

## NOIP DP专题

P2339 [USACO04OPEN] Turning in Homework G

Sol

P8867 [NOIP2022] 建造军营

Sol

CF1153F Serval and Bonus Problem

Sol

[ABC221G] Jumping sequence

Sol

格子跳跃

Sol

P4042 [AHOI2014/JSOI2014] 骑士游戏

Sol

P1912 [NOI2009] 诗人小G

Sol

P3262 [JLOI2015] 战争调度

Sol

CF868F Yet Another Minimization Problem

Sol

P5336 [THUSC2016] 成绩单

Sol

P4749 [CERC2017] Kitchen Knobs

Sol

CF908G

Sol

P8392 [BalticOI 2022 Day1] Uplifting Excursion

Sol

P4007 小 Y 和恐怖的奴隶主

Sol

CF1481F AB Tree

Sol

CF1188D Make Equal

Sol

# NOIP DP专题

## noip dp知识点一览：

背包，树上dp，记忆化搜索，环形dp，数位dp，区间dp，序列dp，dp的优化（数据结构，单调队列，斜率优化，凸性优化，四边形不等式优化，wqs三分），状压dp（and轮廓线dp），以及博弈论中sg函数类似于dp的计算

写在前面：应hfu的要求，DP内容只会讲题，知识点将一笔带过，且不会按知识点将题目分类

## P2339 [USACO04OPEN] Turning in Homework G

贝茜有  $C$  ( $1 \leq C \leq 1000$ ) 门科目的作业要上交，之后她要去坐巴士和奶牛同学回家。

每门科目的老师所在的教室排列在一条长为  $H$  ( $1 \leq H \leq 1000$ ) 的走廊上，他们只在课后接收作业，交作业不需要时间。贝茜现在在位置0，她会告诉你每个教室所在的位置，以及走廊出口的位置。她每走1个单位的路程，就要用1秒。她希望你计算最快多久以后她能交完作业并到达出口。

每门作业都只能在  $T_i$  时间后交。

$1 \leq C \leq 1000, 1 \leq H \leq 1000, 0 \leq B \leq H, 0 \leq X_i \leq H, 0 \leq T_i \leq 10000$ 。

## Sol

显然若作业  $X_A < X_B < X_C$ ，如果先去  $B$ ，之后必定还要走到  $A$  再走到  $C$ ，不如先走到  $A$  再去  $C$  的路上顺便交  $B$ 。

要某个时间后才能交作业非常不好处理，来逆向考虑。

如果我们知道最终需要多少时间，从终点跑到起点就相当于要在某个时间之间交完作业。

答案是有单调性的，二分一个最终时间再判断是否可行即可。

先对  $x$  数组排序，令  $f_{l,r,0/1}$  表示交完区间  $[l, r]$  的作业，在左/右端点的最小价值。

转移就是向左右拓展。复杂度  $n^2 \log n$

---

## P8867 [NOIP2022] 建造军营

题目太长不放了。 <https://www.luogu.com.cn/problem/P8867>

简化题意：图上任意选点和边，使得断开任意一条非选边后选的点联通，求方案数。

## Sol

显然割边问题首先缩点成一棵树（点双割边没影响），求方案数就在树上跑dp就行。

令  $f_{u,0/1}$  表示在  $u$  的子树内的所有边和点（不包括  $u \rightarrow fa_u$  这条边），是否选了点的方案数。

注意：点  $u$  是缩完后的点。自身内部有点数和边数，设为  $V_u$  和  $E_u$ 。

每种状态涵盖的情况太多不太好转，一般就加状态数或者增加限制。

令  $f_{u,0/1}$  表示若有军营则必须与  $u$  联通的方案数。

考虑转移。

- $x$  扫到  $y$  之前的子树没有兵营，子树  $y$  内也没有，这条边可选可不选

$$f_{x,0} \leftarrow f_{x,0} \times f_{y,0} \times 2$$

- $x$  子树之前有兵营，子树  $y$  内也没有，这条边可选可不选：

$$f_{x,1} \leftarrow f_{x,1} \times f_{y,0} \times 2$$

- $x$  子树之前有兵营，子树  $y$  内也有，这条边必须选：

$$f_{x,1} \leftarrow f_{x,1} \times f_{y,1}$$

- $x$  子树之前没有兵营，子树  $y$  内也有，这条边必须选：

$$f_{x,1} \leftarrow f_{x,0} \times f_{y,1}$$

更新同时进行，注意拿变量记录一下增加的值。扫完四种情况再加上。

还要考虑当前点内部有原树上的多个点，要乘一下。

统计答案时是强制只选点  $u$  内部的点且  $u \rightarrow fa_u$  的边强制不选，再加上所有点都不选的方案数。

## CF1153F Serval and Bonus Problem

有一段长为  $l$  的线段，有  $n$  个区间，左右端点在  $[0, l]$  间均匀随机（可能不是整数）

求期望被至少  $k$  段区间覆盖的长度，对 998244353 取膜

### Sol

这题我们只关心端点的位置关系，有  $2n$  个端点把线段分为  $2n + 1$  段。

只要对每一段，计算这一段被至少  $k$  段区间覆盖的概率。

令  $f[i][j]$  表示有  $i$  个端点，第  $i$  个端点后面的区间被  $j$  个线段覆盖的方案数。转移直接枚举  $i$  是一个区间的开始还是结束，如果是结束还要乘  $j$  转移，表示这个端点可以匹配前面  $j$  个的任意一个。

顺便维护一个 0/1 表示合法的方案数。最终除以总方案数就是答案。

## [ABC221G] Jumping sequence

有一个无限大的平面直角坐标系，初始时你在  $(0, 0)$  处。给你一个长度为  $n$  的序列  $d$ ，你可以移动  $n$  步，每一步可以选择：

- 向上移动  $d_i$  距离，从  $(x, y)$  到  $(x, y + d_i)$
- 向下移动  $d_i$  距离，从  $(x, y)$  到  $(x, y - d_i)$
- 向右移动  $d_i$  距离，从  $(x, y)$  到  $(x + d_i, y)$
- 向左移动  $d_i$  距离，从  $(x, y)$  到  $(x - d_i, y)$

你想在  $n$  步结束后位于  $(A, B)$  位置，问是否存在这样的方案，如果存在需输出任意一种方案。

- $1 \leq N \leq 2000$
- $|A|, |B| \leq 3.6 \times 10^6$
- $1 \leq D_i \leq 1800$

### Sol

题目有一个很妙的trick，直接DP不可接受，可以把曼哈顿距离转换为切比雪夫距离，即转换为  $(x + y, x - y)$ 。

可以发现一次操作对两维的影响是独立的，对两维分别跑01可行性背包（bitset优化）即可通过。

还有一种  $O(nD)$  的做法自行了解。

## 格子跳跃

有  $n$  个格子从左到右依次挨着，一开始有两枚棋子分布在  $A, B$  某一个或两个格子里，有  $m$  个操作，第  $i$  次操作要求你把其中一个棋子移到  $X_i$  上，移动一个棋子的代价是两个格子之间的距离，求移完所有棋子的代价之和的最小值。所有数据小于  $10^5$ 。

## Sol

Dp! 令  $f_{i,j}$  表示第  $i$  次操作后另一个棋子在  $j$  的方案数。时空起飞，芜湖!!

来仔细看下转移：

- $f_{i,j} = f_{i-1,j} + |x_i - x_{i-1}|$
- $f_{i,x_{i-1}} = f_{i-1,j} + |j - x_i|$

拆掉绝对值，前者是全局加常数，后者是单点修改。

维护下线段树喵。

## P4042 [AHOI2014/JSOI2014] 骑士游戏

在这个游戏中，JYY 一共有两种攻击方式，一种是普通攻击，一种是法术攻击。两种攻击方式都会消耗 JYY 一些体力。采用普通攻击进攻怪兽并不能把怪兽彻底杀死，怪兽的尸体可以变出其他一些新的怪兽，注意一个怪兽可能经过若干次普通攻击后变回一个或更多同样的怪兽；而采用法术攻击则可以彻底将一个怪兽杀死。当然了，一般来说，相比普通攻击，法术攻击会消耗更多的体力值（但由于游戏系统 bug，并不保证这一点）。

游戏世界中一共有  $N$  种不同的怪兽，分别由 1 到  $N$  编号，现在 1 号怪兽入侵村庄了，JYY 想知道，最少花费多少体力值才能将所有村庄中的怪兽全部杀死呢？

对于所有数据  $2 \leq N \leq 2 \times 10^5$ ,  $1 \leq R_i, \sum R_i \leq 10^6$ ,  $1 \leq K_i, S_i \leq 5 \times 10^{14}$ 。

## Sol

DP 的转移形成了环？令  $dp_i$  为消灭第  $i$  只怪兽的最小代价，考虑转移方程：

$$dp_i = \min(K_i, S_i + \sum_{j=1}^{R_i} dp_{v_j})$$

求出每个点的最小值当然是从值最小的点向其他点推。

把所有的  $dp_i$  丢到堆里，每次选最小的一个去更新其他的，当一个点的  $\sum dp_{v_j}$  都被拿出来过也将这个点丢到堆里。

最终答案是  $dp_1$ 。

## P1912 [NOI2009] 诗人小G

题目见链接: <https://www.luogu.com.cn/problem/P1912>

### Sol

式子是naive的, 令  $f_i$  表示选了前  $i$  句的最小不协调度

$$f_i = \min(f_j + |\text{sum}_i - \text{sum}_j + i - j - 1 - L|^P)$$

没有什么办法优化? 可以往单调性的方向思考。

发现随着  $i$  的增大, 决策点也是单增的, 分治即可通过。

---

## P3262 [JLOI2015] 战争调度

给你一棵  $n$  层的满二叉树, 每个节点可以染黑白两种颜色。对于每个叶子节点及其某个祖先节点, 如果它们均为黑色则有一个贡献值, 如果均为白色则有另一个贡献值。要求黑色的叶子节点数目不超过  $m$ , 求最大总贡献值。每一对满足条件的节点都有一个贡献值。

对于 100% 的数据,  $2 \leq n \leq 10$ ,  $m \leq 2^{n-1}$ ,  $0 \leq w_{ij}, f_{ij} \leq 2000$ 。

### Sol

对限制设状态: 令  $f_{u,k,S}$  表示当前点  $u$ ,  $u$  及其子树内染了  $k$  个点,  $u$  到根节点的状态为  $S$  的最大贡献度。

转移就是子树间的背包合并。 $S$  可以在树上向下递归时通过枚举当前点状态来去掉这一维。

---

## CF868F Yet Another Minimization Problem

题目描述: 给定一个序列  $a$ , 要把它分成  $k$  个子段。每个子段的费用是其中相同元素的对数。求所有子段的费用之和的最小值。

$2 \leq n \leq 10^5$ ,  $2 \leq k \leq \min(n, 20)$ ,  $1 \leq a_i \leq n$ 。

### Sol

显然令  $f_{i,j}$  为前  $i$  个数分成  $j$  段的费用最小值。

因为当  $j$  不变,  $i$  增加时  $f_{i,j}$  一定不降, 因此有决策单调性。

我喜欢分治写法。

---

## P5336 [THUSC2016] 成绩单

期末考试结束了，班主任 L 老师要将成绩单分发到每位同学手中。L 老师共有  $n$  份成绩单，按照编号从 1 到  $n$  的顺序叠放在桌子上，其中编号为  $i$  的成绩单分数为  $W_i$ 。

成绩单是按照批次发放的。发放成绩单时，L 老师会从当前的一叠成绩单中抽取连续的一段，让这些同学来领取自己的成绩单。当这批同学领取完毕后，L 老师再从剩余的成绩单中抽取连续的一段，供下一批同学领取。经过若干批次的领取后，成绩单将被全部发放到同学手中。

然而，分发成绩单是一件令人头痛的事情，一方面要照顾同学们的心理情绪，不能让分数相差太远的同学在同一批领取成绩单；另一方面要考虑时间成本，尽量减少领取成绩单的批次数。对于一个分发成绩单的方案，我们定义其代价为：

$$a \times k + b \times \sum_{i=1}^k (\max_i - \min_i)^2$$

其中  $k$  是分发的批次数，对于第  $i$  批分发的成绩单， $\max_i$  是最高分数， $\min_i$  是最低分数， $a$  和  $b$  是给定的评估参数。现在，请你帮助 L 老师找到代价最小的分发成绩单的方案，并将这个最小的代价告诉 L 老师。当然，分发成绩单的批次数  $k$  是你决定的。

$$n \leq 50, a \leq 1500, b \leq 10, w_i \leq 1000.$$

### Sol

看到题显然区间DP，但是这题的区间比较神奇，是抽出一段区间后剩下的拼起来再抽。

DP没法算or不好转移？可能是设计的一种状态覆盖的情况过多，可以考虑增设限制条件。

令  $dp_{l,r,a,b}$  为  $[l,r]$  中选空权值在  $[a,b]$  的最小代价， $ans_{l,r}$  为将  $[l,r]$  选空的最小代价。

$$ans(l,r) = \min\{dp(l,r,a,b) + A + B \times (b-a)^2\}.$$

$$dp(l,r,a,b) = \min\{ans(l,k) + dp(k+1,r,a,b)\}. \text{ (左半区间全部选空, 右半区间剩余 } [a,b] \text{)}$$

$$dp(l,r,a,b) = \min\{dp(l,k,a,b) + ans(k+1,r)\}. \text{ (左半区间剩余 } [a,b], \text{ 右半区间全部选空)}$$

$$dp(l,r,a,b) = \min\{dp(l,k,a,b) + dp(k+1,r,a,b)\}. \text{ (左右区间均都剩余 } [a,b] \text{)}$$

关于  $dp$  和  $ans$  的初值，则显然有： $ans(i,i) = A$ ， $dp(i,i,1 \cdots W_i, W_i \cdots W_{\max}) = 0$ 。

---

## P4749 [CERC2017] Kitchen Knobs

题目见链接：<https://www.luogu.com.cn/problem/P4749>

### Sol

循环周期是1或者7，问题是每次区间加  $x$  并对 7 取模，求将整个序列变成 0 的最小操作数。

区间加差分后就是一次单点加和一次单点减一个相同的数，目标是要将差分序列变成 0。

单点操作是任意的，差分序列上数的出现顺序不再重要，我们只关心每个数出现了几次。

一种暴力的思路是设  $f(c_1, c_2, c_3, c_4, c_5, c_6)$  表示 1 ~ 6 出现的次数，复杂度  $O(n^6)$  级别。

怎样去减少状态？最优状态是什么？

匹配的过程是将匹配的数加起来且和模 7 为 0，能够匹配多少组最终答案就是  $n - \text{匹配的组数}$ 。

注意到 1 6, 2 5, 3 4 优先匹配是最优的，先配对这三组数，然后就剩下了至多三种数，再对这三种数做 dp，复杂度  $O(n^3)$ 。

---

## CF908G

给  $n \leq 10^{700}$ ，问 1 到  $n$  中每个数在各数位排序后得到的数的和。答案膜  $1e9+7$ 。

## Sol

直接进行数位 DP 是不可行的，因为不知道后加的数对当前数的贡献。

对于单个数字  $d$ ，若原数中有  $k$  个大于  $d$  的数，那它对答案的贡献就是  $d \times 10^k$ 。

于是可以对 1 到 9 每一个数字分开统计，令  $f_{i,j,0/1}$  表示放了  $i$  个数，有恰好  $j$  位大于等于  $d$ ，是否贴上线的答案。

---

## P8392 [BalticOI 2022 Day1] Uplifting Excursion

有  $2m + 1$  种物品，重量分别为  $-m, -m + 1, \dots, m - 1, m$ 。重量为  $i$  的物品有  $a_i$  个。

你需要拿走若干物品，使得这些物品重量之和恰好为  $l$ 。在此基础上，你需要拿尽可能多的物品。

问在物品重量之和恰好为  $l$  的基础上，你最多能拿多少物品。

对于所有数据，满足  $1 \leq m \leq 300$ ,  $-10^{18} \leq l \leq 10^{18}$ ,  $0 \leq a_i \leq 10^{12}$ 。

## Sol

注意到数据范围除了  $m$  都不可算，显然应该复杂度应该和  $m$  有关，朴素的背包是不可取的。

先考虑怎么贪，如果所有的物品都选了，肯定是一直扔掉重量最大的。

最终一定会扔到  $[L - m, L + m]$  的区间，因为超出了一定可以通过加减  $\pm m$  的来回到区间。

再二进制分组优化一下，复杂度  $m^3 \log m$ 。

---

## P4007 小 Y 和恐怖的奴隶主

小 Y 是一个喜欢玩游戏的 Oler。一天，她正在玩一款游戏，要打一个 Boss。

虽然这个 Boss 有  $10^{100}$  点生命值，但它只带了一个随从——一个只有  $m$  点生命值的“恐怖的奴隶主”。

这个“恐怖的奴隶主”有一个特殊的技能：每当它被扣减生命值但没有死亡（死亡即生命值  $\leq 0$ ），且 Boss 的随从数量小于上限  $k$ ，便会召唤一个新的具有  $m$  点生命值的“恐怖的奴隶主”。

现在小 Y 可以进行  $n$  次攻击，每次攻击时，会从 Boss 以及 Boss 的所有随从中的等概率随机选择一个，并扣减 1 点生命值，她想知道进行  $n$  次攻击后扣减 Boss 的生命值点数的期望。为了避免精度误差，你的答案需要对 998244353 取模。

在所有测试点中， $1 \leq T \leq 1000, 1 \leq n \leq 10^{18}, 1 \leq m \leq 3, 1 \leq k \leq 8$ 。

## Sol

显然看到  $n \leq 10^{18}$  的操作想到矩阵快速幂转移状态。

不难设状态  $f(t, a, b, c)$  表示时刻  $t$  生命值为 1, 2, 3 的随从分别有  $a, b, c$  个。

枚举一下发现合法的状态  $S$  大约有 170 个，直接暴力矩阵求解的复杂度约为  $T|S|^3 \log_2 n$  无法通过。

怎么优化掉重复的操作？

注意到转移矩阵是不变的，而  $f$  的递推过程是用初始向量去乘转移矩阵的  $n$  次方。

预处理出转移矩阵的  $2^k$  次方，即可在  $O((T|S|^2 + |S|^3) \log_2 n)$  的时间内解决。（直接并行运算）

---

## CF1481F AB Tree

给定一棵  $n(n \leq 10^5)$  个节点的树，根为 1，每个节点会分配到一个字符 **a** 或 **b**。

要求整棵树中字符 **a** 的数量为  $x$ ，字符 **b** 的数量为  $n - x$ 。

定义节点  $v$  上的字符串：

- 若  $v$  是根节点，则  $v$  的上的字符串为根节点分配到的字符。
- 否则， $v$  上的字符串为父节点上的字符串的末尾加上  $v$  分配到的字符。

请为每个节点分配字符，在满足字符 **a**，**b** 数量要求的前提下，使得所有节点上的字符串的种类最少。

## Sol

设树的深度是  $D$ ，显然答案的下界是  $D$ 。答案的上界是什么？

上界是  $D + 1$ ，最坏情况是其他深度的点都是同一种字符，而深度  $k$  有两种字符染色。



当第  $k$  层两种颜色都不够染完这一层的所有点时，由于每一个当前层的非叶子节点子树内都至少有一个叶子，所以两种颜色中至少有一种不少于当前层的非叶节点个数，选择这种颜色染这一层的非叶节点和若干个叶子，剩余的点染另一种颜色（这种颜色被染完）。

剩余的工作时判断答案是否可能等于  $D$ 。

答案等于  $D$  等价于每一层的颜色相同，否则这一层会产生两种不同的串。

问题转化为判定是否存在一个深度的子集，深度在这个子集内的点数为  $x$ 。

分析子集？01背包！！

暴力跑 01背包时  $O(n^2)$  的。

计  $c_d$  为深度为  $d$  的点，考虑不同体积的  $c_d$  最多有多少种：由于  $c_d$  之和为  $n$ ，不同的  $c_d$  只有  $\sqrt{n}$  种。二进制优化后为  $n\sqrt{n}\log_2 n$ 。我们只关心方案是否存在，用bitset卡卡就过了。

如何找到方案？

令  $f_{i,j}$  为前  $i$  种数子集和为  $j$  的方案是否存在，若当前还剩  $A$  个点，只需找到第一次  $f_{?,A}$  为  $true$  的时候将这一层全刷为颜色  $a$ 。

---

## CF1188D Make Equal

给出  $n(n \leq 10^5)$  个数字  $a_1, a_2, \dots, a_n (a_n \leq 10^{17})$ ，每次操作可以给其中一个数加上 2 的非负整数次幂。求最少的操作次数，使得这  $n$  个数相等。

## Sol

将所有数都变成  $z$  的代价时  $\sum \text{popcount}(z - a_i)$ 。不妨将  $a_i$  降序排列， $z = a_1 + t$ ，另设  $b_i = a_1 - a_i$ 。

则  $z - a_i = b_i + t$ ，问题变为给定序列  $b$ ，整体加  $t$  后的最小  $\text{popcount}$  之和。

注意到  $b_i + t$  的低位不会受  $t$  的高位影响，这启发我们从低位到高位递推  $t$ 。

朴素地，我们设  $f(i, S)$  表示当前考虑到了  $t$  的第  $i$  位，集合  $S$  中的元素在第  $i$  位发生了进位，前  $i$  位  $\text{popcount}$  之和的最小值。

考虑  $f(i, S)$  可以贡献到哪些状态，有两种决策： $t$  的这二位填 0/1。转移是快速的。

$S$  看似很大，但二进制下第  $i$  位越容易进位的数在模  $2^i$  下越大，处理每一位时将  $b_i$  按模  $2^i$  排序，进位的  $j$  个数一定是排序后最大的  $j$  个数，因此是  $O(n)$  的。

总复杂度  $O(n \log_2 n \log_2 a)$ 。

---

留一道要仔细推的树上dp：[P2081 \[NOI2012\] 迷失游乐园](#)

以及终极区间dp: [P10041 \[CCPC 2023 北京市赛\] 史莱姆工厂](#)

还有若只数位DP: [P3281 \[SCOI2013\] 数数](#) 后面这道究极数位DP仅供参考, 别浪费时间: ~~CF582D~~