Solution(Day2)

方阵

a) 对于所有的求和操作，我们均可以用前缀和直接算出，以下我们只考虑求最大最小值操作。

b) **40%**的数据

对于每个询问，直接暴力求值，时间复杂度 **O**（**n*m*q**）。

c) **60%**的数据

对于每个询问，在其中一维使用 **RMQ** 优化，另一维枚举，时间复杂度 **O**（**n*q**）。

d) **80%**的数据

维护 **2** 维 **RMQ**，查询的时候直接查询四块的最大（最小）值即可，时间复杂度 **O(n*m*log(n)*log(m)+q)**。

e) **100%**的数据

方法 **d** 并不能过，因为空间开不下。

定义 **f[i][j][k]**表示以（**i**，**j**）为左上角，大小为 **2^k** 的正方形方阵的最大（最小）值是多少。

维护类似 **RMQ**。

可以发现，对于每个询问，最多 **8** 个正方形即可覆盖。

所以时间复杂度 **O**（**n*m*log(n)+q**）

合影

a) 对于 **15%**的数据

我们可以直接枚举所有置换，再进行判断即可，时间复杂度 **O(T*n!)**

b) 对于 **50%**的数据

定义 **f[S]**表示已经选了元素集合为 **S** 的合法方案数是多少，转移方程为 **f[S]**=$\sum f[S-2^x]$，满足 **x** 是 **S** 中的元素且所有限制均不矛盾，可以先预处理每个位置不合法状态的集合，这样时间复杂度 **O**（**n*T***2^n）**,**如果你写成 **O**（**n*n*T***2^n），那么恭喜你，你只有 **30**（常数小的除外）分了。

c) 对于余下所有数据，我们需要观察出一些性质。

如果我们对于一条限制（**x**，**y**），建一条 **x->y** 的边，就会发现在这张图中每个点的出边最多只有一条。

如果在这张图中有环，则答案一定为 **0**。

如果这张图无环，则这些边一定构成一个森林。

所以我们可以跑树形 **dp**。

定义 **f[i]**表示第 **i** 棵子树的方案数是多少。

转移时，相当于有很多的儿子s_1，s_2**…**s_n（设每个儿子的子树大小为 **size**），相当于对于一个序列中，先选 1 个位置给s_1，

再在余下的位置中，选 2 个位置给s_2**…**，相当于一些组合数的乘积再乘以每个儿子内部的方案数。

d) 对于 **70%**的数据

我们可以利用组合恒等式C_x^y=C_{x-1}^y+C_{x-1}^{y-1}预处理组合数，或是预处理逆元暴力求组合数，复杂度均为 **O**（**T*n*n**）**[m** 一定小于等于 **n**，所以 **m** 近似为 **n]**

e) 对于 **100%**的数据

我们可以先预处理阶乘，阶乘的逆元，然后组合数就可以 **O**（**1**）了，总复杂度为 **O**（**T*(n+m)**）

D2T3 筹备计划

n,q<=200

记录每个位置上学生个数。暴力枚举所有可行位置，计算距离和。复杂度 **O(q*n*n)**

n,q<=2000

考虑每个位置上维护与上一个位置的距离和之差 **d**。我们发现每个学生对这个值的贡献为：在这个学生和他前面的所有 **d** 减 **1**，所有后面的 **d** 加 **1**。这样可以对所有操作 **O(n)**处理，同时计算答案时，只需维护 **d** 的前缀和，找前缀和最小的可行位置即可。

不存在 **type=3** 和 **type=4** 的操作

我们发现 **d** 值满足单调性**(**因为每个位置的学生数量都大于等于 **0**，所以每个位置对 **d** 的贡献满足单调性，加起来也满足**)**，所以可以用个线段树维护 **d** 值，查到最后一个小于等于 **0** 的位置便是距离和最小，同时位置最靠左的点。

n,q<=200000

我们发现答案只可能是在距离和最小的点靠左最近的可行点，和靠右最近的可行点距离和较小的那一个。这个东西也可以在线段树上维护一下进行查询，所以复杂度为 **O(qlogn)**。