笛卡尔树初步

SGColin

目录

1 笛卡尔树

1.1 概念与性质

笛卡尔树是一棵基于序列生成的**二叉树**。

我们称序列中每个节点的下标为编号，权值为键值。那么序列生成的笛卡尔树满足：

1. 键值满足小（大）根堆的性质（父节点键值小于 (大于) 任意子节点）

2. 中序遍历满足节点编号递增，即左子树中的节点在序列中都在父节点之前，右子树中的节点都在父节点之后。

1.2 构造

以序列极值的位置作为根，然后左（右）子节点分别是以该点分开的左（右）区间的极值。然后递归的进行左（右）区间的建树。真正实现是类比虚树的思想，只需要用一个单调栈**维护最右链**，即可实现线性复杂度建出笛卡尔树。

我们以小根堆的要求来描述建树过程。具体的，我们从左往右扫描序列，依次加入笛卡尔树中。若栈顶的键值比当前位置大，说明栈顶的子树已经访问完毕，此时栈顶应该在当前位置的左子树中，因此设当前位置左子节点为栈顶，将栈顶退栈。一直找到栈为空或栈顶键值小于当前位置停止，若栈不为空则栈顶就是当前节点的父亲。

```
for (int i = 1; i <= n; ++i) {
  while (top && a[stk[top]] > a[i]) {
    son[i][0] = stk[top]; --top;
  }
  if (top) son[stk[top]][1] = i;
  stk[++top] = i;
}
```

2 题目小结

2.1 [POJ 2559] Largest Rectangle in a Histogram

2.1.1 Description

POJ 2559

给出若干个并排的矩形条，宽度都为 1，第 i 个条高度为 $a[i]$，求这个图形包含的最大的矩形。

2.1.2 Solution

Code

建出来小根的笛卡尔树，统计出每个节点对应的区间长度 $w[i]$，答案就是 $\max\{a[i] \times w[i]\}$。

2.2 [ICPC 2016 Hong Kong] G. Scaffolding

2.2.1 Description

Kattis - scaffolding

要搭一个 n 列的脚手架，第 i 列需要摞 h_i 根短竹。每次至多搬 m 根竹子上脚手架，只能在脚手架左右和向上移动，并且保证脚下得有竹子。问最少需要多少次能把整个脚手架搭建完。

2.2.2 Solution

Code

建出来小根的笛卡尔树，考虑把在子节点搭建过程中剩余的竹子放到当前位置。注意这里我们的每个节点代表的是一个宽为子树总宽度，长为 $h_i - h_{fa}$ 的矩形。

设 $f[u]$ 表示节点 u 的子树全部搭建完成最少需要几次，$g[u]$ 表示搬完子树能剩下多少竹子。

那么有 $f[u] = f[ls] + f[rs] + \lceil \frac{rem}{m} \rceil$ ，其中 $rem = \max(0, (h_u - h_{fa}) * wid_u - (g[ls] + g[rs]))$

则 $g[u] = f[u] * m - sum[u]$，其中 $sum[u] = \sum_{v \in u} h_v - wid_u * h_{fa}$，即子树所有矩形的大小。

答案即 $f[root]$。

2.3 [CCPC-Wannafly Winter Camp Day1 (Div1)] E. 我爱割葱

2.3.1 Description

Comet OJ 41

葱一共有 $n(n \leq 100)$ 棵，第 i 棵葱的高度为 a_i。

一共要割最多 k 刀葱，每刀可以在某一高度割去连续一段葱。以高度 h 在区间 $[l, r]$ 割一刀葱是合法的，当且仅当区间里的葱的高度都不小于 h，此时，这个区间中的葱小于等于 h 的未被割的部分都会被割掉。下面的葱被割掉以后，上面的葱不会掉下来。求 k 刀割掉的葱的总长度的最大值。

2.3.2 Solution

Code

问题即为选 k 个不相交的矩形使得面积和最大。

建出来小根的笛卡尔树，注意到这是一个树形背包的问题。设 $f[u][j][k]$ 表示 u 子树内割了 j 刀，占下表面的 k 个格子时的最大收益。设 $g[u][i][j]$ 表示 u 子树内不考虑当前矩形，割了 j 刀，占下表面的 k 个格子时的最大收益，那么显然 $g[u] = f[ls] * f[rs]$ ，即左右子树的卷积。

考虑当前节点的贡献，有

$f[u][i][wid_u] = \max\{f[u][i][wid_u], g[u][i][wid_u], g[u][i-1][x] + height_u * (wid_u - x)\}$

原本总复杂度是 $O(n^5)$ 的，但是如果两维都只枚举到子树大小，根据树形背包的结论复杂度是 $O(n^3)$ 的。另外 f 数组卷积可能会有一些位置状态本身不合法，所以需要初始化成 $-\infty$。

2.4 [SPOJ PERIODNI] Periodni

2.4.1 Description

SPOJ PERIODNI　BZOJ 2616

给定一个 n 列的表格，每列的高度各不相同，但底部对齐，然后向表格中填入 k 个相同的数，填写时要求不能有两个数在同一列，或中间连续的同一行。即两个数填在同一行不同列是合法的，当且仅当它们中间的表格断开。求所有填写方案对 10^9+7 的余数。

2.4.2 Solution

Code

考虑建出来小根笛卡尔树，还是考虑每个矩形内部的答案。设 $f[u][i]$ 表示 u 子树内放 i 个数的方案数，设 $g[u][i]$ 表示 u 子树内不考虑当前矩形，放 i 个数的方案数，显然有 $g[u]=f[ls]*f[rs]$，即左右子树的卷积。

考虑当前位置的贡献，枚举当前矩形内有多少列还是空的，有

$$f[u][i]=\sum_{j=0}^{i} g[u][i-j]\times\binom{wid_u-(i-j)}{j}\times\binom{height_u}{j}\times j!$$

最后乘上 $j!$ 是因为横纵坐标是两两组合的，因此匹配的方案数为 $j!$。

复杂度瓶颈在每个节点都进行了两次卷积，为 $\mathrm{O}(n^2)$。