

树形与数位动态规划

2019年1月26日

黄哲威 hzwer

北京大学16级计算机科学



第三节 目标

- 一些树形dp补充
- 数位dp介绍
- 数位dp练习

NOIP2014 联合权值

- » 无向连通图 G 有 n 个点， $n-1$ 条边。点从 1 到 n 依次编号，编号为 i 的点的权值为 W_i ，每条边的长度均为 1。
- » 图上两点 (u, v) 的距离定义为 u 点到 v 点的最短距离。
- » 对于图 G 上的点对 (u, v) ，若它们的距离为 2，则它们之间会产生 $W_u \times W_v$ 的联合权值。
- » 请问图 G 上所有可产生联合权值的有序点对中，联合权值最大的是多少？所有联合权值之和是多少？
- » 其中 $0 \leq n \leq 10^5$

NOIP2014 联合权值

暴力做法：枚举树上两点，如果它们的深度相同或者相差 2，然后判断它们的距离是否为 2，再考虑贡献答案

判断过程可以bfs，lca等等

NOIP2014 联合权值

树形 dp：一个结点距离相差 2 的点要不然是儿子的儿子，不然是兄弟

先考虑第一部分

只要记录一个结点儿子的权值和 $sum[x]$ ，以及权值的最大值 $mx[x]$

$$ans1_x = \max\{mx_{y_i} * w_x\}$$

$$ans2_x = \sum sum_{y_i} * w_x$$

NOIP2014 联合权值

树形 dp：一个结点距离相差 2 的点要不然是儿子的儿子，不然是兄弟

考虑第二部分

如果枚举一个结点的儿子两两计算的话，会被箭形图卡成 n^2 只要再用前缀和/最大值维护一下即可

Codefest 16.The Chocolate Spree

- 在 n 个点的点权树上选两条不相交的路径，使路径上的点权和最大
- $n \leq 100000$

BZOJ1369.Gem

- 给出一棵树，要求你为树上的结点标上权值，权值可以是任意的正整数，唯一的限制条件是相邻的两个结点不能标上相同的权值，要求一种方案，使得整棵树的总价值最小。
- $N \leq 10000$

BZOJ1369.Gem

- 给出一棵树，要求你为树上的结点标上权值，权值可以是任意的正整数，唯一的限制条件是相邻的两个结点不能标上相同的权值，要求一种方案，使得整棵树的总价值最小。
- $N \leq 10000$
- 结论是 $\log n$ 种权值就能染色了
- $F[i, j]$ 表示以 i 为根的子树，根的颜色是 j 的最小权值

小奇的仓库

喵星系有 n 个星球，星球以及星球间的航线形成一棵树。

从星球 a 到星球 b 要花费 $[\text{dis}(a,b) \text{ Xor } M]$ 秒。（ $\text{dis}(a,b)$ 表示 ab 间的航线长度，Xor为位运算中的异或）

为了给仓库选址，小奇想知道，星球 i （ $1 \leq i \leq n$ ）到其它所有星球花费的时间之和。

$n \leq 10^5$, $M \leq 15$

有部分数据 $M=0$ ，部分数据 $M=1$

小奇的仓库

对于 $M=1$ 的情况，树形DP到一个点时记录有多少个0，多少个1，然后每当一条路径到2，那部分就再记录一个值。

小奇的仓库

对于 $M=1$ 的情况，树形DP到一个点时记录有多少个0，多少个1，然后每当一条路径到2，那部分就再记录一个值。

$M \leq 15$ ，只要把“0”、“1”、大于等于2变成0~16

Codefest 16.The Chocolate Spree

$\text{best}[x]$ 表示 x 子树内的最长链

$\text{down}[x]$ 表示子树内从 x 出发的最长链

$\text{up}[x]$ 表示从 x 出发，终点在 x 子树外的最长链

Codefest 16.The Chocolate Spree

$\text{best}[x]$ 表示 x 子树内的最长链

$\text{down}[x]$ 表示子树内从 x 出发的最长链

$\text{up}[x]$ 表示从 x 出发，终点在 x 子树外的最长链

最终一定存在点，使得最优解的一条链 a 经过它，另一条链 b 在它的
一个儿子的子树内且链 a 不经过这个儿子

枚举这个儿子 y ，答案就是 $\text{best}[y]$ 加上链 a

链 a 有若干种情况，记录 down 的前后缀最大/次大值讨论一下即可

BZOJ1040. 骑士

- 有N个骑士，每个人有一个战斗力，每个人有一个他讨厌的骑士，求一个骑士军团，使得军团战斗力的和最大且不包含有矛盾的两人。
- $1 \leq N \leq 10^6$

BZOJ1040. 骑士

- 有N个骑士，每个人有一个战斗力，每个人有一个他讨厌的骑士，求一个骑士军团，使得军团战斗力的和最大且不包含有矛盾的两人。
- $1 \leq N \leq 10^6$
- 整个图实际上是一个无向环套树森林。
- 对于每一棵环套树，先dfs找环，找到环以后断环为链并将断开的两个点强制其中一个点为根且不选做一次树形DP，对另一个点做同样操作。

BZOJ2427. 软件安装

- 现在我们的手头有 N 个软件，对于一个软件 i ，它要占用 W_i 的磁盘空间，它的价值为 V_i 。我们希望能从中选择一些软件安装到一台磁盘容量为 M 计算机上，使得这些软件的价值尽可能大（即 V_i 的和最大）。
- 每个软件可能依赖零个或者一个软件
- $0 \leq N \leq 100, 0 \leq M \leq 500$

BZOJ2427. 软件安装

显然图是一些环和树

若有环必须选整个环，或者直接舍弃

将环缩点完之后变成一堆点和森林，建立虚点向所有无入度的点连边

BZOJ2427. 软件安装

显然图是一些环和树

若有环必须选整个环，或者直接舍弃

将环缩点完之后变成一堆点和森林，建立虚点向所有无入度的点连边

缩点用闭包传递或者tarjan，剩下的就是个树形dp问题

$f(i,j)$ 表示对 i 及其子树花费 j 代价产生的最大价值

数位DP

- » 数位DP就是处理和具体的数有关的DP。
- » 大多数的题目形式类似于：统计[L,R]之间有多少整数满足xx性质。这个xx性质往往是数位之间的性质。L、R一般很大，如 10^{100000} ，算法复杂度往往和数的位数相关。

数位DP

- » 数位DP就是处理和具体的数有关的DP。
- » 大多数的题目形式类似于：统计 $[L,R]$ 之间有多少整数满足xx性质。这个xx性质往往是数位之间的性质。 L 、 R 一般很大，如 10^{100000} ，算法复杂度往往和数的位数相关。
- » 不过NOIp范围内，数据范围可能并不会真的给到 10^{100000} 。出题人通常只想确保你写的不是完全的暴力，因此可能数据范围只给到 10^9 。这就给其他一些骗分算法诸如分段打表留下了发挥的空间。
- » 数位DP主要麻烦的地方在于前导零的处理、 $\leq R$ 之类限制的处理。撇开这些通常就是一个普通的序列DP。

数字计数

- » 给定两个正整数 a 和 b ，求在 $[a,b]$ 中的所有整数中，每个数字(digit, i.e. 0~9)各出现了多少次。
- » $a,b \leq 10^{12}$

数字计数

- » 常用转化：统计 $[0, a-1]$ 和 $[0, b]$ 的答案，相减就是 $[a, b]$ 的答案。
- » 许多写过数位dp的同学，对于 $\leq b$ 这个限制的处理往往都是 $dp[i][0/1][0/1]$ 的时候记一下确定了前 i 位，当前有没有分出大小，前缀是否都是零之类的限制。然后枚举下一位是什么进行转移。
- » 如果考虑不清楚这个状态的设计转移以及各种边界情况，可以换一个方法：注意到任何一个 $\leq b$ 的数 x ，一定和 b 有一个 k 位公共前缀。这 k 位他们相同，第 $k+1$ 位 $x < b$ 。
- » 枚举公共前缀的长度 k ，枚举第 $k+1$ 位上 x 的值，之后转化为没有任何限制的情况。
- » 对于 $k=0$ ，变为枚举最高位的位置及数值，同样转化为没有限制的情况。

数字计数

- » 于是只需要统计：
- » $f[i][0\sim 9]$ 表示所有*i*位数中0~9出现了几次。（其实就= $i * 10^{(i-1)}$ ）
- » $g[i]$ 表示*i*位数有多少个。（= 10^i ）
- » 复杂度 $O(w)$

数字计数

- » 联赛难度的几乎所有数位DP都可以通过枚举公共前缀以及下一位的具体数值直接转换为没有任何限制的序列DP问题。
- » 这使得NOIp中看似没有单独讨论数位DP的必要。
- » 但事实上这种方法还是有或多或少的细节，所以大家还是把数位DP看成一类特殊的DP。初学者一般比较难以一次写对，但好好对拍就没事了。

WINDY数

- » windy定义了一种windy数。不含前导零且相邻两个数字之差至少为2的正整数被称为windy数。 windy想知道， 在A和B之间， 包括A和B， 总共有多少个windy数？ 将答案对1000000007取模。
- » $A, B \leq 2 \times 10^9$

WINDY数

- » 按照上题的方法转化为没有限制的形式，于是只需要统计长度为 n 的序列，有多少个满足相邻两个数差至少为2（这个序列的第一个元素确定）。
- » 这用形如 $dp[i][c]$ 的dp来记确定了前 i 位，最后一位的数值是 c ，总共的方案数。来进行处理非常简单。
- » 具体实现请同学们自行摸索。数位DP大多数说起来简单写起来烦，还是建议大家多写点题找找感觉。

WINDY数

- » 如果有同学就是痛恨数位dp拒绝写标算，这类数据范围不大的题也可以分段打表。
- » 假设数据范围很小，就可以全部打表，离线预处理好 $F[i]$ 表示 $[0,i]$ 之间有多少合法的windy数，把这些数全部当常量打到代码里。读入数据 $O(1)$ 解决询问。
- » 但是这题数据范围有 10^9 ，这么多数据数存不下的。此时我们对数据范围进行分块。设 $S=O(\sqrt{B})$ ，这里取60000左右。然后预处理出所有 $F[k*S]$, $k=1,2,\dots$
- » 这样读入询问后，找到离 B 最近的一个 $k*S$ ，从 $F[k*S]$ 开始暴力计算 $F[B]$ 。例如 $B=60111$ ，那么因为已经算出了 $F[60000]$ ，暴力for $i=60001 \dots 60111$ 判断每个 i 是否是windy数即可。
- » 这样至多只需要暴力 $O(S)$ 步，算法的复杂度是 $O(\sqrt{B})$ 的。

计数

- » 给你一堆非零的数字可重集 S （一个数字可能出现多次）。你可以随意排列，然后在里面插入任意多个0，这样就能得到无穷多个数。
- » 比如说给定 $\{1,2\}$ ，那么可以生成数字12,21,102,120,201,210,1002,1020,等等。
- » 现给定一个数 n ，问 S 生成的数中有多少个 $\leq n$ 。
答案对 10^9+7 取模。
- » $n \leq 10^{50}$

计数

- » 设 n 是 m 位的。往 S 中补0直到 $|S|=m$ 。
- » 问题就转化为将 S 中的数重新排列，有多少小于等于 n 。
- » 枚举公共前缀搞一发之后直接变成，给定一个可重集 S' ，问 S' 中的元素组成的不同的排列个数。
- » 设数 i 有 a_i 个，那么答案就是 $(\sum a_i)! / (\prod a_i!)$

花神的数论题

- » 用 $\text{sum}(i)$ 表示数 i 二进制表示下1的个数。
- » 问 $\text{sum}(1) \sim \text{sum}(n)$ 这 n 个数的乘积对
 1000000007 取模的结果。
- » $n \leq 10^{15}$

花神的数论题

- » 乘积看起来有点假？
- » sum 可能的取值只有几十种。
- » 计算每个 sum 对应的数有多少个。
- » $\text{dp}[i][k]$ 表示长度为 i 的序列总共有 k 个1，这样的序列个数。
- » 变成水题。

泊神的数论题

- » 求 $[l, r]$ 中，各位数字之和能够整除原数的数的个数。
- » $l, r \leq 10^{18}$

泊神的数论题

- » 换汤不换药。
- » 在dp序列的过程中不知道数位和K是多少，不好计算模？
- » 在外面暴力枚举K，计算有多少序列同时满足和等于K，以及能被K整除即可。
- » $dp[i][j][c]$ 表示长度为i的序列，总和是j，现在组成的数模K等于c，的总方案数。

幸运数

- » 如果一个数各个数位上的数字之和是质数，并且各个数位上的数字的平方和也是质数，则称它为幸运数。
- » 例如：120是幸运数，因为120的数字之和为3，平方和为5，均为质数，所以120是一个幸运数字。
- » 给定 x, y ，求 x, y 之间(包含 x, y 即闭区间 $[x, y]$)有多少个幸运数。
- » $x, y \leq 10^{16}$

幸运数

- » 这种都是一样的，变成序列问题，就直接暴力记 $dp[i][j][k]$ 表示 i 位，和是 j ，平方和是 k 的方案数。
- » 到时候再把 j 和 k 都是质数的方案拿出来统计一下好了。

淘金

- » 数轴 $1 \dots N$ 上每个整点有1单位的金子。
- » 一阵风吹过，位置 i 的金子被吹到了 $f(i)$ ，其中 $f(i)$ 是 i 各个数位的乘积。如果 $f(i)$ 跑到了 $1 \dots N$ 之外，则认为这块金子已经不存在。
- » 问：风过后，拥有最多金子的前 K 个地点，其上的金子个数之和。
- » $N \leq 10^{12}, K \leq 10^5$

淘金

- » 乘积可能的值挺大的，有 9^{12} 。看起来不能像前两题一样枚举了？
- » 其实还是可以枚举的，因为乘积能够得到的不同的数个数很少。
- » 具体有多少？
- » 乘积由1~9构成，也即只由质因子2,3,5,7构成。而2,3,5,7的幂次最多只有36,24,12,12。
- » 枚举公共前缀变成序列问题，即 $dp[i][a][b][c][d]$ 表示前i位，乘积是 $2^a 3^b 5^c 7^d$ 的方案数。
- » 枚举所有可行方案，nth_element或者sort取前K大即可。

Q & A