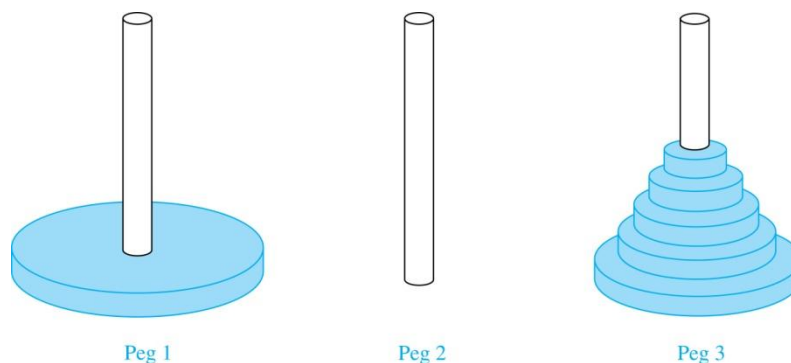


□解:

- 令 H_n 表示解 n 个盘子的汉诺塔所需移动的次数. 建立一个关于序列 $\{H_n\}$ 的递推关系. 最初, n 个盘子在柱子1. 按照游戏规则, 我们可以用 H_{n-1} 次移动将上边的 $n-1$ 个盘子移动到柱子3 (在这些移动中保证最大的盘子不动). 如下图所示:



- 然后我们就可以用一次移动将最大的盘子从柱子1移动到柱子2, 再使用 H_{n-1} 次移动将柱子3上的那 $n-1$ 个盘子移到柱子2, 把他们放在最大的盘子上面.
- 这样就能满足游戏的要求, 因此 $H_n = 2H_{n-1} + 1$. 其中初始条件 $H_1 = 1$ (因为依照游戏规则一个盘子可以用1次移动从柱子1到柱子2).

□ 扩展: 我们可以迭代方法求解这个递推关系:

$$\begin{aligned}H_n &= 2H_{n-1} + 1 \\&= 2(2H_{n-2} + 1) + 1 = 2^2H_{n-2} + 2 + 1 \\&= 2^2(2H_{n-3} + 1) + 2 + 1 = 2^3H_{n-3} + 2^2 + 2 + 1 \\&\dots\dots \\&= 2^{n-1} + 2^{n-2} + 2^{n-3} + \dots + 2 + 1 \text{ (因为 } H_1 = 1 \text{)} \\&= 2^n - 1\end{aligned}$$

【基础知识: 等比序列的求和公式 $\sum_{j=0}^n ar^j = \frac{ar^{n+1}-a}{r-1}, r \neq 1$ 】

□例:对于不含2个连续0的 n 位二进制位串的个数, 找出递推关系和初始条件, 有多少个这样的5位二进制位串?

□解:

➤设 a_n 表示不含2个连续0的 n 位二进制位串数. 为得到一个关于 $\{a_n\}$ 的递推关系, 由求和法则, 不含2个连续0的 n 位二进制位串数=以0结尾的这种二进制位串数+以1结尾的这种二进制位串数. 我们假定 $n \geq 3$, 二进制位串数至少为3位.

- 1)不含2个连续0并以1结尾的 n 位二进制位串, 就是在不含2个连续0的 $n-1$ 位二进制位串的尾部加上一个1. 因此存在 a_{n-1} 个这样的位串.
- 2)不含2个连续0并以0结尾的 n 位二进制位串在它的 $n-1$ 位必须是1, 否则就不满足要求(不含2个连续0). 精确地说, 它就是在不含2个连续0的 $n-2$ 为二进制位串的尾部加上10. 因此存在 a_{n-2} 个这样的位串. 如下所示:

□解(续):

Number of bit strings of length n with no two consecutive 0s:		
End with a 1:	<div>Any bit string of length $n-1$ with no two consecutive 0s</div>	a_{n-1}
End with a 0:	<div>Any bit string of length $n-2$ with no two consecutive 0s</div>	a_{n-2}
Total:		$a_n = a_{n-1} + a_{n-2}$

- 可以断言对于 $n \geq 3$, $a_n = a_{n-1} + a_{n-2}$.
- 其中初始条件 $a_1 = 2$, 因为1位的二进制位串是0或者1. $a_2 = 3$, 因为2位的二进制位串中满足条件的有01,10,11.
- 使用递推关系可以得到 $a_5 = a_4 + a_3 = (a_3 + a_2) + a_3 = 2 \times (a_2 + a_1) + a_2 = 13$

- 例:一个十进制串作为一个编码字. 如果它包含偶数个0, 则它有效. 比如1200是有效的, 120则不是有效的. 设 a_n 是有效的 n 位编码字的个数. 找出一个关于 a_n 的递推关系.
- 解:从少一位构成 n 位有效编码字有两种方式,
- 一个 $n-1$ 位的有效编码字后面加一个非0的数字. 后面的非0数字共有9种(1-9), 所以 $a_{n-1} * 9$;
 - 一个 $n-1$ 位的无效编码字后面加上一个0. 有 10^{n-1} 个 $n-1$ 位编码字, 这其中 a_{n-1} 个有效编码字, 所以 $10^{n-1} - a_{n-1}$ 种.
 - 综上, $a_n = a_{n-1} * 9 + 10^{n-1} - a_{n-1} = 8a_{n-1} + 10^{n-1}$, $a_1 = 9$, 因为串0是无效的, 数字1-9都是有效的.

□例:求关于 C_n 的递推关系,它是通过对 $n+1$ 个数 $x_0 \cdot x_1 \cdot x_2 \dots x_n$ 的乘积中加括号来规定乘法的次序的方式数. 例如 $C_3 = 5$, 因为有以下五种方式

$$((x_0 \cdot x_1) \cdot x_2) \cdot x_3 \quad (x_0 \cdot (x_1 \cdot x_2)) \cdot x_3 \quad (x_0 \cdot x_1) \cdot (x_2 \cdot x_3)$$

$$x_0 \cdot ((x_1 \cdot x_2) \cdot x_3) \quad x_0 \cdot (x_1 \cdot (x_2 \cdot x_3))$$

□解:

- 无论从哪个位置加入括号, 总有一个 “.” 运算符在括号的外面. 该运算符出现在 $n+1$ 个数的两个数之间, 例如 x_k 和 x_{k+1} 之间. 那么在 $x_0 \cdot x_1 \cdot x_2 \dots x_k$ 的乘积中有 C_k 种方式确定它们乘积中的括号不同而出现的不同乘法次序. 另外在 $x_{k+1} \cdot x_{k+2} \cdot x_{k+3} \dots x_n$ 的乘积中有 C_{n-k-1} 种方式确定它们乘积中的括号不同而出现的不同乘法次序.

□解(续):

➤ 由于 k 值可能是0到 $n-1$ 的任何一个值, 所以

$$\begin{aligned}C_n &= C_0 C_{n-1} + C_1 C_{n-2} + \dots + C_{n-2} C_1 + C_{n-1} C_0 \\&= \sum_{k=0}^{n-1} C_k C_{n-k-1}\end{aligned}$$

➤ 其中初始条件 $C_0=1, C_1=1$.

➤ 备注: 序列 $\{C_n\}$ 是**卡特兰数**的序列. 这个序列还是除了该例之外的许多不同计数问题的解.