

POJ2352 【基础】

题目大意：

按 y 递增的顺序给出 n 颗 ($1 \leq n \leq 15000$) 星星的坐标 (y 相等则 x 递增)。规定每个星星的等级等于在它左边且在它下边 (包括水平和垂直方向) 的星星的数量, 求出等级为 0 到 $n-1$ 的星星分别有多少个。

输入：

第一行有一个整数 n , 接下来有 N 行, 每一行有两个整数 $x \ y$ ($0 \leq x, y \leq 32000$), 表示一颗星星的 $x \ y$ 坐标。

输出：

N 行, 第 1~ N 行依次为等级 0~ $N-1$ 的星星的数量。

题解：

这题满足树状数组的两个要求：1、给 $A[i]$ 加上一个数 (给坐标 i 加上一颗星星)；2、求 $\text{sigma}(A[k], k=1..i)$ (求坐标 i 左边的星星数量)，因此是一题裸的树状数组题目。

POJ1195 【基础】

题目大意：

对于一个最大不超过 1024×1024 的矩阵, 有四种操作：

0 s : 清空矩阵, 并将矩阵大小设置为 $s \times s$

1 $x \ y \ a$: 矩阵 $m[x][y]$ 的值加上 a

2 $x1 \ y1 \ x2 \ y2$: 求子矩阵 $m[i][j]$ 的和, 其中 $x1 \leq i \leq x2, y1 \leq j \leq y2$

3: 结束操作

输入：

有若干行操作, 每一行的格式如上述, 其中 $x \ y \ x1 \ y1 \ x2 \ y2$ 均为整数, 并且范围在 $0 \sim 1024$ 中。

输出：

每一次操作 2 的时候，输出一行，包含一个整数，即子矩阵元素的和。

题解：

二维的树状数组，和一维的很像，多一层扩展而已，参见

<http://162.105.203.28/course/ada09/students/00748261-CSW-TreeArray.pdf>。

$tree[i][j]$ 记录从 $[0][0]$ 到 $[i][j]$ 区域的元素之和，所以

$sum = sum(x2+1, y2+1) - sum(x2+1, y1) - sum(x1, y2+1) + sum(x1, y1)$ （一个大的矩形减去两个小的矩形再加上重复相减的部分）

P0J2299 【基础】

题目大意：

给出 N ($1 \leq N \leq 500000$) 个数，求最少进行多少次交换两个数才能将这些数按升序排列。

输入：

有若干组测试数据。每一组测试数据第一行有一个整数 N ，接下来有 N 行，每一行有一个整数，不大于 999999999，表示第 i 个数。

输出：

每一组测试数据输出一行，包含一个整数，即最少的操作次数。

题解：

容易知道交换次数等于这个数列的逆序数，也就是每一位元素的逆序数之和。

可以用树状数组来求。因为我们要求的是“编号比 x 小，数值比 x 大”的数有多少个，所以先对数值进行降序排列后，等价于求 x 前面有多少个元素的编号比 x 的编号要小。那么我们只需要按顺序，先做一次求和（前面有多少元素比当前元素的编号小），然后再把当前元素加入进去，就可以了。

P0J3067 【基础】

题目大意：

11 区要修新的高铁网络。已知关东有 n 座城市，关西有 m 座城市

($1 \leq n, m \leq 1000$)，一共有 k 条 ($1 \leq k \leq 1000 * 1000$) 线路要修。并且关东关西的城市都已经编号，如果有线路 $e1$ $e2$ ， $e1$ 的在关西的城市编号小于 $e2$ ，但是

e_1 在关东的大于 e_2 ，那么就说 e_1 e_2 是交叉的。给出 k 条线路的信息，求有多少个交叉点。

输入：

第一行有一个整数 t ，表示有 t 组测试数据。每一组测试数据第一行有三个整数 n, m, k ，下面有 k 行，每一行有两个整数，即一条线路在关西和关东的城市编号。

输出：

每一组测试数据输出一行，格式为 Test case %测试数据组号%: %交叉点数%，其中测试数据组号是从 1 开始编号的。

题解：

设一条线路 e 在关西编号为 u 在关东编号为 v 。仿照 POJ2299，我们要求的是“ u 比 $e[i].u$ 小， v 比 $e[i].v$ 大”的数的个数的和，因此按 u 降序- v 降序进行排序，然后按顺序扫下去，因为当前的 $e[i].u$ 一定比前面的 u 小，所以只要前面有多少 v 比当前的 $e[i].v$ 大就可以了。而求这个就是按树状数组的求一次和加一个进去来操作即可。

POJ1990 【中等】

题目大意：

有 N 只 ($1 \leq N \leq 20000$) 牛站在 X 轴上，每一头牛都站在一个整数点上，可以发出一个声调 $w[i]$ ($1 \leq w[i] \leq 20000$)。两头牛之间要沟通的话，需要消耗 $\max(w[i], w[j]) * \text{dis}(i, j)$ 的能量， $\text{dis}(i, j)$ 表示两头牛的 x 坐标之差。求所有牛之间进行沟通消耗的能量之和。

输入：

第一行有一个整数 N 。接下来有 N 行，每一行有两个整数，表示一只牛的 x 坐标和声调。

输出：

一行，包含一个整数，即能量之和。

题解：

首先按 x 坐标进行排序，方便后续按顺序进行处理。维护两个数组 $wNum$ 和 $wDis$ ， $wNum[x]$ 表示一侧比声调 x 小的声调的牛的数量， $wDis[x]$ 表示一侧比声调 x 小的声调的牛到端点的距离之和。那么对于牛 i ，可以算出其一侧所有比它声调小的牛到它的距离之和，乘以它的声调就得到它对一侧的沟通能量之和。对左侧和右侧都算一次，就得到了总的能量之和了。

P0J2481 【中等】

题目大意：

有 n 头牛 ($1 \leq n \leq 100000$)，每一头牛都有两个测验值 S, E 。如果对于牛 i 和牛 j 来说，它们的测验值满足下面的条件则证明牛 i 比牛 j 强壮： $S_i \leq S_j$ 且 $E_j \leq E_i$ 且 $E_i - S_i > E_j - S_j$ 。现在已知每一头牛的两个测验值，要求输出每头牛有几头牛比其强壮。

输入：

有若干组测试数据。每一组测试数据第一行有一个整数 n ， $n=0$ 时测试数据结束。接下来有 n 行，每一行有两个整数，表示第 i 头牛的 $s\ e$ 值。

输出：

每一组测试数据输出 n 行，依次是有多少头牛比第 i 头牛强壮。

题解：

首先按照 e 降序， e 相同时 s 升序的方式来进行排序，这样使得排在后面的牛一定不会比排在前面的牛更强。然后通过树状数组，依次求每一头牛前面有多少头牛比它强，然后加入这头牛。注意需要记录一下两头牛的 $s\ e$ 值相同的数量，然后需要把这个值减掉。

P0J2155 【中等偏上】

题目大意：

对于一个 $n*n$ ($2 \leq n \leq 1000$) 的矩阵，规定两种操作：

C $x_1\ y_1\ x_2\ y_2$: 对 $m[i][j]$ $x_1 \leq i \leq x_2, y_1 \leq j \leq y_2$ 中的元素取异或 (0 变 $1, 1$ 变 0)

Q $x\ y$: 查询 $m[x][y]$ 元素的值

输入：

第一行有一个整数 t ，表示有 t 组测试数据。

每一组测试数据第一行有两个整数 n 和 m ， m 表示有多少个操作。接下来有 m 行，每一行有一个操作，格式如上述。其中 x x_1 x_2 y y_1 y_2 全部为整数。一开始矩阵内所有的元素都是 0。

输出：

每一组条 Q 的操作输出一行，包含一个数字 0 或者 1，表示查询的矩阵元素的值。

题解：

这题和树状数组的“区域查值，单点更新”刚好反了过来，变成了“区域更新，单点查值”。但是由于更新的方式很特殊，是异或更新，这就意味着我们只需要记录更新的次数就可以知道点的值。因此，我们把原来的更新操作和查询操作反过来，将原来的区域查询的值作为当前的点异或次数，那么更新的时候只需要给四个角落的 $[x_1][y_1]$ 、 $[x_2][y_2]$ 加上 1， $[x_1][y_2]$ 、 $[x_2][y_1]$ 减去 1。因为这样一来对更新区域内的点的 sum 会+1（因为加上了 $[x_1][y_1]$ 的值），区域外的也都不影响（要么+0，要么+1 -1，要么+1 +1 -1 -1）。这样就可以。

P0J3321 【中等偏上】

题目大意：

一棵具有 n 个 ($1 \leq n \leq 100000$) 节点的树，一开始每个节点上都有一个苹果。

有 m 个 ($1 \leq m \leq 100000$) 操作：

$C\ i$ 是摘掉第 i 个节点上面的苹果（若苹果不存在，则为加上一个苹果）

$Q\ i$ 是查询以第 i 个节点为根的子树有几个苹果（包括第 i 个节点）

输入：

第一行有一个整数 N ，接下来有 $N-1$ 行，每一行有两个整数 $u\ v$ ，表示 u 是 v 的父节点。然后有一行包含一个整数 m 。下面有 m 行，每一行有一个操作，格式同上。

输出：

对于每一个 $Q\ i$ 操作，输出一行，包含一个整数，即第 i 个节点为根的子树有几个苹果。

题解：

这题关键在于将一颗树映射到一个数组中，并且每一棵子树都对应一个连续的区间。我查了一下，使用深搜的方法，记录进入一个节点最早的时间，作为这个节点的编号；和这个节点的子树都搜索完毕以后的时间，作为这个节点管辖

区间的下限。设一个节点管辖区间的上限是 $\text{low}[u]$ ，管辖下限是 $\text{high}[u]$ ，并且可知这个节点的编号就是 $\text{low}[u]$ 。这样就完成了一个符合要求的映射。改变的操作按树状数组的正常操作即可。求和的操作就是求 $\text{sum}(\text{high}[u]) - \text{sum}(\text{low}[u] - 1)$ 。