A. Lovelive! School Idol Project

题意

给你$n(1 \le n \le 2 \times 10^5)$个区间$[l,r](1 \le l \le r \le 10^9)$, 接下来$m(1 \le m \le 2 \times 10^5)$个询问. 每个询问给出你一个区间$[L,R](1 \le L \le R \le 10^9)$, 问$[L,R]$有多少个子区间包含至少一个区间$[l,r]$. 答案对$10^9+9$取模.

题解

预处理每个区间$[l,r]$. 如果区间$[l_i,r_i] \subset [l_j,r_j](i \ne j)$, 那么包含$[l_j,r_j]$的区间一定包含$[l_i,r_i]$. 因此删去$[l_j,r_j]$而只保留$[l_i,r_i]$即可. 这部分的工作可以对所有区间$[l,r]$按照$l,r$分别为第一, 第二关键字排序后, 利用单调栈来完成. 具体来说, 按照排序后从小到大顺序枚举每个区间, 如果当前栈顶区间右端点大于等于当前考虑区间, 则栈顶元素一定包含当前区间, 于是弹栈即可. 这样一直继续下去, 直到栈为空或者栈顶区间右端点小于当前考虑区间. 然后将当前考虑区间压入栈顶. 于是栈中元素关于l和r均是严格递增. 因此栈中所有区间就是最后保留下来的所有区间.

记这些保留下来的所有区间为
$A_1=[l_1,r_1],A_2=[l_2,r_2],\ldots,A_s=[l_s,r_s]$(按从小到大排序). 对于某个询问$[L,R]$, 记$f(S)$为$[L,R]$中包含集合$S$的子区间数. 根据容斥原理, $[L,R]$包含至少一个区间A_i的子区间个数为:

$$\sum_{1\le i\le s} f(A_i)-\sum_{1\le i<j\le s} f(A_i\cup A_j)+\sum_{1\le i<j<k\le s} f(A_i\cup A_j\cup A_k)$$
$$+\ldots+(-1)^{s+1}\sum_{1\le i_1<i_2<\ldots<i_s\le s} f(A_{i_1}\cup A_{i_2}\cup\ldots\cup A_{i_s})$$

考虑$A_{i_1}\cup A_{i_2}\cup\ldots\cup A_{i_t}$(下标从小到大排列). 注意到, 包含这些所有区间等价于包含区间$[l_{i_1},r_{i_t}]$. 且该集合对上式计算贡献为$(-1)^{t+1}$.

固定$u=i_1,v=i_t$. 当$v-u>1$时, 枚举区间$[l_u,r_v]$除了$[l_u,r_u]$和$[l_v,r_v]$外还包含的集合个数(记为w), 则该区间对上式贡献为:

$$\sum_{w=0}^{v-u-1}(-1)^{w+2+1}C_{v-u-1}^{w}=-\sum_{w=0}^{v-u-1}(-1)^{w}C_{v-u-1}^{w}=-(1-1)^{v-u-1}=0$$

因此只需要考虑$v-u=1$和$v-u=0$时的贡献即可.
因此$[L,R]$包含至少一个子区间$[l,r]$的子区间个数为:

$$\sum_{1\le i\le s} f([l_i,r_i])-\sum_{1\le i<s} f([l_i,r_{i+1}])$$

而我们知道:

$$f([l,r]) = \begin{cases} 0 & [l,r] \not\subset [L,R] \\ (l-L+1)(R-r+1) & [l,r] \subset [L,R] \end{cases}$$

而l,r均是严格增的. 因此可以求出$l_i, r_i, l_i \cdot r_i, l_i \cdot r_{i+1}$各自的前缀和, 然后通过对$L$和$R$分别在$l$和$r$中二分求出其包含的区间后计算即可. 复杂度$O((n+m)logn)$.

预计难度: Mid

B. Color the Edges

题意

给你一个$n(1 \le n \le 1000)$个点的完全图. 你需要把每条边染一种颜色, 满足有公共端点的两条边颜色不同. 给出一种使用颜色数最少的染色方案.

题解

当$n=1$时, 不需要染色. 故只需要0种颜色.

当$n>1$且n为奇数时, 设有k种颜色, 则每种颜色至多使用$\frac{n-1}{2}$次. 所以$k \cdot \frac{n-1}{2} \le \frac{n(n-1)}{2} \Rightarrow k \le n$.

另一方面, 让边(i,j)染颜色$((i+j) \bmod n)+1$, 则这种构造满足要求. 故最少需要n种颜色.

当n为偶数时, 设有k种颜色, 则每种颜色至多使用$\frac{n}{2}$次. 所以$k \cdot \frac{n}{2} \le \frac{n(n-1)}{2} \Rightarrow k \le n-1$.

另一方面, 考虑前$n-1$个点构成的完全图, 则$n-1$为奇数. 按照奇数的情况来对前$n-1$条边染$n-1$种颜色. 而对于第n个点, 让边(i,n)染颜色$((i+i) \bmod (n-1))+1$, 则这种构造满足要求. 故最少需要$n-1$种颜色.

预计难度: Easy

D. Three Leaves

题意

给你$n(1 \le n \le 10^6)$个编号为$1,2,\ldots,n$的点. 输出从这n个点中取k个点组成一棵叶子节点恰好为3的树的方法数. 输出$k=1,2,\ldots,n$的情况. 答案对998244353取模.

题解

可以证明得到的树一定是一个根节点连出3条是链的子树的情况. $k=1,2,3$时答案为0. $k \ge 4$时, 答案为:

$$C_n^k \cdot k \cdot \frac{1}{3!} \sum_{\substack{s_1+s_2+s_3=k-1 \\ s_1,s_2,s_3 \ge 1}} C_{k-1}^{s_1} C_{k-1-s_1}^{s_2} C_{k-1-s_1-s_2}^{s_3} (s_1)!(s_2)!(s_3)!$$

其中C_n^k表示从n个点中选出k个点的方法数, k表示选根节点的方法数, s_1, s_2, s_3为三个子树的大小. 除3!是因为后面表达式计算中3个子树的选择顺序不加区分. $C_{k-1}^{s_1}$为从剩下的$k-1$个点中选出s_1个点的方法数, $C_{k-1-s_1}^{s_2}$为从剩下$k-1-s_1$个点中选出s_2个点的方法数. $C_{k-1-s_1-s_2}^{s_3}$为从剩下$k-1-s_1-s_2$个点中选出s_3个点的方法数. $(s_1)!, (s_2)!, (s_3)!$分别为这三条子树链链按照从根到叶子形成排列的方法数. 根据乘法原理, 将它们相乘就是k时的答案.

将上式化简可得: 原式=

$$\frac{n!\cdot k}{6(n-k)!k!}\sum_{\substack{s_1+s_2+s_3=k-1\\ s_1,s_2,s_3\geq 1}}(k-1)! = \frac{n!}{6(n-k)!}\sum_{\substack{s_1+s_2+s_3=k-1\\ s_1,s_2,s_3\geq 1}}1$$

后面的求和表达式可以看作将$k-1$个球排成一排, 通过插两个板将它分为大小为s_1, s_2, s_3个连续段的方法数, 也相当于在$k-2$个空隙中插2个板的方法数. 故其值为C_{k-2}^2. 因此答案为:

$$\frac{n!}{6(n-k)!}C_{k-2}^2 = \frac{n!(k-2)(k-3)}{12(n-k)!}$$

分母部分可以利用线性逆元累乘求出, 或者算出$n!$的逆元后累乘求出. 时间复杂度为$O(n)$.

预计难度: Easy

I. Math Problem

题意

给你两个正整数$n, m(1 \le n \le 10^{18}, 1 < m \le 10^7)$. 设$n! = am^k$, 其中$a$是一个不被$m^k$整除的正整数. 求$a \bmod m^k$的值.

题解

将m作素因子分解: 设$m = p_1^{t_1} p_2^{t_2} \dots p_s^{t_s}$. 对于每个素数$p_i$, 我们把$n!$表示为$n! = \alpha_i p_i^{\beta_i}$, 其中$\alpha_i$是一个不被$p_i$整除的正整数. 则题目中所求的$k = \min\{[\frac{\beta_1}{t_1}], \dots, [\frac{\beta_s}{t_s}]\}$. 根据勒让德定理, 我们有:

$$\beta_i = \sum_{j=0}^{\infty} [\frac{n}{p_i^j}]$$

由此可求出k. 于是如果我们能求出每个α_i, 那么

$$a = \alpha_i (p_1^{kt_1} \dots p_{i-1}^{kt_{i-1}} p_{i+1}^{kt_{i+1}} \dots p_s^{kt_s})^{-1} p_i^{\beta_i - kt_i} \pmod{p_i^{t_i}}$$

(此处$(p_1^{kt_1} \dots p_{i-1}^{kt_{i-1}} p_{i+1}^{kt_{i+1}} \dots p_s^{kt_s})^{-1}$为$p_1^{kt_1} \dots p_{i-1}^{kt_{i-1}} p_{i+1}^{kt_{i+1}} \dots p_s^{kt_s}$在模$p_i^{t_i}$下的逆元, 可以用扩展欧几里得算法或欧拉定理求出). 于是利用中国剩余定理即可求出a.

最后考虑怎么求α_i. 记tab_j为$[1, j]$中所有与p_i互素的数的乘积(特别地, $tab_0 = 1$). 我们暴力求出$tab_0, \dots, tab_{p_i^{t_i} - 1} (mod\ p_i^{t_i} - 1)$. 记$Tab = tab_{p_i^{t_i}} (mod\ p_i^{t_i})$. 那么令$n$对$p_i$一直作带余除法,

即:

$$
\begin{gathered}
n = q_0 = q_1 p_i + r_1 \\
q_1 = q_2 p_i + r_2 \\
\dots \\
q_{w-1} = q_w p_i + r_{w-1} \\
q_w = r_w
\end{gathered}
$$

则$\alpha_i = Tab^{\sum_{j=0}^{w}[\frac{q_j}{p_i^{t_i}}]} tab_{q_0 \bmod p_i^{t_i}} \dots tab_{q_w \bmod p_w^{tw}} \cdot$

这是因为, 设$[1, j]$中所有与p_i互素的数为$v_1, v_2, \dots, v_{h(j)}$, 则:

$$
\begin{gathered}
n! = v_1 \cdot v_2 \cdot \dots \cdot v_{h(q_0)} \\
\cdot p_i v_1 \cdot p_i v_2 \cdot \dots \cdot p_i v_{h(q_1)} \\
\cdot p_i^2 v_1 \cdot p_i^2 v_2 \cdot \dots \cdot p_i^2 v_{h(q_2)} \\
\cdot \dots \\
\cdot p_i^w v_1 \cdot p_i^w v_2 \cdot \dots \cdot p_i^w v_{h(q_w)} \\
= tab_{q_0} \dots tab_{q_w} p_i^{h(q_1)+2h(q_2)+\dots+wh(q_w)} \\
= tab_{q_0} \dots tab_{q_w} p_i^{\beta_i}
\end{gathered}
$$

结合$tab_{q_j} = Tab^{[\frac{q_j}{p_i^{t_i}}]} tab_{q_j \bmod p_i^{t_i}}$即可得到.

总时间复杂度: $O(m + logmlogn)$.