送分题

前3个操作昨天课件里讲过了做法就不赘述了，考虑怎么在有区间最值的时候维护区间历史最小值和。

考虑构造一个序列$c_i = a_i - b_i$，由于区间取max可以转化成对区间最小值的加减操作，因此我们对于区间上a_i最小值所在的位置集合与其他位置构成的集合需要分开考虑，分别对这两个集合维护c_i的最小值、最小值个数、次小值即可。

复杂度可以证明是$O(n\log^3 n)$，但是效率是$O(n\log^2 n)$级别的，目前还不知道怎么证明。

水题

考虑$F(n)$的通项公式$\frac{\varphi^n-\phi^n}{\sqrt{5}}$，其中$\varphi = \frac{1+\sqrt{5}}{2}, \phi = \frac{1-\sqrt{5}}{2}$，我们只要维护$\varphi^n$和$\phi^n$的值即可求出$F(n)$的值。

（要构建一种类(a,b)表示$a+b\sqrt{5}$，搞出加减乘除操作之后就可以直接用了。）

所以第3个询问就很简单了，考虑第4个询问。

我们要求的式子可以改写成：

$$\sum_{i=1}^{k}\sum_{j=i}^{k}\frac{\varphi^{\sum_{p=i}^{j}b_p} - \phi^{\sum_{p=i}^{j}b_p}}{\sqrt{5}}$$

很显然我们只需要知道怎么求$\sum_{i=1}^{k}\sum_{j=i}^{k}\varphi^{\sum_{p=i}^{j}b_p}$就可以了。

考虑将每个点的值a_i表示成φ^{a_i}，那么其实就是要求区间积的和。

这个和是可以合并的，考虑两个区间分别为$P = u_1, u_2, \dots, u_x$和$Q = v_1, v_2, \dots, v_y$，考虑合并成$R = u_1, u_2, \dots, u_x, v_1, v_2, \dots, v_y$，记$R_{ans}$表示区间$R$的区间积的和，那么有：

$$R_{ans} = P_{ans} + Q_{ans} + \sum_{i=1}^{x}\prod_{d=i}^{x}u_d \cdot \sum_{j=1}^{y}\prod_{d=1}^{j}v_d$$

很显然我们只需要维护区间的前缀积的和与后缀积的和即可完成转移，分别记为R_{pre}, R_{suf}，那么要维护R_{pre}和R_{suf}，我们需要维护整个区间的积R_{prd}。

现在考虑修改操作，第一个操作改变了树的形态，我们用LCT维护信息即可，第二个操作是链覆盖，设链的大小为sz，覆盖的值为x，那么

$$R_{prd} = x^{sz}, R_{pre} = R_{suf} = \frac{x^{sz+1} - x}{x-1}$$

$$R_{ans}=\sum_{i=1}^{sz}\sum_{j=i}^{sz}x^{j-i+1}=\sum_{l=1}^{sz}x^l\cdot(sz-l+1)=sz\cdot\frac{x^{sz+1}-x}{x-1}-\frac{sz\cdot x^{sz+1}-\frac{x^{sz+1}-x}{x-1}}{x-1}$$
$$=\frac{\frac{x^{sz+1}-x}{x-1}-x\cdot sz}{x-1}=\frac{x^{sz+1}-x-x(x-1)\cdot sz}{(x-1)^2}$$

因为要快速幂，算上LCT，总复杂度是$O(n\log^2 n)$。

简单题

将$n!$表示成$p_1^{a_1}p_2^{a_2}\dots p_m^{a_m}$，那么问题可以转化成求出$k$个不同的$m$维向量$x_1\sim x_k$，使得每一维都是非负整数，并且$\sum_{i=1}^k x_i=\left(a_1,a_2,\dots,a_g\right)$，最后把方案数乘上$k!$的逆元即可。

因为要求的是不同的向量，我们可以考虑用容斥将这个限制给去掉。

因为有$\binom{k}{2}$条不能相等的限制，暴力的话我们可以以$2^{\binom{k}{2}}$枚举限制是否被打破，假设打破的限制集合为e，那么计算的就是至少打破了限制集合e的方案数，容斥系数为$(-1)^e$。

考虑将向量看成点，限制看成边，那么每一个集合e相当于将图分成了若干个联通块，然而答案只和集合大小有关，与集合顺序无关，也就是代表一个整数拆分，因此我们可以枚举k的整数拆分，对于一个整数拆分计算贡献。

考虑一个整数拆分$p_1,p_2,\dots,p_m$，有$\sum_{i=1}^m p_i=k$，那么我们要计算的有两个部分，一个是计算将点填进这个整数拆分的方案数，另一个是这个整数拆分的容斥系数。

设大小为i的块有a_i个，有$\sum_{i=1}^k a_i\cdot i=k$，那么将$k$个点填进整数拆分$p$中的方案数就是

$$\frac{k!}{\prod_{i=1}^k (i!)^{a_i}\cdot a_i!}$$

现在考虑容斥系数怎么求，令S_n表示n个点的联通图集合，$e(G)$表示G的边数，那么容斥系数就是：

$$\sum_{G_1\in S_{p_1}}\sum_{G_2\in S_{p_2}}\dots\sum_{G_m\in S_{p_m}}(-1)^{\prod_{i=1}^m e(G_i)}=\prod_{i=1}^m\sum_{G_i\in S_{p_i}}(-1)^{e(G_i)}$$

记$g_i=\sum_{G_i\in S_{p_i}}(-1)^{e(G_i)}$，所以系数就是$\prod_{i=1}^m g_{p_i}$。

考虑g_n怎么求，我们可以用“全集”减去“不连通的”来得到“联通的”。

假设有x条边，那么当$x=0$时，全集为$(-1)^0=1$；当$x>0$时，全集为$\sum_{i=0}^x(-1)^i\binom{x}{i}$，这个是等于0的。

不连通的，考虑1所在的联通块大小为s，那么贡献是$\binom{n-1}{s-1}g_s$，剩下的$n-s$个点可以任意连边，也就是$n-s$个点的全集，只有当$n-s=1$时，全集才为1，因此：

$$g_n=\begin{cases}1,n=0\\0-\binom{n-1}{n-2}g_{n-1}=-(n-1)g_{n-1},n>0\end{cases}$$

那么点分配的方案数和系数都考虑完了后，考虑向量和，每一维都是独立的，因此我们只需要考虑这样一个问题：

$$\sum_{i=1}^{m}p_ix_i=a$$

求这个方程的非负整数解的个数，考虑完全背包，复杂度$O(ma)$，然后把每一维的答案乘起来即可。

至此这个问题就解决了，具体复杂度因为有个整数拆分也不是很好分析，极限数据本机跑了$2s$左右。