第十二次直播课

习题讲解

李嘉政

Dec 2023

Table of Contents

Table of Contents

Solution

$f_{i,j,k}$ 表示前 i 个物品，组出体积为 j，重量为 k 的最大价值和。转移就是基础的背包，$f_{i,j,k} = \max(f_{i-1,j,k}, f_{i-1,j-v_i,k-m_i} + w_i)$，时间复杂度 $\mathcal{O}(nVM)$。

Table of Contents

Solution

$f_{x,0/1}$ 表示在 x 的子树中，x 这个点是否选取的最大权值和。如果这个点没选，那么儿子的状态就是任意的，于是 $f_{x,0} = \sum_{y \in son_x} \max(f_{y,0}, f_{y,1})$；如果这个点选了，那它的儿子们都不能选，所以 $f_{x,1} = \sum_{y \in son_x} f_{y,0}$。时间复杂度 $\mathcal{O}(n)$。

Table of Contents

Solution

注意到答案就是每个节点作为根时，最大的深度乘 k 再减去对应的换根代价。这意味着我们要在换根的时候维护出最大的深度。设 x 节点最大深度为 mx_x，次大深度为 smx_x。为什么需要次大深度呢？因为在从 x 换根到 $y \in son_x$ 时，我们需要知道 x 的最大深度是否来自于 y，否则直接用 x 的最大深度更新即可。具体来说，如果 $mx_x = mx_y + 1$，则换根后 $mx_y = \max(mx_y, smx_x + 1)$，否则 $mx_y = \max(mx_y, mx_x + 1)$。而实际上，$smx_x$ 也是类似容易维护的。时间复杂度 $\mathcal{O}(n)$。

Table of Contents

Solution

等价于求出子树大小排序。时间复杂度 $\mathcal{O}(n)$。

Table of Contents

Solution

等价于我们需要求出每个节点为根是，离它最远和次远的距离。这是容易 dfs 得到的，换根使用最远和次远也容易计算出下一个，类似上上题。时间复杂度 $\mathcal{O}(n)$。

Table of Contents

Solution

注意到严格次大一定是最大减一。实际上等价于求出直径，做法比较多，像上一题一样换根也行。时间复杂度 $\mathcal{O}(n)$。

Table of Contents

Solution

和蓝桥舞会没有区别。

Table of Contents

Solution

和上一题也没啥区别。

Table of Contents

Solution

和蓝桥舞会毫无区别。

Table of Contents

Solution

石子合并有若干种做法。最基础的就是 $\mathcal{O}(n^3)$ 的区间 dp，设 $f_{l,r}$ 表示将区间 $[l,r]$ 合并成一堆的最小代价。转移就是枚举断点，即将 $[l,k]$ 和 $[k+1,r]$ 两个区间合并，转移式子为 $f_{l,r}=\min_k f_{l,k}+f_{k+1,r}+\sum_{i=l}^{r}a_i$，后面的 $\sum$ 显然可以用前缀和优化，时间复杂度 $\mathcal{O}(n^3)$。
实际上可以注意到 $\sum$ 满足四边形不等式 (蒙日矩阵)，可以基于单调性给出 $\mathcal{O}(n^2)$ 的做法，甚至有 $\mathcal{O}(n\log n)$ 的做法。

Table of Contents

Solution

标准的二维数点。
把给出的区间 $[l, r]$ 看成一个二维平面上的点 (l, r)，询问区间 $[l, r]$ 看作二维平面上的一个矩阵 $[l, r] \times [l, r]$，那么每次询问等价于询问矩形内点数。这是什么？二维前缀和！时间复杂度 $\mathcal{O}(n^2 + q)$。
当然实际上存在 $\mathcal{O}(n \log n + q \log q)$ 的做法，这里不再赘述。