



# 线段树例题

唐文斌

# Brackets<sub>(SPOJ61,BRCKTS)</sub>

- 维护一个长度为  $N$  的小括号序列  $A$
- 实现两种操作:
  - *Replace*(  $i$  ) : 将第  $i$  个位置的括号反向
  - *Check* : 测试当前序列是否合法
- 规模:
  - $N \leq 30000$

# Brackets<sub>(分析)</sub>

- 不合法的情况:
  - 右括号先于左括号出现, “()())”
  - 左右括号数不等, “()((”
- 维护信息:
  - 用(5, 3)表示 “)())\*)\*\*\*(((”
  - 用(a,b)表示当前这段序列左边多a个右括号, 右边多b个左括号
  - (0,0)为合法情况
- 线段树

# 动态最值 (SCOI2006)

- 维护数组  $A$
- 实现两个操作:
  - $Delete(i)$  : 删除第  $i$  个数, 右边的数左移
  - $RMQ(i, j)$  : 求  $A[i..j]$  的最小值
- 规模:
  - $A$  的大小  $N \leq 10^6$
  - 操作次数  $M \leq 10^6$



## 动态最值(分析)

- 删除相当于把数设为无穷大
- $RMQ$ 时先计算  $i$  和  $j$  在原数组中的位置, 然后再计算一般的  $RMQ$
- 用线段树维护
  - 记录最小值
  - 增加附加信息: “剩余元素个数”

# Housewife Wind<sub>(PKU2763)</sub>

- 给定一棵边带权的树  $T$
- 实现两种操作:
  - $Change(x, t)$  : 将树  $T$  中的边  $x$  权值改为  $t$
  - $Query(a, b)$  : 查询点  $a$  到点  $b$  的路径权和
- 规模:
  - 节点数目  $N \leq 10^5$
  - 操作次数  $Q \leq 10^5$

# Housewife Wind<sub>(分析)</sub>

- 转为有根树
  - 距离拆分
- 预处理
  - 在线LCA查询
- 设 $d[u]$ 表示 $u$ 到 $root$ 路径的权和
  - 则 $change(x, t)$ 影响的是以某个点 $v$ 为根的子树(边 $x$ 为 $v' \rightarrow v$ ,  $v'$ 是父亲)
- 子树在树的欧拉序列(或者先序遍历)中是连续的
  - 树  $\rightarrow$  序列
  - 线段树维护

# K-th Number<sub>(POJ2104)</sub>

- 给定一个数组  $A[1..M]$
- 每次询问为:
  - $Query(i, j, k)$
  - 查询  $A[i..j]$  中第  $k$  大的数是多少
- 规模  $N \leq 10^5$ , 询问次数  $M \leq 5000$



# K-th Number<sub>(分析)</sub>

## ○ 预处理

- 建立线段树, 每个线段保存该区间内元素排序好的序列

## ○ 查询 $Query(i, j, k)$

- 把 $[i, j]$ 进行区间分解
- 二分 $W$ , 每次统计这些区间内一共有多少个数比 $W$ 大, 用 $\log N$ 次统计可求出第 $k$ 大元素

## ○ 如何统计原子区间内比 $W$ 大的元素总个数?

# K-th Number<sub>(分析)</sub>

- 统计原子区间内一共有多少个数比 $W$ 大
  - 区间内的数已排序, 用二分每个区间求比 $W$ 大的数 $\log N$
  - 累加所有 $2\log N$ 个区间比 $W$ 大的数, 共 $\log^2 N$
  - 总时间:  $\log^3 N$
- 实现: 一个归并排序可以同时构造线段树和每个节点内的排序数组.
  - 空间:  $O(N\log N)$
  - 查询复杂度: 不超过 $O(\log^3 N)$

# Dynamic Ranking<sub>(ZJU2112)</sub>

- 给定一个数组  $A[1..M]$
- 实现两个操作:
  - $\text{Modify}(i, t)$  将  $A[i]$  的值改为  $t$
  - $\text{Query}(i, j, k)$  查询  $A[i..j]$  中第  $k$  大的数是多少

# Dynamic Ranking<sub>(分析)</sub>

- *Modify*操作:

- 不可再使用有序表
- 每个节点内部存储一棵平衡树

- 时间复杂度:

- *Modify* : 修改  $O(\log N)$  棵平衡树,  $O(\log^2 N)$
- *Query* :  $O(\log^3 N)$

# The Ultimate Bamboo Eater

- 平面上有  $N$  个点, 每个点  $i$  有两个值  $W_i$  与  $L_i$
- 要求选择一条最长的路径  $P_1 \rightarrow P_2 \rightarrow \dots \rightarrow P_k$
- 满足:
  - $W_{P_i} < W_{P_{i+1}}$
  - Manhattan Distance  $(P_i, P_{i+1}) \leq L_{i+1}$
- 规模:
  - $N \leq 10^5$
  - $X_i, Y_i, W_i, L_i \leq 10^6$



# The Ultimate Bamboo Eater

- 动态规划

- $O(N^2)$

- 旋转坐标轴 → 二维线段树



# Cockroach Race<sub>(Ural1369)</sub>

- 给定平面上的 $N$ 个点
- 有 $M$ 次查询:
  - 查询距离一个点 $P$ 最近的点(如果有多个, 求所有)
- 规模:
  - $N \leq 10^5$
  - $M \leq 10^4$

# ● ● ● | Cockroach Race<sub>(分析)</sub>

- 标准算法:
  - Voronoi ,  $O(N\log N + M\log N)$
- 四分树 – Quad-Tree





# 附录

- <<The Ultimate Bamboo Eater>>
  - <http://acmicpc-live-archive.uva.es/nuevoportal/data/problem.php?p=3294>