轮抽排后，局面变为 $(3,2,2,2,\cdots,2,2)$ 或 $(4,2,2,2,\cdots,2)$ ，剩下的人数一定是 $\lfloor\frac{N-1}{2}\rfloor$ 。

如果某个局面剩下 $2m$ 个人，我们发现，在某一个人出局之前，每一个人的手牌一定每轮都+1要么每轮都-1。

如果某个局面剩下 $2m+1$ 个人，经过两轮之后，如果没有人出局，那么每个人的手牌数目都不变，这是造成循环的原因，循环节的大小就是 $4m+2$ 。

手玩几个数据，我们发现几点性质：

1. 某个局面剩下 $2m$ 个人，那么有人出局的那一轮之后，必然只剩下个 m 人。并且局面满足如下形式 $(P+1,P,P,P,\cdots,P,P)$ ，或 $(P+2,P,P,P,\cdots,P,P)$ ，而是前者还是后者只与初始的 N 的奇偶性有关，其中 $P=2^k,k\in\mathbb{Z}^+$ 。 2. 在第一轮之后，某个局面剩下 $2m+1$ 个人，必然出现循环节。

用数学归纳法是比较好证明的：

我们称 m 个人的初始局面，是经过最早的一轮之后剩下 m 个人的局面。若该局面满足 $(P+1,P,P,P,\cdots,P,P)$ 形式，则称此为 A 形式；若该局面满足 $(P+2,P,P,P,\cdots,P,P)$ 形式，则称此为 B 形式。 $(P=2^k,k\in\mathbb{Z}^+)$

从第二轮开始考虑。我们现在证明的是，若 N 为奇数，则任意可达的初始局面，满足 $(P+1,P,P,P,\cdots,P,P)$ 形式，第一次抽牌一定是抽2张；若 N 为偶数，则初始局面满足满足 $(P+2,P,P,P,\cdots,P,P)$ 形式，第一次抽牌一定是抽1张。令这个命题为 p 。

一开始的局面有个 $n=\lfloor\frac{N-1}{2}\rfloor$ 人，满足命题 p 。

当该局面有 $2m$ 个人，且满足命题 p 时，显然有 $P\geq 2$ 。

1. 若 N 为奇数，初始局面是 $(P+1,P,P,P,\cdots,P,P)$ ， $P+1\geq 3$ ，第一次抽排是抽2张。每轮之后，奇数位的值会-1，偶数位的值会+1。直到局面变为 $(2,2P-1,1,2P-1,1,2P-1,\cdots,1,2P-1)$ ，这期间不会有某一个人的手牌归零。再过一轮，奇数位的值将会都变为0，这些人出圈，局面变为 $(2P+1,2P,2P,2P,\cdots,2P,2P)$ ，接下来该抽2张牌，因此满足命题 p 。 2. 若 N 为偶数，初始局面是 $(P+2,P,P,P,\cdots,P,P)$ ， $P+1\geq 4$ ，第一次抽排是抽1张。每轮之后，偶数位的值会-1，奇数位的值会+1。直到局面变为 $(2P+1,1,2P-1,1,2P-1,\cdots,2P-1fi1)$ ，这期间不会有某一个人的手牌归零。再过一轮，偶数位的值将会都变为0，这些人出圈，局面变为 $(2P+2,2P,2P,2P,\cdots,2P,2P)$ ，接下来该抽1张牌，因此满足命题 p 。

而当该局面有 $2m+1$ 个人，且满足命题 p ，显然有 $P\geq 2$ 。

1. 若 N 为奇数，局面是 $(P+1,P,P,P,\cdots,P,P)$ ，先抽2张牌。一轮之后局面变为 $(P+1,P+1,P-1,P+1,P-1,\cdots,P+1,P-1)$ ，因为 $P\geq 2$ 故不会有数归零。下一轮将先抽1张牌，一轮之后的局面变为 $(P+1,P,P,P,\cdots,P,P)$ ，出现循环。 2. 若 N 为偶数，局面是 $(P+2,P,P,P,\cdots,P,P)$ ，先抽1张牌。一轮之后局面变为 $(P+2,P-1,P+1,P-1,P+1,\cdots,P-1,P+1)$ ，因为 $P\geq 2$ 故不会有数归零。下一轮将先抽2张牌，一轮之后的局面变为 $(P+2,P,P,P,\cdots,P,P)$ ，出现循环。

证毕。

所以对于 N ，若 $N<=2$ ， $F(N)=0$ 。否则令 $m=\lfloor\frac{N-1}{2}\rfloor,S=\min\{\frac{m}{2^k}s.t.\ 2^k|m,k\in\mathbb{Z}\}$ 。若 $S=1$ ， $F(N)=0$ ；否则 $F(N)=2S$ 。

知道了这一点后，我们考虑题目怎么做。既然是除以最大的2次幂，这就可以用01trie维护。在点分治之

后，以起点的 v_i 加上到分治中心的距离建01trie。对于到分治中心距离为 l 的点，即将01trie里的值进行 l 次+1，询问答案即可。

对于01trie的+1操作：前提是该01trie中，是从低位向高位插入的。对一个数+1，其二进制的变化，形如 $111110\cdots \to 000001\cdots$ （低位在前），就相当于在01trie上交换01节点。

对于查询：我们先将读入的 v_i 减去1，在其0分支与1分支将答案求和。01trie上的一个点，其代表了前 i 位一样的所有数，我们要维护这些数除以 2^i 下取整的和。求答案的时候要注意除去2的幂次，因为这些的答案是0。

时间复杂度： $O(n\, log^2\, n)$ 。

3 C. Pty loves lines

做法1

题意即为求所有可能的 $\frac{n(n-1)}{2} - \sum_i \frac{a_i(a_i-1)}{2}$ ，要求满足 $\sum a_i = n$ 。

左侧是个常数，现考虑右侧的所有可能取值。

由于 $a_i = 1$ 时贡献是0，因此若 $\sum a_i = x$ 的方案能凑出一个取值s，那么和比x更大的也可以。

只需设 f_n 表示使得右侧为n的最小 $\sum a_i$ ，类似背包dp的转移，时间复杂度 $O(n^3)$.

做法2

设 $f[i][j]$ 表示用 i 条直线是否能凑出 j 个交点。

可以用bitset来优化转移，做到 $O(n^4/w)$ 的时间复杂度，这是难以通过 1 s的时间限制的。

观察发现答案有相当长一段连续可行后缀，打表后发现最大不可行交点数不超过 35000 ，于是时间可以优化到 $35000 \times n^2/w$ ，可以通过本题。

4 D. Pty hates prime numbers

做法1：

考虑经典容斥：

枚举每个数选和不选，设选的数的乘积是 S ，那么 $ans+ = (n/S) * (-1)^{(the\ number\ of\ selected\ number)}$

时间复杂度： $O(2^k)$ 。

做法2：

假设 $k = 8$ ，发现前 8 个质数的乘积： $2*3*5*7*11*13*17*19 = 9699690$ 非常小。

注意到[x不是前8个质数的倍数]=[x mod 9699690不是前8个质数的倍数]。

n 的答案 $=$ (9699690 的答案) $\times (\lfloor \frac{n}{9699690} \rfloor) + (n\, mod\, 9699690$ 的答案)

预处理 $n = 0 \sim 9699690$ 的答案即可 $O(1)$ 回答。

做法3：

结合以上两个做法：

1. $k <= 8$ 时随便做。
2. $k > 8$ 时，先容斥第 9 个质数到第 k 个质数，到前 8 个质数时直接利用预处理好的答案 $O(1)$ 回答。

时间复杂度： $O(\sum_{i=1}^{8}\frac{9699690}{p[i]}+T*2^8)$ ，可以轻松通过本题。

5 E. Pty loves book

将所有的串一起丢入 ac 自动机中，考虑计算 ac 自动机的每个点 x 代表的串 s 的 $\sum_{i=1}^{|s|}f(s,i,|s|)^5$ 。

设 $tot[x][k]$ 为 x 代表的串所有后缀的价值 k 次方和， $fail_{tot}[x][k]$ 为 x 代表的串所有左端点在 x 的 $fail$ 之前的后缀的 k 次方和。

$tot[x]$ 从 x 的 $fail$ 的 tot 和 $fail_{tot}[x]$ 转移过来。

考虑如何计算 $fail_{tot}[x]$ 的值，可以发现这个就是 x 找 $fail$ 的时候从 x 的父亲开始跳经过的所有点的 $fail_{tot}$ (包括 x 的父亲)加上因为多加的一个字符所多出的价值也就是 x 的所有 $fail$ 的价值和。(注意要特殊计算 x 整个串的价值)。

对于 5 次方多维护 0 到 4 次方的和就可以进行快速合并。

时间复杂度 $O((|S|+\sum|T|)\times 5^2)$ 。

6 F. Pty loves lcm

当 $y-x+1>2$ 时， $f(x,y)$ 约为 $\frac{x^{y-x+1}}{(y-x)!}$,所以 $f(x,y)<10^{18}$ 的个数大约有 10^6 个，可以暴力预处理。

对于 $y-x=1$ ， $gcd(x,y)=gcd(x,x+1)=1$ ，所以就是求 $\sum_{i=1}^{n}\phi(i)\phi(i+1)$ 这样形式的东西。

由于 $n\le 10^9$ ，所以可以分段打表。

分段打表是每 10^6 个打一个表，那么表大小大约 $10kb$ 。

分段打表与正解都要使用区间筛算法，如果筛 $[l,r]$ 的区间先处理出 $\sqrt{r}$ 以内的质数，每个质数枚举它在 $[l,r]$ 之间的倍数筛掉并乘上对应的 φ 值，最后再乘上筛剩的质数的 φ 值。

时间复杂度为 $O((r-l)\log\log r)$ 。

7 G. Pty loves graph

相当于哈密顿环上有若干条弦，你要安排他们在圆周内或圆周外，并且两个真相交（即除了包含与相离之外的其余情况）的弦不能同时在内或外。

从哈密顿环上的任何一个地方断开，真相交关系仍然会保留。将每条弦看做一个点，对两个有限制的点间连边。

相当于判定新图是否能够黑白染色。我们只关心图的连通染色性，因此按照左端点从大到小，右端点从小到大枚举两个相交区间中右边的那个，再维护一个左端点小于当前区间左端点的右端点集合，发现我们需要连边的是右端点的一段区间。最终图中这一段区间是连通并且同色的，后面再次连向这个区间时，只需要连一条边即可。并且右端点只会删除，不会插入，所以可以用并查集和set来优化连边的过程。每一次将连边的区间内的所有右端点合并。最后再判断图是否是二分图即可。

时间复杂度 $O(n\log n)$ ，空间复杂度 $O(n)$ 。

时空常数较优秀的线段树做法也能通过。

8 H. Pty loves string

考虑一个出现位置 $S[l..r]$ ，会满足 $S[l..l+x-1]$ 和 $S[1..x]$ 相等， $S[r-y+1..r]$ 和 $S[n-y+1..n]$ 相等，且 $l+x=r-y+1$ 。

前两个条件容易想到border，用KMP即可求出对于所有的 $l+x-1$ 合法的 x 。

考虑KMP做完从 i 向 i 的border连一条边，这样我们会得到一棵树，一个 x 的所有出现位置即为其子树。

那么问题可以变为：有两棵树，问两个子树内有多少个点编号相同？

这个问题可以转化为二维数点，用数据结构（扫描线+树状数组）维护即可。

复杂度 $O(n\log n)$ 。

9 I. Pty loves SegmentTree

设 f_n 表示根节点区间为 $[1,n]$ 的答案，首先有 $f_1=1$ ，然后讨论根节点为A类点还是B类点，有

$$f_n=(A-B)f_{n-k}f_k+B\sum_{i=1}^{n-1}f_if_{n-i}$$

我们先算出 f_k ，然后令新的 $A=(A-B)f_k$

考虑 $\{f_n\}$ 的生成函数 $F(x)$ ，容易得到 $F(x)=Ax^kF(x)+BF^2(x)+x$ ，解方程得到 $F(x)=\frac{1-Ax^k\pm\sqrt{A^2x^{2k}-2Ax^k-4Bx+}}{2B}$

由于 $F(x)$ 常数项为0，易知选择负号

我们尝试规避平方根的影响，令 $Q^2(x)=A^2x^{2k}-2Ax^k-4Bx+1$

首先我们有

$$2BF(x)=1-Ax^k-Q(x) \tag{1}$$

两边求导得到

$$2BF'(x)=-kAx^{k-1}-Q'(x)$$

$$\Longrightarrow Q'(x)=-kAx^{k-1}-2BF'(x) \tag{2}$$

回到开始，我们有 $Q^2(x)=A^2x^{2k}-2Ax^k-4Bx+1$

两边求导得到

$$2Q'(x)Q(x)=2kA^2x^{2k-1}-2kAx^{k-1}-4B$$

$$Q'(x)Q(x)=kA^2x^{2k-1}-kAx^{k-1}-2B$$

注意到我们在(2)中求得了 $Q'(x)$ 的取值，代入得到

$$(-kAx^{k-1}-2BF'(x))Q(x)=kA^2x^{2k-1}-kAx^{k-1}-2B$$

两边乘 $Q(x)$ 得到

$$(Akx^{k-1}+2BF'(x))Q^2(x)=Q(x)(2B+kAx^{k-1}-kA^2x^{2k-1})$$

在(1)中我们知道了 $Q(x)$ 的取值， $Q^2(x)$ 为定义，代入得到

$$(Akx^{k-1}+2BF'(x))(A^2x^{2k}-2Ax^k-4Bx+1)=(1-Ax^k-2BF(x))(2B+kAx^{k-1}-kA^2x^{2k-1})$$

展开

$$2BF'(x)(A^2x^{2k}-2Ax^k-4Bx+1)+Akx^{k-1}+A^3kx^{3k-1}-2A^2kx^{2k-1}-4AkBx^k$$

$$=-2BF(x)(2B+kAx^{k-1}-kA^2x^{2k-1})+2B+kAx^{k-1}-kA^2x^{2k-1}-2ABx^k-kA^2x^{2k-1}+kA^3x^{3k-1}$$

化简

$$F'(x)(A^2x^{2k}-2Ax^k-4Bx+1)-2Akx^k=-F(x)(2B+kAx^{k-1}-kA^2x^{2k-1})-Ax^k+1$$

两边取 x^n 的系数，得到

$$(n+1)f_{n+1}+A^2(n-2k+1)f_{n-2k+1}-2A(n-k+1)f_{n-k+1}-4Bnf_n$$

$$=-2Bf_n-kAf_{n-k+1}+kA^2f_{n-2k+1}+[n=k](A(2k-1))$$

整理得到

$$(n+1)f_{n+1}=2B(2n-1)f_n+A(2n-3k+2)f_{n-k+1}-A^2(n-3k+1)f_{n-2k+1}+[n=k](A(2k-1))$$

至此我们得到了 f_{n+1} 关于 f_n , f_{n-k+1} , f_{n-2k+1} 的递推式，直接递推即可
复杂度 $O(n)$ 。

10 J. Pty plays game

如果双方只有一个人，只需要比较两个人打死对面的时间即可。
$t=min(h_1/d_2,h_2/d_1)$
士兵2赢当且仅当 $h_1/d_2<h_2/d_1$ ，移项得 $h_1d_1<h_2d_2$
假设两人打架时间为 t ，可以得到 $t=min(h_1/d_2,h_2/d_1)$
假设是 h_1/d_2 ，那么 t 时间后两人的血量分别是 $h_1-(h_1/d_2)*d_2,h_2-(h_1/d_2)*d_1$
现在士兵2要跟下一个人打，观察他的攻击力乘以生命值: $(h_2-(h_1/d_2)*d_1)*d_2=h_2d_2-h_1d_1$

如果认为一个士兵的攻击力乘以生命值就是他的“战斗力”，两个士兵打完以后，战斗力高的人的战斗力会减少战斗力低的人的战斗力。

最后剩下的一定是战斗力高的一队。

由于攻击力和生命值都写成了关于 x 的一次函数，所以一队的战斗力就是一个二次函数。

问题变成了询问最小的使得二次函数为正的非负整数 x 。

这题应该会卡精度，所以要写个高精度什么的。

时间复杂度： $O(n)$ 。