

动态规划

石家庄二中集训 高一

张若天

2021 年 2 月 3 日

清华大学 交叉信息研究院

大家好!

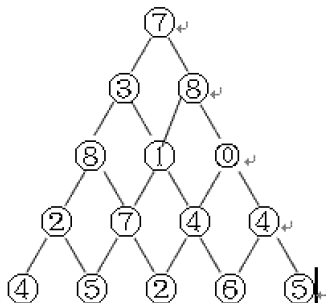
- 大家好!
- 我是张若天, 石家庄二中 2016 届。
- NOI2015 一等奖, NOIP2014 一等奖, 现在在清华大学交叉信息院读研究生。
- 有问题欢迎随时提问 ~

基本概念

经典入门题

数字三角形

如图所示的数字三角形，从顶部出发，在每一结点可以选择向左走或得向右走，一直走到底层，要求找出一条路径，使路径上的值的和最大。



- 枚举所有路径，取最大值。

- 枚举所有路径，取最大值。
- n 层的三角形的路径数量是 2^n 。

```
int f(int x,int y){  
    if(x==n){  
        return a[x][y];  
    }  
    return a[x][y]+max(f(x+1,y),f(x+1,y+1));  
}  
int ans=f(1,1);
```

```
int f(int x,int y){  
    if(x==n){  
        return a[x][y];  
    }  
    return a[x][y]+max(f(x+1,y),f(x+1,y+1));  
}  
int ans=f(1,1);
```

复杂度 2^n 。


```
int f(int x,int y){  
    if(x==n){  
        return a[x][y];  
    }  
    return a[x][y]+max(f(x+1,y),f(x+1,y+1));  
}  
int ans=f(1,1);
```

复杂度 2^n 。

显然对于确定的 x,y , $f(x,y)$ 的返回值是一定的, 与前面如何走到 x,y 无关。

```
int f(int x,int y){  
    if(x==n){  
        return a[x][y];  
    }  
    return a[x][y]+max(f(x+1,y),f(x+1,y+1));  
}  
int ans=f(1,1);
```

复杂度 2^n 。

显然对于确定的 x,y , $f(x,y)$ 的返回值是一定的, 与前面如何走到 x,y 无关。

所以考虑能否把所有的 $f(x,y)$ 只计算一次。

```
int f(int x,int y){
    if(vis[x][y]) return g[x][y];
    else vis[x][y]=1;
    if(x==n){
        return g[x][y]=a[x][y];
    }
    return g[x][y]=a[x][y]+max(f(x+1,y),f(x+1,y+1));
}
int ans=f(1,1);
```

```
int f(int x,int y){  
    if(vis[x][y]) return g[x][y];  
    else vis[x][y]=1;  
    if(x==n){  
        return g[x][y]=a[x][y];  
    }  
    return g[x][y]=a[x][y]+max(f(x+1,y),f(x+1,y+1));  
}  
int ans=f(1,1);
```

复杂度 n^2 。

```
int f(int x,int y){
    if(vis[x][y]) return g[x][y];
    else vis[x][y]=1;
    if(x==n){
        return g[x][y]=a[x][y];
    }
    return g[x][y]=a[x][y]+max(f(x+1,y),f(x+1,y+1));
}
int ans=f(1,1);
```

复杂度 n^2 。

这种方法也叫作记忆化搜索。

```
int f(int x,int y){
    if(vis[x][y]) return g[x][y];
    else vis[x][y]=1;
    if(x==n){
        return g[x][y]=a[x][y];
    }
    return g[x][y]=a[x][y]+max(f(x+1,y),f(x+1,y+1));
}
int ans=f(1,1);
```

复杂度 n^2 。

这种方法也叫作**记忆化搜索**。

记忆化搜索在满足 1)函数返回结果只与参数有关 2)同样参数的函数会被多次调用 的情况下可以优化算法。

```
int f(int x,int y){
    if(vis[x][y]) return g[x][y];
    else vis[x][y]=1;
    if(x==n){
        return g[x][y]=a[x][y];
    }
    return g[x][y]=a[x][y]+max(f(x+1,y),f(x+1,y+1));
}
int ans=f(1,1);
```

复杂度 n^2 。

这种方法也叫作**记忆化搜索**。

记忆化搜索在满足 1)函数返回结果只与参数有关 2)同样参数的函数会被多次调用 的情况下可以优化算法。

是一种用空间换时间的方式。

上面函数的递归的作用只是确定了计算顺序。
为什么不只用数组呢？


```

int f(int x,int y){
    if(vis[x][y]) return g[x][y];
    else vis[x][y]=1;
    if(x==n){
        return g[x][y]=a[x][y];
    }
    return g[x][y]=a[x][y]+max(f(x+1,y),f(x+1,y+1));
}
int ans=f(1,1);
-----
int f[1005][1005];
for(int i=1;i<=n;i++) f[n][i]=a[n][i];
for(int i=n-1;i>=1;i--){
    for(int j=1;j<=i;j++){
        f[i][j]=a[i][j]+max(f[i+1][j],f[i+1][j+1]);
    }
}
ans=f[1][1]

```

- 上面这两种优化方式叫做动态规划。
- 动态规划的两种实现方式：记忆化搜索，递推。
- 上面函数的参数叫做动态规划中的状态。
- 从一个状态计算出另一个状态叫做状态转移。

注意：一般情况下有明确的计算顺序才可以使用递推方法。

滑雪

你在一个 $n \times n$ 的带权值的网格上滑雪，权值表示每个格子的高度，每一秒可以决定往上下左右四个方向滑，但是只能滑向更低的地方。求从哪个格子开始滑雪可以滑的时间最长。

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

- 处理动态规划题目设计状态十分重要。
- 状态既要能完全表示当前状态，又不能有冗余信息。
- 还要求有最优子结构、无后效性的性质。

关于最优子结构

最优子结构：把原问题化到规模更小的问题后，原问题的最优解一定能从规模更小问题的最优解推出。

数字三角形 W
数字三角形

要求走到最后和 $\text{mod } 100$ 最大。

在 $\text{mod } 100$ 这个条件下，刚刚的状态就没有了最优子结构。(为什么?)

关于最优子结构

最优子结构：把原问题化到规模更小的问题后，原问题的最优解一定能从规模更小问题的最优解推出。

数字三角形 W
数字三角形

要求走到最后和 $\text{mod } 100$ 最大。

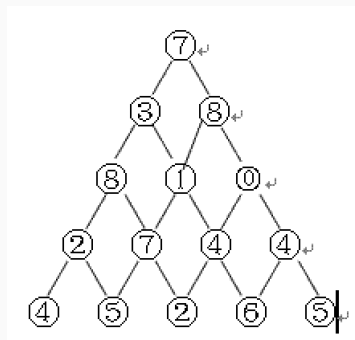
在 $\text{mod } 100$ 这个条件下，刚刚的状态就没有了最优子结构。(为什么?)
一般可以通过加状态维数来解决。

- 状态
- 状态转移方程
- 阶段

动态规划实际上是将本质相同（后续转移决策相同）的大量信息压缩成“单一状态”，将本质相同的大量转移压缩成“单一转移”。

数字三角形 WWW

数字三角形必须经过某一个点，使之走的路程和最大。



传球游戏

n 个同学站成一个圆圈，其中的一个同学手里拿着一个球，当老师吹哨子时开始传球，每个同学可以把球传给自己左右的两个同学中的一个（左右任意），当老师再次吹哨子时，传球停止，此时，拿着球没传出去的那个同学就是败者，要给大家表演一个节目。

聪明的小蛮提出一个有趣的问题：有多少种不同的传球方法可以使得从小蛮手里开始传的球，传了 m 次以后，又回到小蛮手里。两种传球的方法被视作不同的方法，当且仅当这两种方法中，接到球的同学按接球顺序组成的序列是不同的。比如有 3 个同学 1 号、2 号、3 号，并假设小蛮为 1 号，球传了 3 次回到小蛮手里的方式有 1->2->3->1 和 1->3->2->1，共 2 种。

tyvj1008/P5888

最长上升子序列 (LIS)

给一个数组 a_1, a_2, \dots, a_n ，找到最长的上升子序列

$a_{b_1} < a_{b_2} < \dots < a_{b_k}$ ，其中 $b_1 < b_2 < \dots < b_k$ 。

$$n \leq 5 \times 10^3$$

最长上升子序列 (LIS)

给一个数组 a_1, a_2, \dots, a_n ，找到最长的上升子序列

$a_{b_1} < a_{b_2} < \dots < a_{b_k}$ ，其中 $b_1 < b_2 < \dots < b_k$ 。

$$n \leq 5 \times 10^3$$

$$n \leq 10^5$$

最长上升子序列 (LIS)

给一个数组 a_1, a_2, \dots, a_n ，找到最长的上升子序列

$a_{b_1} < a_{b_2} < \dots < a_{b_k}$ ，其中 $b_1 < b_2 < \dots < b_k$ 。

$$n \leq 5 \times 10^3$$

$$n \leq 10^5$$

栈优化 (/ 树状数组优化): 维护最可能的最长上升子序列。

最长上升子序列 (LIS)

给一个数组 a_1, a_2, \dots, a_n , 找到最长的上升子序列

$a_{b_1} < a_{b_2} < \dots < a_{b_k}$, 其中 $b_1 < b_2 < \dots < b_k$ 。

$$n \leq 5 \times 10^3$$

$$n \leq 10^5$$

栈优化 (/ 树状数组优化): 维护最可能的最长上升子序列。

练习: poj 1050 最大子矩形, P1854 花店橱窗布置

最长公共子序列 (LCS)

给两个数组 a_1, a_2, \dots, a_n , b_1, b_2, \dots, b_n 找到最长的公共子序列 $a_{c_1}, a_{c_2}, \dots, a_{c_k}$, 和 $b_{d_1}, b_{d_2}, \dots, b_{d_k}$, 其中 $c_1 < c_2 < \dots < c_k$, $d_1 < d_2 < \dots < d_k$, 满足 $a_{c_1} = b_{d_1}, a_{c_2} = b_{d_2}, \dots, a_{c_k} = b_{d_k}$ 。

$$n \leq 10^3$$

5

3 2 1 1 5

1 2 3 2 5

P1439 **【模板】最长公共子序列**
求两个排列 P_1, P_2 的 LCS 长度。

$$n \leq 10^5$$

5

3 2 1 4 5

1 2 3 4 5

P1439 **【模板】最长公共子序列**
求两个排列 P_1, P_2 的 LCS 长度。

$$n \leq 10^5$$

5

3 2 1 4 5

1 2 3 4 5

如果不是排列？例如每个数字出现次数 ≤ 10 次。

P1280 尼克的任务

尼克每天上班之前都连接上英特网，接收他的上司发来的邮件，这些邮件包含了尼克主管的部门当天要完成的全部任务，每个任务由一个开始时刻与一个持续时间构成。

尼克的一个工作日为 N 分钟，从第 1 分钟开始到第 N 分钟结束。当尼克到达单位后他就开始干活。如果在同一时刻有多个任务需要完成，尼克可以任选其中的一个来做，而其余的则由他的同事完成，反之如果只有一个任务，则该任务必需由尼克去完成，假如某些任务开始时刻尼克正在工作，则这些任务也由尼克的同事完成。如果某任务于第 P 分钟开始，持续时间为 T 分钟，则该任务将在第 $P+T-1$ 分钟结束。

写一个程序计算尼克应该如何选取任务，才能获得最大的空暇时间。

$$n \leq 10^4$$

最大正方形

Luogu P1387

在一个 $n*m$ 的只包含 0 和 1 的矩阵里找出一个不包含 0 的**最大正方形**，输出边长。

$n, m \leq 100$

对于最后队列的一段区间 $[i, j]$ ，不论这一段区间如何插入，除了最后一个插入的对象外，剩下的对后续插入没有影响。

$F[L][R][0/1]$ 表示区间 $[L, R]$ 最后插入的是 L 或 R，方案数。

枚举转移即可。

逆序对数列

对于一个数列 $\{a_i\}$ ，如果有 $i < j$ 且 $a_i > a_j$ ，那么我们称 a_i 与 a_j 为一对逆序对数。若对于任意一个由 $1 \sim n$ 自然数组成的数列，可以很容易求出有多少个逆序对数。那么逆序对数为 k 的这样自然数数列到底有多少个？由于这个数可能很大，你只需输出该数对 10000 求余数后的结果。 $n \leq 1000, k \leq 1000$

bzoj2431

逆序对数列

对于一个数列 $\{a_i\}$ ，如果有 $i < j$ 且 $a_i > a_j$ ，那么我们称 a_i 与 a_j 为一对逆序对数。若对于任意一个由 $1 \sim n$ 自然数组成的数列，可以很容易求出有多少个逆序对数。那么逆序对数为 k 的这样自然数数列到底有多少个？由于这个数可能很大，你只需输出该数对 10000 求余数后的结果。 $n \leq 1000, k \leq 1000$

bzoj2431

$F[i][j]$ 表示 $1 \sim i$ 的排列有 j 个逆序对有多少个。

逆序对数列

对于一个数列 $\{a_i\}$ ，如果有 $i < j$ 且 $a_i > a_j$ ，那么我们称 a_i 与 a_j 为一对逆序对数。若对于任意一个由 $1 \sim n$ 自然数组成的数列，可以很容易求出有多少个逆序对数。那么逆序对数为 k 的这样自然数数列到底有多少个？由于这个数可能很大，你只需输出该数对 10000 求余数后的结果。 $n \leq 1000, k \leq 1000$

bzoj2431

$F[i][j]$ 表示 $1 \sim i$ 的排列有 j 个逆序对有多少个。前缀和优化。

P2679[NOIP2015 提高组] 子串

有两个仅包含小写英文字母的字符串 A 和 B 。

现在要从字符串 A 中取出 k 个互不重叠的非空子串，然后把这 k 个子串按照其在字符串 A 中出现的顺序依次连接起来得到一个新的字符串。请问有多少种方案可以使得这个新串与字符串 B 相等？

注意：子串取出的位置不同也认为是不同的方案。

$$n \leq 1000, m \leq 200, k \leq m$$

6 3 2
aabaab
aab

7

区间动态规划

沙子合并

设有 N 堆沙子排成一排，其编号为 $1, 2, 3, \dots, N$ ($N \leq 300$)。每堆沙子有一定的数量，可以用一个整数来描述，现在要将这 N 堆沙子合并成为一堆。每次只能合并相邻的两堆，合并的代价为这两堆沙子的数量之和，合并后与这两堆沙子相邻的沙子将和新堆相邻，合并时由于选择的顺序不同，合并的总代价也不相同。如有 4 堆沙子分别为 1, 3, 5, 2，我们可以先合并 1、2 堆，代价为 4，得到 4, 5, 2，又合并 1, 2 堆，代价为 9，得到 9, 2，再合并得到 11，总代价为 $4 + 9 + 11 = 24$ ，如果第二步是先合并 2, 3 堆，则代价为 7，得到 4 7，最后一次合并代价为 11，总代价为 $4 + 7 + 11 = 22$ ；问题是：找出一种合理的方法，使总的代价最小。输出最小代价。

$$n \leq 300$$

tyvj1055

环形情况？

常用技巧：将环断开，序列加倍。

P1880 [NOI1995] 石子合并

在一个圆形操场的四周摆放 N 堆石子, 现要将石子有次序地合并成一堆. 规定每次只能选相邻的 2 堆合并成新的一堆, 并将新的一堆的石子数, 记为该次合并的得分。

试设计出一个算法, 计算出将 N 堆石子合并成 1 堆的最小得分和最大得分。

$$n \leq 300$$

P1063 [NOIP2006 提高组] 能量项链

在 Mars 星球上，每个 Mars 人都随身佩带着一串能量项链。在项链上有 N 颗能量珠。能量珠是一颗有头标记与尾标记的珠子，这些标记对应着某个正整数。并且，对于相邻的两颗珠子，前一颗珠子的尾标记一定等于后一颗珠子的头标记。因为只有这样，通过吸盘（吸盘是 Mars 人吸收能量的一种器官）的作用，这两颗珠子才能聚合成一颗珠子，同时释放出可以被吸盘吸收的能量。如果前一颗能量珠的头标记为 m ，尾标记为 r ，后一颗能量珠的头标记为 r ，尾标记为 n ，则聚合后释放的能量为（Mars 单位），新产生的珠子的头标记为 m ，尾标记为 n 。需要时，Mars 人就用吸盘夹住相邻的两颗珠子，通过聚合得到能量，直到项链上只剩下一颗珠子为止。显然，不同的聚合顺序得到的总能量是不同的，请你设计一个聚合顺序，使一串项链释放出的总能量最大。

例如：设 $N=4$ ，4 颗珠子的头标记与尾标记依次为 $(2, 3)$ $(3, 5)$ $(5, 10)$ $(10, 2)$ 。我们用记号 $@$ 表示两颗珠子的聚合操作， $(j@k)$ 表示第 j ， k 两颗珠子聚合后所释放的能量。则第 4、1 两颗珠子聚合后释放的能量为： $(4@1)=10*2*3=60$ 。这一串项链可以得到最优值的一个聚合顺序所释放的总能量为 $((4@1)@2)@3 = 10*2*3+10*3*5+10*5*10=710$ 。

tyvj1056/Luogu 1063

关路灯 (Luogu P1220)

某一村庄在一条路线上安装了 n 盏路灯，每盏灯的功率有大有小。

老张就住在这条路中间某一路灯旁，他有一项工作就是每天早上天亮时一盏一盏地关掉这些路灯。

他每天都是在天亮时首先关掉自己所处位置的路灯，然后可以向左也可以向右去关灯。

请你为老张编一程序来安排关灯的顺序，使从老张开始关灯时刻算起所有灯消耗电最少。

$$n \leq 1000$$

关路灯 (Luogu P1220)

某一村庄在一条路线上安装了 n 盏路灯，每盏灯的功率有大有小。

老张就住在这条路中间某一路灯旁，他有一项工作就是每天早上天亮时一盏一盏地关掉这些路灯。

他每天都是在天亮时首先关掉自己所处位置的路灯，然后可以向左也可以向右去关灯。

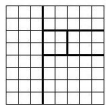
请你为老张编一程序来安排关灯的顺序，使从老张开始关灯时刻算起所有灯消耗电最少。

$$n \leq 1000$$

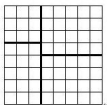
动机：任何时刻关掉的路灯都是连续的区间。

POJ 1191 棋盘分割

将一个 $8 * 8$ 的棋盘进行如下分割：将原棋盘割下一块矩形棋盘并使剩下部分也是矩形，再将剩下的部分继续如此分割，这样割了 $(n-1)$ 次后，连同最后剩下的矩形棋盘共有 n 块矩形棋盘。（每次切割都只能沿着棋盘格子的边进行）



允许的分割方案



不允许的分割方案

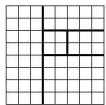
原棋盘上每一格有一个分值，一块矩形棋盘的总分为其所含各格分值之和。现在需要把棋盘按上述规则分割成 n 块矩形棋盘，并使各矩形棋盘总分的**均方差**最小。

均方差 $\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$ ，其中平均值 $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$ ， x_i 为第 i 块矩形棋盘的总分。请编程对给出的棋盘及 n ，求出 σ 的最小值。

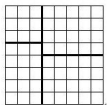
$$n \leq 15$$

POJ 1191 棋盘分割

将一个 $8 * 8$ 的棋盘进行如下分割：将原棋盘割下一块矩形棋盘并使剩下部分也是矩形，再将剩下的部分继续如此分割，这样割了 $(n-1)$ 次后，连同最后剩下的矩形棋盘共有 n 块矩形棋盘。（每次切割都只能沿着棋盘格子的边进行）



允许的分割方案



不允许的分割方案

原棋盘上每一格有一个分值，一块矩形棋盘的总分为其所含各格分值之和。现在需要把棋盘按上述规则分割成 n 块矩形棋盘，并使各矩形棋盘总分的**均方差**最小。

均方差 $\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$ ，其中平均值 $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$ ， x_i 为第 i 块矩形棋盘的总分。请编程对给出的棋盘及 n ，求出 σ 的最小值。

$n \leq 15$ ，二维区间 DP

玩具取名

某人有一套玩具，并想法给玩具命名。首先他选择 WING 四个字母中的任意一个字母作为玩具的基本名字。然后他会根据自己的喜好，将名字中任意一个字母用“WING”中的某两个字母代替，使得自己的名字能够扩充得很长。现在，他想请你猜猜某一个很长的名字，最初可能是由哪个字母变形过来的。

Len ≤ 200 , 每个字母可替换的两个字母组数 ≤ 16

P4290

玩具取名

某人有一套玩具，并想法给玩具命名。首先他选择 WING 四个字母中的任意一个字母作为玩具的基本名字。然后他会根据自己的喜好，将名字中任意一个字母用“WING”中的某两个字母代替，使得自己的名字能够扩充得很长。现在，他想请你猜猜某一个很长的名字，最初可能是由哪个字母变形过来的。

Len ≤ 200 , 每个字母可替换的两个字母组数 ≤ 16

P4290 F[i][j][k] i-j 这一段是否可以用 k 扩展过来。

P3205 [HNOI2010] 合唱队

为了在即将到来的晚会有更好的演出效果，作为 AAA 合唱队负责人的小 A 需要将合唱队的人根据他们的身高排出一个队形。假定合唱队一共 n 个人，第 i 个人的身高为 h_i 米 ($1000 \leq h_i \leq 2000$)，并已知任何两个人的身高都不同。假定最终排出的队形是 A 个人站成一排，为了简化问题，小 A 想出了如下排队的方式：他让所有的人先按任意顺序站成一个初始队形，然后从左到右按以下原则依次将每个人插入最终排出的队形中：

- 第一个人直接插入空的当前队形中。
- 对从第二个人开始的每个人，如果 he 比前面那个人高 (h 较大)，那么将他插入当前队形的最右边。如果他比前面那个人矮 (h 较小)，那么将他插入当前队形的最左边。

当 n 个人全部插入当前队形后便获得最终排出的队形。

小 A 心中有一个理想队形，他想知道多少种初始队形可以获得理想的队形。

请求出答案对 19650827 取模的值。

$$n \leq 1000$$

P1006 [NOIP2008 提高组] 传纸条

在一个矩阵内找出两条从 $1,1$ 到 m,n 的路径（一条从 $1,1$ 到 m,n 一条从 m,n 到 $1,1$ ，除了起点终点外不能相交），并且路径上的权值之和最大。

$n, m \leq 50$

背包问题

部分背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：

Task 1. 物品可以拿任意实数份

Task 2. 物品可以拿 $0 \sim 1$ 内任意实数份。

部分背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：

Task 1. 物品可以拿任意实数份

Task 2. 物品可以拿 $0 \sim 1$ 内任意实数份。

贪心

完全背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：物品可以拿任意自然数份。

完全背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：物品可以拿任意自然数份。刚刚的贪心为什么不对？

完全背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：物品可以拿任意自然数份。刚刚的贪心为什么不对？

计数？Luogu 1474 货币系统

01 背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：物品可以拿 0 份或 1 份。

01 背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：物品可以拿 0 份或 1 份。

练习：开心的金明，金明的预算方案。

多重背包

有 n 个物品，每个物品有体积 v_i 和价值 w_i ，你有一个体积为 V 的背包，问最多能装多少价值的物品？

其中：第 i 物品可以最多拿 a_i 份 (整数)。

飞扬的小鸟

<https://www.luogu.com.cn/problem/P1941>

【跳过】 Sliding Window

给定一个大小已知的数组以及一个大小已知的滑动窗口，窗口每个时刻向后移动一位，求出每个时刻窗口中数字的最大值和最小值。

poj2823

【跳过】单调队列优化多重背包

对于每个物品体积 v_i ，分成 v_i 类转移。

《背包九讲》

<https://raw.githubusercontent.com/tianyicui/pack/master/V2.pdf>

树上的动态规划

树的直径

给一棵树，求树上的最长链。

$$n \leq 10^5$$

树的直径

给一棵树，求树上的最长链。

$$n \leq 10^5$$

- 树形 DP
- 两遍 BFS

树的重心

给一棵树，求树的**重心**，即删掉某个点后剩下的**最大连通块大小最小**。

$$n \leq 10^5$$

P1352 没有上司的舞会

Ural 大学有 N 个职员，编号为 $1-N$ 。他们有从属关系，也就是说他们的关系就像一棵以校长为根的树，父结点就是子结点的直接上司。每个职员有一个快乐指数。现在有个周年庆宴会，要求与会职员的快乐指数最大。但是，没有职员愿和直接上司一起与会。

$$n \leq 10^5$$

选课

在大学里每个学生，为了达到一定的学分，必须从很多课程里选择一些课程来学习，在课程里有些课程必须在某些课程之前学习，如高等数学总是在其它课程之前学习。现在有 N 门功课，每门课有个学分，每门课有一门或没有直接先修课（若课程 a 是课程 b 的先修课即只有学完了课程 a ，才能学习课程 b ）。一个学生要从这些课程里选择 M 门课程学习，问他能获得的最大学分是多少？

luogu2014

选课

在大学里每个学生，为了达到一定的学分，必须从很多课程里选择一些课程来学习，在课程里有些课程必须在某些课程之前学习，如高等数学总是在其它课程之前学习。现在有 N 门功课，每门课有个学分，每门课有一门或没有直接先修课（若课程 a 是课程 b 的先修课即只有学完了课程 a ，才能学习课程 b ）。一个学生要从这些课程里选择 M 门课程学习，问他能获得的最大学分是多少？

luogu2014

树形背包

Easy: <http://noi.openjudge.cn/ch0206/>

百度/Google POJ DP 列表

<https://oi-wiki.org/dp/>

- POJ 3321 Apple Tree
- POJ 1155 TELE
- POJ 1947 Rebuilding Roads

BZOJ 1037: [ZJOI2008] 生日聚会 Party

参加 party 的人中共有 n 个男孩与 m 个女孩，打算坐成一排玩游戏，就座的方案应满足如下条件：对于任意连续的一段，男孩与女孩的数目之差不超过 k 。

求方案数除以 12345678 的余数。

$$n, m \leq 150, k \leq 20$$

BZOJ 1037: [ZJOI2008] 生日聚会 Party

$dp[i][j][k][l]$ 前 i 男 j 女，男减女最大为 k ，女减男最大为 l 的方案数。

BZOJ 1037: [ZJOI2008] 生日聚会 Party

$dp[i][j][k][l]$ 前 i 男 j 女，男减女最大为 k ，女减男最大为 l 的方案数。
枚举当前第 $i + j + 1$ 个人是男是女即可。

Questions?

谢谢大家！

感谢石家庄二中的培养。

Email: bkt@qq.com

QQ: 401794301

Telegram: @Ruotian

<https://zrt.io>



ℒ_Tℒ_X