

河內塔問題 (Tower of Hanoi problem) , 由法國數學家 Édouard Lucas 於 1883 年提出，當年並被製成玩具來販售；他在描述這個問題的時候編了一個小故事：在古老的印度，有一座神廟，據說它是宇宙的中心。在廟宇中放置了一塊上面插有三根長木釘的木板，在其中的一根木釘上，從上至下被放置了 64 片直徑由小至大的圓環形金屬片。古印度教的天神指示祂的僧侶們將 64 片的金屬片移至三根木釘中的其中一根上。規定在每次的移動中，只能搬移一片金屬片，並且在過程中必須保持金屬片由上至下是直徑由小至大的次序，也就是說不論在那一根木釘上，圓環形的金屬片都是直徑較小的被放在上層。

假設現在開始的時候 n 個套還都套在木樁 A 上。我們希望是否能找到一個解答，將 A 樁上的盤子藉著 B 木樁當中間橋樑，全部移到 C 木樁上的最少次數。

搬動時必須遵守上述的規則，需要搬動的次數是 a_n ，顯然 $a_1=1$ (次)。當 $n=2$ 時，總共搬動 $1+1+1$ (次) = $2 \times 1 + 1$ (次)，完成搬動 2 層圓盤需要 a_2 ， $a_2=2 \times 1 + 1 = 2 \times a_1 + 1 = 3$ (次)。當 $n=3$ 時，總共搬動 $3+1+3$ (次) = $2 \times 3 + 1$ (次)，完成搬動 3 層圓盤需要 a_3 ， $a_3=2 \times 3 + 1 = 2 \times a_2 + 1 = 7$ (次)。利用遞迴關係式 $a_n=2a_{n-1}+1$ (次)， $a_1=1$ (次)，可推出 $a_n+1=2^2 \times 2^{n-2}$ ，得 $a_n=2^n - 1$ (次)。

所以說起始有 n 個圓盤在同一根柱子 (A)，越上層的圓盤越小。您得將原先第二層到第 n 層的圓盤移到另外第二根柱子 (B)，總共需搬動 $2^{n-1} - 1$ 次。之後，將原先的第一層圓盤 (最大者) 移至第三根空柱 (C) 上；再將柱子 (B) 的 $n-1$ 個圓盤移到柱子 (C) 上，總共需搬動 $2^{n-1} - 1$ 次。完成整個過程需要搬動的次數是 $(2^{n-1} - 1) + 1 + (2^{n-1} - 1) = 2^n - 1$ (次)。

程式碼：

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <time.h>
4
5 void hanoi(int n, char A, char B, char C) {
6     if(n == 1) {
7         printf("Move sheet from %c to %c\n", A, C);
8     }
9     else {
10        hanoi(n-1, A, C, B);
11        hanoi(1, A, B, C);
12        hanoi(n-1, B, A, C);
13    }
14 }
15
16 int main() {
17
18     clock_t start, end;
19     int n;
20     printf("請輸入盤數:");
21     scanf("%d", &n);
22     start = clock();
23     hanoi(n, 'A', 'B', 'C');
24     end = clock();
25     double diff = end-start;
26     printf("execution time : %f sec", diff / CLOCKS_PER_SEC );
27     return 0;
28 }
```

程式運行時間:

execution time : 3.979000 sec

CPU型號:

Intel® Core™ i5-1135G7