



# 算法基础

郭炜 微博 <http://weibo.com/guoweiofpku>

<http://blog.sina.com.cn/u/3266490431>

刘家瑛 微博 <http://weibo.com/pkuliujiaying>



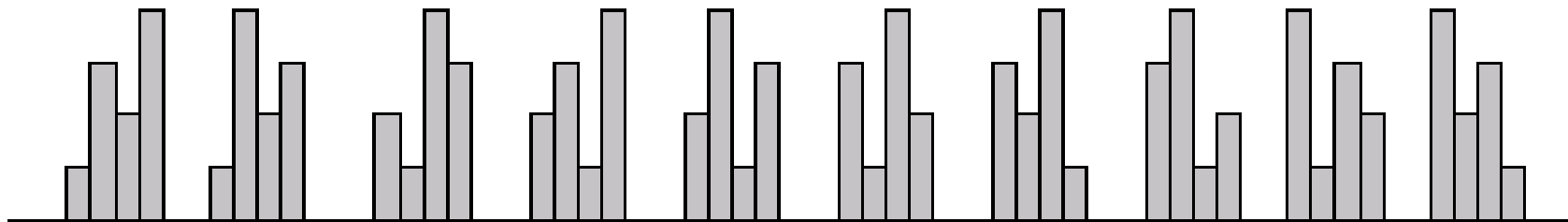
北京大学  
PEKING UNIVERSITY

# 动态规划

美丽栅栏

## 例题： POJ 1037 一个美妙的栅栏

- $N$  个木棒, 长度分别为  $1, 2, \dots, N$ .
- 构成美妙的栅栏
  - 除了两端的木棒外, 每一跟木棒, 要么比它左右的两根都长, 要么比它左右的两根都短。
  - 即木棒呈现波浪状分布, 这一根比上一根长了, 那下一根就比这一根短, 或反过来



All cute fences made of  $N = 4$  planks, ordered by their catalogue numbers.

## 例题： POJ 1037 一个美妙的栅栏

- 问题：符合上述条件的栅栏建法有很多种，对于满足条件的所有栅栏，按照字典序(从左到右，从低到高)排序。
- 给定一个栅栏的排序号，请输出该栅栏，即每一个木棒的长度。

## 例题： POJ 1037 一个美妙的栅栏

- 输入数据
  - 第一行是测试数据的组数  $K$  ( $1 \leq K \leq 100$ )。接下来的 $K$ 行，每一行描述一组输入数据。
  - 每一组输入数据包括两个整数  $N$  和  $C$ 。  $N$  ( $1 \leq N \leq 20$ ) 表示栅栏的木棒数,  $C$ 表示要找的栅栏的排列号。
- 输出数据
  - 输出第 $C$ 个栅栏，即每一个木棒的长度
- 设20个木棒可组成的栅栏数是 $T$ ；我们假设  $T$  可以用64-bit长整数表示，  $1 < C \leq T$

- 输入样例

2

2 1

3 3

- 输出样例

1 2

2 3 1

# 解题思路

- 问题抽象：给定1到N 这N个数字,将这些数字高低交替进行排列，把所有符合情况的进行一个字典序排列,问第C个排列是一个怎样的排列
- 总体思想
  - 动归 + 排列计数
- 动归

# 动归解题思路

- 1) 设  $A[i]$  为  $i$  根木棒所组成的合法方案数目。看看能否找出  $A[i]$  和  $A[i-1]$  或  $A[i-j]$  之间的递推关系(所有木棒总数是  $i$ )。称  $i$  根木棒的合法方案集合为  $S(i)$
- 2) 在选定了某根木棒  $x$  作为第一根木棒的情况下, 剩下  $i-1$  根木棒的合法方案数是  $A[i-1]$ 。但是, 这  $A[i-1]$  种方案, 并不是每种都能和  $x$  形成新的合法方案。将第一根比第二根长的方案称为 DOWN 方案, 第一根比第二根短的称为 UP 方案, 则,  $S(i-1)$  中, 第一根木棒比  $x$  长的 DOWN 方案, 以及第一根木棒比  $x$  短的 UP 方案, 才能和  $x$  构成  $S(i)$  中的方案。



## 动归解题思路

- 3) 置  $A[i] = 0$ 。先枚举  $S(i)$  中的第一根木棒  $x$ 。然后针对每个  $x$ ，枚举  $x$  后面的那根木棒  $y$ 。如果  $y > x$  ( $x < y$  的情况类推)，则：

$A[i] +=$  以  $y$  打头的 DOWN 方案数

但以  $y$  打头的 DOWN 方案数，又和  $y$  的长短有关。

于是难以直接从  $A[i-1]$  或  $A[i-j]$  推出  $A[i]$

# 动归解题思路

- 4) 考虑将 $A[i]$ 这种粗略的状态描述方式细化，即加上限制条件后分类。设

$$A[i] = \sum B[i][k] \quad k = 1 \dots i$$

$B[i][k]$  是 $S(i)$ 中以第 $k$ 短的木棒打头的方案数。尝试对  $B$  进行动归。第 $k$ 短，指的是 $i$ 根木棒中第 $k$ 短。

# 递归解题思路

- 5)  $B[i][k] = \sum B[i-1][M]_{(DOWN)} + \sum B[i-1][N]_{(UP)}$

$$M = k \dots i-1, N = 1 \dots k-1$$

还是没法直接推。于是把B再分类细化：

$$B[i][k] = C[i][k][DOWN] + C[i][k][UP]$$

$C[i][k][DOWN]$  是S(i)中以第k短的木棒打头的DOWN方案数。然后试图对C进行递归

$$C[i][k][UP] = \sum C[i-1][M][DOWN]$$

$$M = k \dots i-1$$

$$C[i][k][DOWN] = \sum C[i-1][N][UP]$$

$$N = 1 \dots k-1$$

初始条件:  $C[1][1][UP]=C[1][1][DOWN] = 1$

# 递归解题思路

- 5)  $B[i][k] = \sum B[i-1][M]_{(\text{DOWN})} + \sum B[i-1][N]_{(\text{UP})}$

$$M = k \dots i-1, N = 1 \dots k-1$$

还是没法直接推。于是把B再分类细化：

$$B[i][k] = C[i][k][\text{DOWN}] + C[i][k][\text{UP}]$$

$C[i][k][\text{DOWN}]$  是S(i)中以第k短的木棒打头的DOWN方案数。然后试图对C进行递归

$$C[i][k][\text{UP}] = \sum C[i-1][M][\text{DOWN}]$$

$$M = k \dots i-1$$

$$C[i][k][\text{DOWN}] = \sum C[i-1][N][\text{UP}]$$

$$N = 1 \dots k-1$$

初始条件:  $C[1][1][\text{UP}] = C[1][1][\text{DOWN}] = 1$

n根木棒的总方案数是

$$\text{Sum} \{ C[n][k][\text{DOWN}] + C[n][k][\text{UP}] \}$$
$$k = 1 \dots n;$$

# 动归解题思路

- 经验：当选取的状态，难以进行递推时（分解出的子问题和原问题形式不一样，或不具有无后效性），考虑将状态**增加限制条件后分类细化**，即增加维度，然后在新的状态上尝试递推