信心赛(Simple) Solution

Problem A 简单棋盘问题

Tag: 切比雪夫距离和曼哈顿距离的转化，二分

对于 1 号测试点，显然无需移动，输出 0 即可；

对于 2 号测试点，记录每个国王所在位置后 BFS 即可得到答案；

对于 3 号测试点，在#2 的基础上加上记忆化搜索即可.

对于 4,5 号测试点，

注意到国王从(x,y)移动到(a,b)所需步数为 Max(|x-a|,|y-b|).

充分性：可以斜向移动 Min(|x-a|,|y-b|)次，然后直接走直线即可到达目标，此时花费次数即为 Max(|x-a|,|y-b|)步.

必要性：每次移动只能使横纵坐标变化至多 1，所以至少要移动 Max(|x-a|,|y-b|)次

于是对于每个询问暴力计算出这个值累加求和即可，

如果初始化采用 memset，没有注意到答案为 long long 类型，则只能够通过 4 号测试点.

对于 6,7 号测试点，

注意到 Max(|x-a|,|y-b|)实际上求的是两点之间的切比雪夫距离.

有 Max,Min 不好求解，于是可以通过将坐标系旋转 45 度转化为曼哈顿距离.

问题转化为给定 N 个点(Kxi,Kyi)，Q 组询问(Txi,Tyi)到这 N 个点的曼哈顿距离和.

注意到曼哈顿距离中 x,y 坐标对答案的贡献相互独立，所以可以分开计算贡献.

问题转化为给定 N 个数 Kxi，Q 次询问 Txi 到这 N 个点的距离之和，

注意到对于 Kxi<Txi,对答案的贡献为 Txi-Kxi，

对于 Kxi>=Txi,对答案的贡献为 Kxi-Txi.

于是将 Kxi 排序后预处理出前缀和，

查询时直接在有序数组上二分找出分界点，

利用前缀和 O(1)计算出答案.

如果初始化采用 memset，没有注意到答案为 long long 类型，则只能够通过 6 号测试点.

时间复杂度 O((n+q)logn).

Problem B 简单数据结构

Tag：线段树，堆

小调查：有多少选手现场学习并实现了 segment tree beats 呢？

#1：n,m<=3000，直接模拟，考察选手对 std::sort 的使用

#2: RMQ 问题，可以通过 ST 表或者线段树实现，注意-1 的判断

#3: 区间把比一个数小的数字变成这个数，查询区间最小值.

因为查询内容与区间和值等无关，所以无需 Segment Tree Beats 中的方法实现，

直接在每个线段树节点上维护当前区间 min 值以及区间与哪个数取 max 的 lazy 标记即可

#4：考虑操作 2 怎么解决，因为 K 的和值不超过 5*10^6，考虑与其相关的做法.

对序列建一棵线段树，线段树上每个节点维护当前区间最小值的值和位置，同时用一个小根堆去维护最后的答案，线段树上每次 query(l,r)返回一组{val,pos,l,r}表示[l,r]区间中最小值为 val，位置为 pos，按照 val 去建立这个小根堆.

开始把 query(1,n)的结果入堆，然后重复 K 次以下操作：

每次弹出堆顶，显然这时的 val 是所剩下的数中的最小值，

然后将 query(l,pos-1)以及 query(pos+1,r)的结果入堆,

这样得到的 K 个堆顶显然是最小的 K 个元素，判断-1 后输出即可,

而关于时间复杂度，query 与入堆的次数都是 2K 次，弹出的次数为 K 次,

所以总复杂度为 O(n+sigma(K)*logn).

#5,#6: 发现在#4 与#3 的维护区间 min 值并不会产生影响，于是将#3,#4 一起实现,

即可通过全部数据，时间复杂度 O((n+sigma(K))*logn)，空间复杂度 O(n+sigma(K)).

对于 1<=n,m<=100000 的另解:

如果你想不到堆+线段树维护的方法，可以利用下发课件中开篇提到的平衡树套支持区间取 max 的线段树做到 O((n+m)log^3n)，也可以用分块做到 O(n*sqrt(n)*logn),

均可通过满足 1<=n,m<=100000 的数据得到高分.

Problem C 简单树上路径

Tag：bitset，树上倍增，Hall 定理

#1：答案只可能是 0 或 2，直接判断即可

#2：考虑如何快速得到一个人可以买的特产集合，显然最后到达的点是这 c 个点的 LCA 位置，对于每个点可以倍增预处理出其到 2 的次幂级祖先的特产集合，其到 LCA 的路径上的特产集合也可以通过倍增求出，利用 bitset 优化这个倍增的过程，时间复杂度 $O((n\log n+qcm)/w)$.由于 m,q 很小，所以可以暴力枚举集合判断解决#2.

#3,#4,#5：

同样利用 bitset 得到一个人可以买的特产集合，由于此时 m 比较大，$O(nm\log n/w)$ 的空间超过了 256M 的空间限制，所以可以将询问离线，从大到小枚举 2 的次幂，多组询问同时进行倍增，从而可以将大小为 logn 的一维实现滚动，从而降低空间复杂度为 $O(nm/w)$

对于 c=2 的解法，得到两个集合后，答案只与两个集合的大小,交的大小和并的大小有关，对这几个值的大小分类讨论即可解决(也可以二分答案以避免讨论).

#6,#7：

当 c 达到 3,4,5 时，分类讨论很难解决这样的问题.

考虑二分答案 x，如何判断 x 是否可行，

建立二分图 G，左边有 c*x 个点，每个人有 x 个点，右边有 m 个点表示每种特产，

左边和右边的点有连边，当且仅当左边点对应的人可以购买右边点对应的特产，

而 x 可行等价于这样的二分图 G 存在一个完美匹配.

注意到二分图 G 存在完美匹配可以通过 Hall 定理进行判断，

于是设 dp(S)表示 S 集合中的人可以买的特产数量，

容易发现答案就是对 2^c-1 种 S 取 min(dp(S)/size(S)),也无需二分答案.

这样做单次询问的时间复杂度是 O((2^c+clogn)m/w)，

总复杂度是 O((nmlogn+qm(2^c+clogn))/w)，

运算量大约在 1.5e8 级别，可以通过.