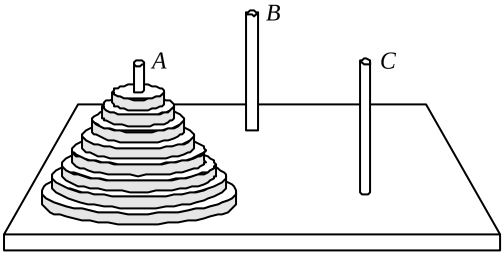
漫談河內塔問題

許介彥

大葉大學　通訊與計算機工程學系

河內塔問題

一平面上豎著 A、B、C 三根木樁，其中的木樁 A 由上而下套著由小而大八個大小相異的圓環，如下圖所示：



假設我們想要將這八個圓環由木樁 A 搬到木樁 C，而且搬動的過程受到以下三項限制：

1. 一次只能搬動一個圓環。
2. 每次搬動都須由某根木樁搬到另一根木樁，圓環不能被暫時放到其他地方。
3. 對任何木樁上任意兩個相疊的圓環而言，上面的圓環一定要比下面的圓環小。

請問：要完成此項工作最少須搬動圓環幾次？

這是有名的河內塔問題（Tower of Hanoi problem），由法國數學家 Édouard Lucas (1842–1891) 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：在印度的某座古老的寺廟前矗立著三根鑲滿了鑽石的柱子，其中之一由上而下套著由小而大 64 個黃金打造的圓盤；根據當地的傳說，當廟裡的和尚在上述限制下將這 64 個圓盤成功地移到另一根柱子的剎那，世界將會毀滅。

一般化的河內塔問題

河內塔問題中圓環的個數為 8，上述故事中圓環的個數為 64；讓我們考慮一個更具一般性的問題：當木樁 A 一開始套著 n 個圓環時，最少須搬動幾次？其中的 n 是任意正整數。

讀者也許根本就懷疑這項工作是否能被完成；經由簡單的試驗，不難發覺至少當 n 不太大時這項工作是可以被完成的。當 n 等於 1 時，很顯然只須搬動一次；當 n 等於 2，不難發現只須搬動三次；當 n 等於 3 時，最少須搬動七次。n 等於 3 的情形已經比 n 等於 1 或 2 的情形難了不少，最好（即次數最少）的搬法已經不是顯而易見；在繼續往下看之前，請讀者利用一點時間確實找出只須七次的搬法（讀者可利用身邊不同大小的硬幣三枚來模擬三個不同尺寸的圓盤）。

利用七次搬動將三個圓環由 A 移到 C 的搬法如下：

1. $A \to C$	5. $B \to A$
2. $A \to B$	6. $B \to C$
3. $C \to B$	7. $A \to C$
4. $A \to C$	

其中，$A \to C$ 是表示將木樁 A 最上面的圓環

由 A 取出，套到木樁 C 上，其餘依此類推。

如果我們定義 a_n 為將 n 個圓環由某根木樁移到另一根木樁所需的最少搬動次數，那麼我們已經知道 $a_1=1$， $a_2=3$， $a_3=7$，我們希望能求出 a_n 的一般式。

請注意我們所定義的 a_n 是移動 n 個圓環所需「最少」的次數；不管 n 是多少，完成此項工作其實都有許多不同的做法。舉例來說，當 $n=1$ 時，最好的搬法是將唯一的圓環由木樁 A 直接移到 C，但這並不是唯一的搬法，另一個搬法是 $A\rightarrow B\rightarrow C\rightarrow A\rightarrow B\rightarrow A\rightarrow C$，共搬了六次。因此，即使簡單如只有一個圓環的情形，這項工作也有無窮多種方法可以完成。

當 $n=3$ 時，前述最好的搬法其實可以看成是分為三個階段。我們注意到最大的圓環在整個過程中只被搬了一次，為了要能將最大的圓環由 A 搬至 C，在搬最大的圓環之前，位於最大的圓環上方的兩個較小的圓環必須先被移到木樁 B 上，這需要三次搬動。而當最大的圓環由 A 被搬到 C 後，較小的兩個圓環接著由 B 被搬到 C，這需要另外三次搬動，因此總共搬了 $3+1+3=7$ 次，如下圖所示：

將三個圓環由 A 搬至 C

$$\rightarrow\begin{cases}\text{步驟 1. 將兩個圓環由}A\text{ 搬至 }B\\ \text{步驟 2. 將一個圓環由}A\text{ 搬至 }C\\ \text{步驟 3. 將兩個圓環由}B\text{ 搬至 }C\end{cases}$$

這樣的看法頗符合遞迴演算法的概念，只要我們知道怎麼將兩個圓環由某根木樁搬到另一根木樁，就有辦法將三個圓環由某根木樁搬到另一根木樁，因此我們可以透過解決搬兩個圓環的問題來解決搬三個圓環的問題（雖然須解決搬兩個圓環的問題兩次）。

知道如何將三個圓環由某根木樁移到另一根木樁後，對解決 $n=4$ 的問題有沒有幫助呢？答案是肯定的，因為我們同樣可以透過解決兩個搬三個圓環的問題來解決搬四個圓環的問題：

將四個圓環由 A 搬至 C

$$\rightarrow\begin{cases}\text{步驟 1. 將三個圓環由}A\text{ 搬至 }B\\ \text{步驟 2. 將一個圓環由}A\text{ 搬至 }C\\ \text{步驟 3. 將三個圓環由}B\text{ 搬至 }C\end{cases}$$

因此，即使 n 再大，我們還是有方法來完成這項工作，因為我們可以透過解決搬 $(n-1)$ 個圓環的問題來解決搬 n 個圓環的問題，而搬 $(n-1)$ 個圓環的問題又可以透過搬 $(n-2)$ 個圓環的問題來解決，而搬 $(n-2)$ 個圓環的問題又可以透過搬 $(n-3)$ 個圓環的問題來解決……，最後原來的問題將被簡化成許多「搬一個圓環」的問題，而搬一個圓環的問題是我們根本不需要思考就知道怎麼解決的！

更具一般性地說，如果我們將三根木樁的名稱以 x、y、z 取代，上述作法是將搬 n 個圓環的問題分為以下三個步驟來解決（假設 n 大於 1）：

將 n 個圓環由 x 搬至 z

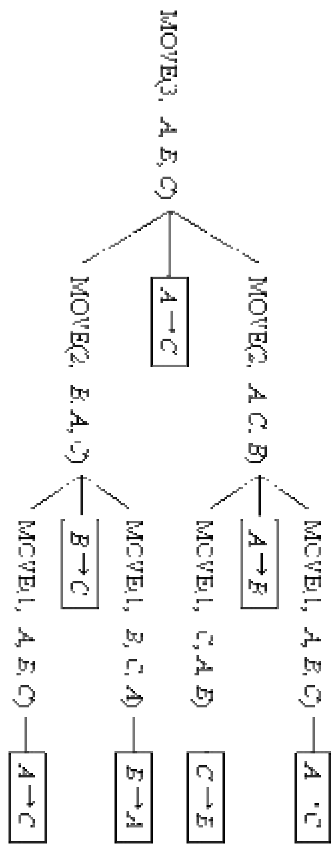
$$\rightarrow\begin{cases}\text{步驟 1. 將 }(n-1)\text{ 個圓環由 }x\text{ 搬至 }y\\ \text{步驟 2. 將一個圓環由 }x\text{ 搬至 }z\\ \text{步驟 3. 將 }(n-1)\text{ 個圓環由 }y\text{ 搬至 }z\end{cases}$$

這樣的做法可以很簡潔地表達成以下的遞迴演算法（假設 n 是任意正整數）：

```
MOVE( n, x, y, z)
  if n = 1
    將一個圓環由 x 搬到 z
  else
    MOVE( n - 1, x, z, y)
    將一個圓環由 x 搬到 z
    MOVE( n - 1, y, x, z)
```

MOVE(n, x, y, z)可用來將 n 個圓環由木棒 x 搬到木棒 z。觀察以上演算法，讀者不難發覺，即使撇開演算法與電腦的關係不談，描述演算法的語法本身其實就是可用來描述一般做事方法的一個很好的表達工具。

如果我們追蹤一下 MOVE(3, A, B, C)的執行，所得即為前述將三個圓環由 A 搬到 C 的最佳搬動次序（見下圖）。



有更好的搬法嗎？

上述做法雖然是解決搬 n 個圓環的問題的一種方式，而且對 $n=3$ 而言是最好的方式，不過到目前為止我們並不知道它是不是對任意正整數 n 而言都是最好的方式。由於步驟 1 與步驟 3 各須將 $(n-1)$ 個圓環由某根木棒搬到另一根木棒，所以至少各須搬動圓環 a_{n-1} 次，而步驟 2 至少須搬動圓環一次，因此上述做法在最好的情形下是一個搬了 $(a_{n-1}+1+a_{n-1})=2a_{n-1}+1$ 次的做法；既然 a_n 是搬動 n 個圓環所需最少的次數，以下關係是成立的：

$$a_n \le 2a_{n-1}+1$$

最好的方法所用的次數一定小於或等於我們的方法所用的次數。

從另一方面來看，不管是用什麼樣的做法將這個問題解決，解決過程中最大的圓環至少都須被搬動一次；由於題目的限制，在搬動最大的圓環之前，最大的圓環上方的其他圓環必須先全被移走，而且最大的圓環要去的木棒上不能有任何圓環，因此在搬動最大的圓環的當時，所有其他比最大的圓環小的 $(n-1)$ 個圓環必定按大小整齊地被擺放在另一根木棒上，所以不管是用什麼方法，在搬動最大的圓環之前至少已經搬動了 a_{n-1} 次，而在將最大的圓環搬到它的最終位置之後至少還須搬動 a_{n-1} 次。因此要解決搬 n 個圓環的問題，不管是用什麼方法，至少都需 $(a_{n-1}+1+a_{n-1})=2a_{n-1}+1$ 次的搬動，即使最好的方法也不例外，因此以下關係是成立的：

$$a_n \ge 2a_{n-1}+1$$

綜合以上兩個式子，a_n 既要小於或等於

$2a_{n-1}+1$ 又要大於或等於 $2a_{n-1}+1$，我們因此確定 a_n 其實就等於 $2a_{n-1}+1$，我們之前的做法其實就是最好的方法。配合已知的起始條件，我們有了數列完整的遞迴定義：

$$a_n=\begin{cases}1 & n=1\\ 2a_{n-1}+1 & n>1\end{cases}$$

由此不難推導出

$$a_n=2^n-1,\quad \forall n\ge 1$$

這就是我們希望求得的一般式。

附帶一提，當 n 等於 64 時，a_{64} 的值為 $2^{64}-1$ = 18,446,744,073,709,551,615，這個數有多大呢？這麼說吧，即使印度老廟中的和尚可以不眠不休，一年 365 天，一天 24 小時不停地搬動，而且身手俐落，搬動一個圓環只需一秒鐘，要完成這項搬 64 個圓環的工作也要大概 5.845×10^{11} 年，這個數字大約是我們目前所知宇宙的年齡（約 1.5×10^{10} 年）的四十倍。

有趣的模式

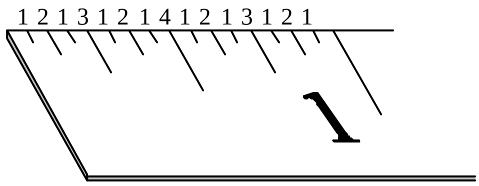
假設我們將 n 個圓環由小而大依序編號，由 1 編至 n。當 $n=3$ 時，至少須搬動七次，如果我們依搬動先後將圓環的編號列出，將得到 1213121 的順序；當 $n=4$ 時則是 121312141213121，這樣的數字模式我們平常如果稍加留意，其實在許多場合都看得到。

舉例來說，如果我們將所有小於 16 的正整數表示成二進位的形式並依序列出：

	4	3	2	1	
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	1
4	0	1	0	0	3
5	0	1	0	1	1
6	0	1	1	0	2
7	0	1	1	1	1
8	1	0	0	0	4
9	1	0	0	1	1
10	1	0	1	0	2
11	1	0	1	1	1
12	1	1	0	0	3
13	1	1	0	1	1
14	1	1	1	0	2
15	1	1	1	1	1

由上表很明顯可以看出，如果我們由 0000 至 1111 將每個數由最右邊的位元往左看所看到的第一個 1 的位置記錄下來，結果將與上述河內塔問題在 $n=4$ 時的搬動次序相同。

上表中，如果只看陰影部分的輪廓，又很容易讓我們聯想到一般英制的直尺上標示刻度的方式。如果我們將刻度線由最短到最長依序編號，然後將尺上的刻度線由左而右依序列出，相同的模式又再次出現了：

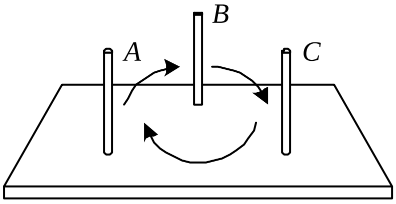


河內塔問題的變形

一般化的河內塔問題可以衍生出幾種有趣的變形，例如：

1. 原來的問題多加上一項限制：每個圓環的搬動不能是 $A\to C$ 或 $C\to A$，也就是說，搬動只能發生於 A 與 B 之間或是 B 與 C 之間。請問：最少須搬幾次才能將圓環全部由 A 搬到 C？
2. 原來的問題多加上一項限制：每個圓環的搬動必須是朝順時鐘方向搬，也就是說，

每次搬動只能是 $A \to B$、$B \to C$、$C \to A$ 等三種情形之一：

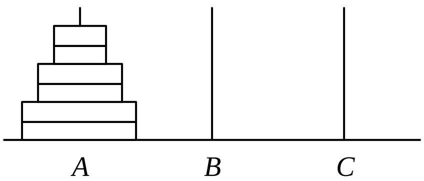


(a) 最少須搬幾次才能將圓環全部由 A 搬到 C？

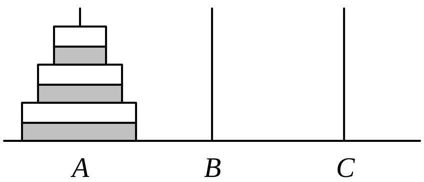
(b) 最少須搬幾次才能將圓環全部由 A 搬到 B？

3. 圓環總共有 $2n$ 個（而不是只有 n 個），不過只有 n 種尺寸，每種尺寸的圓環各有兩個。

(a) 假設每種尺寸的兩個圓環一模一樣，最少須搬幾次才能將它們全部由 A 搬到 C？下圖為 $n=3$ 開始的情形：

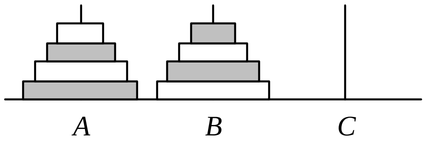


(b) 假設每種尺寸的兩個圓環分別具有黑和白兩個顏色，一開始時位於上方的圓環都是白色而下方的都是黑色，以 $n=3$ 為例：



如果我們除了要將它們全部由 A 搬到 C 之外，還希望最後的結果對各個尺寸的圓環而言都維持著原來上白下黑的關係，請問最少須搬幾次？

(c) 假設每種尺寸的兩個圓環分別具有黑和白兩個顏色，而且一開始時木樁 A 及木樁 B 上分別擺放著 n 個黑白相間的圓環，木樁 A 及木樁 B 上最大的圓環分別為黑色及白色。以 $n=4$ 為例：



請問：要將所有的黑環移到 B 且所有的白環移到 A，最少須搬幾次？

4. 如果圓環有 n 個但是木樁有四根（或是更多根），最少須搬幾次？當木樁有四根時，此問題通常稱為 Reve's puzzle。

以上這些變形問題的搬動過程也都可以利用遞迴的方式來思考。以第一種變形為例，由題目的限制可知最大的圓環至少須移動兩次；在最好的搬法中，最大的圓環正是被搬了兩次。假設 b_n 是在題目所述限制下將 n 個圓環由 A 搬到 C 所需的最少搬動次數，在最好的搬法中，最大的圓環先由 A 被搬到 B，再由 B 被搬到 C；在將它由 A 搬到 B 之前，所有較小的 $(n-1)$ 個圓環必須先由 A 被搬到 C（須 b_{n-1} 次搬動），接下來將最大的圓環由 B 搬到 C 之前，所有較小的 $(n-1)$ 個圓環必須由 C 被搬到 A（須另外 b_{n-1} 次搬動）；最大的圓環到達 C 後，所有較小的 $(n-1)$ 個圓環又必須由 A 被搬到 C（須另外 b_{n-1} 次搬動），因此當 $n>1$，

$$b_n = b_{n-1} + 1 + b_{n-1} + 1 + b_{n-1} = 3b_{n-1} + 2$$

再配合 $n=1$ 時的起始條件，我們有了完整的

遞迴定義：

$$b_n = \begin{cases} 2 & n = 1 \\ 3b_{n-1} + 2 & n > 1 \end{cases}$$

由此很容易可以推導出

$$b_n = 3^n - 1, \quad n \geq 1 .$$

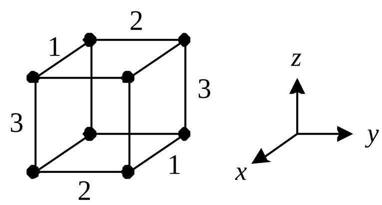
讀者也許有興趣驗證一下用這個搬法搬完 $(3^n - 1)/2$ **次時，所有的** n **個圓環其實正好由木樁** A **被移到了木樁** B **上，正好是這些圓環往目的地旅途的「中點」。**

上述各種變形問題中以木樁數大於三的問題最為困難。

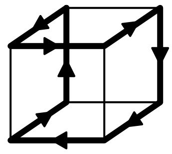
結語

河內塔問題幾乎出現於所有與離散數學、資料結構、演算法有關的書籍中，可以用來練習數學歸納法的証明、遞迴程式的撰寫、遞迴關係的求解等，對學習電腦理論的人來說是必然會接觸到的問題。

除了直尺上的刻度及二進位數中 1 **的位置之外，相當有趣的是，河內塔問題搬動圓環的最佳次序還和某些圖形**（graph）**中由任意一個頂點開始以一筆劃循著圖中的線條經過每個頂點一次並回到起點的走法有著密切的關係，也就是可以對應到這些圖形在述語上所謂的** Hamiltonian cycle。**以** n **等於** 3 **為例，如果我們將一個正立方體的十二個邊依不同的方向分為三群，將與** x **軸、**y **軸、**z **軸平行的邊（各有四條）分別命名為** 1、2、3，**如下圖所示：**



那麼，以正立方體的任意一個頂點為起點，循著 1213121 **的次序走完七個邊後再連回起點所得的路徑將會是正立方體的邊所形成的圖形的一條** Hamiltonian cycle，**如下圖所示：**



不只 $n = 3$ **如此，只要** n **是一個正整數，河內塔問題搬動圓環的最佳次序都正好對應到某類圖形的** Hamiltonian cycle。

表面上看起來不相干的兩個東西背後竟然隱藏著密切的關連，這樣的發現常能為科學家帶來莫大的樂趣。

參考資料

1. **許介彥**(2001)，**遞迴函數的求解技巧，科學教育月刊，第** 238 **期。**
2. **許介彥**(2001)，**遞迴演算法簡介，科學教育月刊，第** 245 **期。**
3. Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, 4th edition, McGraw-Hill, 1999.

作者信箱：chsu@mail.dyu.edu.tw