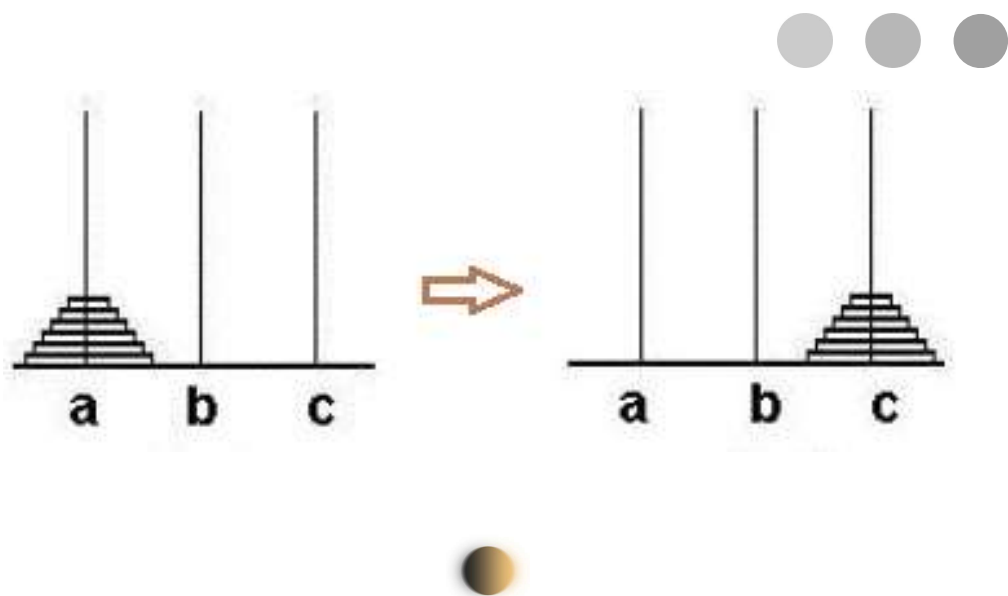


# 算法讲解|递归问题的深入理解——汉诺塔游戏

周行算协 周行算协 2020-11-14 22:53



## 递归到底是什么

大家都已经知道 **递归** 可能就是：  
从前有座山，山里有座庙，庙里有个老和尚在给小和尚讲故事。讲的是什么呢？是从前有座山，山里有座庙，庙里有个老和尚在给小和尚讲故事。讲的是什么呢？是从前有座山，山里有座庙，庙里有个老和尚在给小和尚讲故事。讲的是什么呢.....

## 汉诺塔问题

### 题目描述

设有  $n$  个大小不同的中空圆盘，按从小到大的顺序从 1 到  $n$  编号。将这  $n$  个圆盘任意的迭套在三根立柱上，立柱的编号分别为 A, B, C，这个状态称为初始状态。现在要求找到一种步数最少的移动方案，使得从初始状态转变为目标状态。

移动时有如下要求：

- 一次只能移一个盘；
- 不允许把大盘移到小盘上面。

## 输入格式

第一行一个整数，状态中圆盘总数  $n$ 。

接下来三行每行若干个整数，分别代表初始状态下  $A$ ， $B$ ， $C$  柱子上的圆盘从上到下的编号，如果只有一个数  $0$  就代表这根柱子上没有数。

接下来三行每行若干个整数，分别代表目标状态下  $A$ ， $B$ ， $C$  柱子上的圆盘从上到下的编号，如果只有一个数  $0$  就代表这根柱子上没有数。

## 输出格式

若干行每行一个字符串，格式为 `move I from P to Q`，代表一个移动的操作。

接下来一行一个整数，代表从初始状态到目标状态的最少步数。

## 输入输出样例

### 输出 #1

[复制](#)

```
move 1 from A to B
move 2 from A to C
move 1 from B to C
move 3 from A to B
move 1 from C to B
move 2 from C to A
move 1 from B to C
7
```

### 输入 #1

[复制](#)

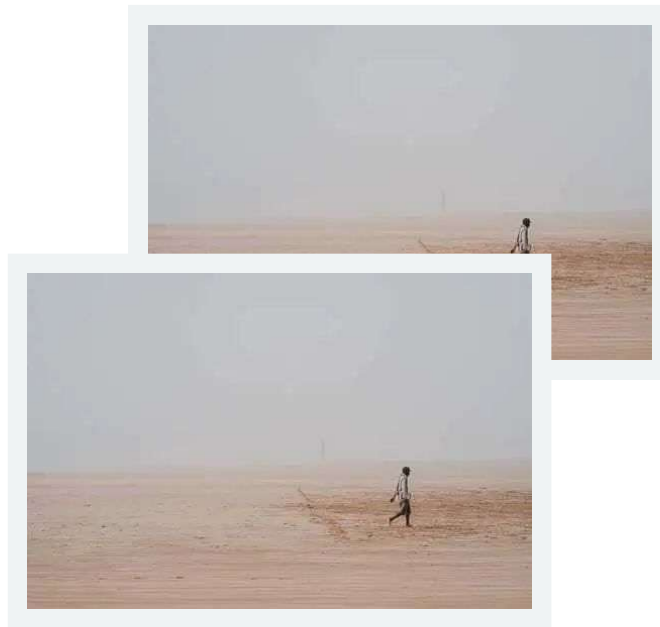
```
5
3 3 2 1
2 5 4
0
1 2
3 5 4 3
1 1
```

## 说明/提示

### 数据规模与约定

对于100% 的数据， $1 \leq n \leq 45$ ， $1 \leq$  每个圆盘的编号  $\leq n$ 。

每行的圆盘描述是从下到上的圆盘编号。



## 问题解析

汉诺塔是大家从小就熟知的趣味游戏，玩多了不难发现有如下规律

1. 当所有塔都排好的时候已经有人说需要移动 $2^n - 1$ 次就可以完成游戏。具体题目有链接  
<https://www.luogu.com.cn/problem/P1760>

(但是这个题要写高精度 普通公式只有60分)

$T(n) = T(n-1) * 2 + 1$  (读者可以自己试着推一下)

2. 后来这个有趣的游戏有个聪明的美国学者发现一种出人意料的简单方法，只要轮流进行两步操作就可以实现。首先把3根柱子按顺序排成"品"字形，把所有的圆盘按从大到小的顺序放在柱子 A上，根据圆盘的数量确定柱子的排放顺序：若n为偶数，按顺时针方向依次摆放 A、B、C;若 n为奇数，按顺时针方向依次摆放 A、C、B。

① 按顺时针方向把圆盘1从现在的柱子移动到下一根柱子，即当n为偶数时，若圆盘1 在柱子A，则把它移动到B;若圆盘1在柱子B，则把它移动到C;若圆盘1在柱子C，则把它移动到A。

② 接着把另外两根柱子上可以移动的圆盘移动到新的柱子上。即把非空柱子上的圆盘移动到空柱子上，当两根柱子都非空时，移动较小的圆盘。这一步没有明确规定移动哪个圆盘，你可能以为会有多种可能性，其实不然，可实施的行动是唯一的。

③ 反复进行①、②操作，最后就能按规定完成汉诺塔的移动。

所以，方法非常简单，就是按照移动规则向一个方向移动圆的金片。例如3阶汉诺塔的移动为：A→C，A→B，C→B，A→C，B→A，B→C，A→C。

**汉诺塔问题也是程序设计中的经典递归问题，下面给出递归实现源代码。**

基础版 代码实现 输入塔数 输出方法

```
1  #include<stdio.h>
2
3  void Move(int n,char x,char y) {
4      printf("move %d from %c to %c\n",n,x,y);
5  }
6
7  void Han(int n,char a,char b,char c) {
8      if(n==1) Move(1,a,c);
9      else {
10         Han(n-1,a,c,b);
11         Move(n,a,c);
12         Han(n-1,b,a,c);
13     }
14 }
15
16 int main() {
17     int n;
18     scanf("%d",&n);
19     Han(n,'a','b','c');
20     return 0;
21 }
```

参考 《ACM程序设计竞赛基础教程》

— THE END —

贴纸：秀米，可商用

文案 朱凌晓 排版 李怡凡

阅读 212



周行算协

分享 收藏 1 4