河內塔~一座移動完就會世界末日的塔

典故:

從前從前, 在越南有個地方叫做河內，河內裡有座山，山上有座塔，塔中有三根聳天而立的大銀棒, 銀棒上串有64 個金盤。塔中的僧侶們會依照一個古老的預言, 並依規則來移動這些金盤;而預言則說當這些金盤全部都移動完畢之後，世界就會毀滅。這個傳說就叫做梵天寺之塔問題 (Tower of Brahma puzzle), 也就是我們熟知的河內塔 (Tower of Hanoi) 問題, 在這個問題中 64個金盤的它每個的大小都是不一樣的, 並在一開始的時候會依序從底部最大排到頂部最小, 而搬移的規則有三個:

- 一次只能搬移一個金盤
- 盤子只能在三根柱子中被搬動 (你不能拿起來放在旁邊, 必須挑一根柱子放下。)
- 尺寸較小的盤子需永遠保持在最上方

這個傳說來自數學家-Anatole Lucas, 是用公式證明此一數學問題的數學家, 其證得河內塔的最佳步驟則是2^N - 1 次, 其中的N為金盤的數量, 若我們要解一個三層的河內塔則需動 2^3 - 1 = 7次;那傳說中的64層就會需要 2^64 - 1次，即便僧侶們各個單身三十年，手速驚為天人的一秒移一盤, 也會需要超過5849億年才有辦法完成。

Recursion-遞迴**:**

將河內塔的三根柱子分為A、B、C，從最簡單的開始依此類推：

一個盤:A→C;

兩個盤:A→B, A→C, B→C;

三個盤:A->C , A->B , C->B , A->C , B->A , B->C , A->C

一個盤的時候, 直接由A柱到C柱

兩個盤的時候, 把不是最大的一個放到B柱, 然後把最大的從A放到C柱, 再把B柱上的小盤放到C柱

三個盤的時候, 步驟相同其實, 就是把不是最大的放到B柱然後把最大的放到C柱, 再借助A柱重複做一次兩個盤的情況的步驟(其實就是重複兩次)。

我們要解決n層河內塔就要解決n-1層河內塔，

我們要解決n-1層河內塔就要解決n-2層河內塔，

……

這就是河內塔與recursion的關係。

用**recursion**解決此問題**:**

以較有數學邏輯的方法思考的話, 我們可以假設有n盤時, 至少須 T(n) 次的移動來完成, 那麼, 我們再加丨盤，即此時共有 n+1 盤；我們知道前 n 盤花了 T(n) 次來移動至另一根柱上, 第 n+1 盤只須花一次就可移至指定的柱上, 所以, 只須再花 T(n) 次的移動將 n 盤移至這一片盤之上, 這就完成了任務──有 n+1 盤移到另一柱上。它們都是在規範內被完成移動, 所以最少的總移動次數:

T(n+1) = T(n) + 1 +T(n)

= 2T(n) + 1

則 T(1) = 1

T(2) = 3 = 2 + 1

T(3) = 2T(2) + 1 = 2(2+1) = 2^2 + 2 + 1

..……

T(n) = 2^(n-1) + 2^(n-2) + ……. + 2 + 1

因(a^n - 1) = (a - 1)(a^(n-1) + a^(n-2) + ... + a + 1)

所以

T(n+1) = 2T(n) + 1

= 2 (2^(n-1) + 2^(n-2) + ……. + 2 + 1) + 1

= 2^(n-1) + 2^(n-2) + ……. + 2 + 1

= (2^(n+1) - 1)/ 2 - 1

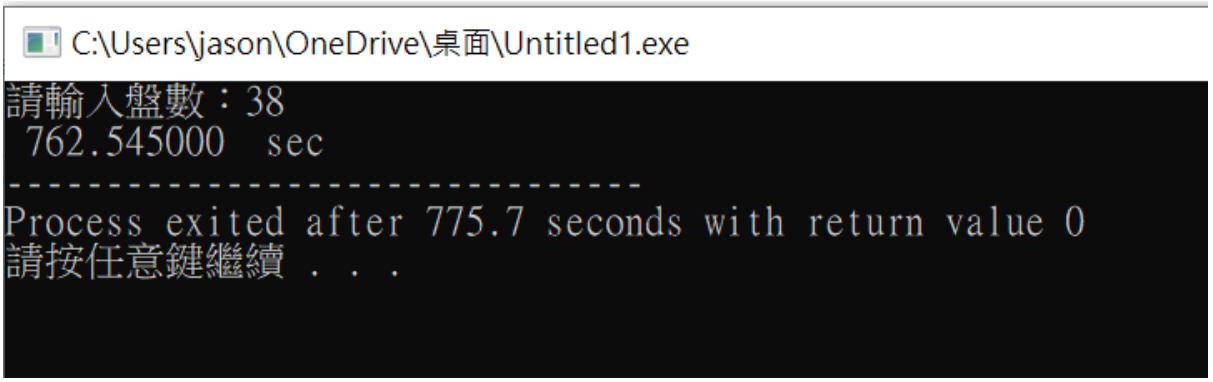
= 2^(n+1) - 1

可得

T(n) = 2^n - 1

程式碼:

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

void hanoi(int n, char A, char B, char C) {
    if(n == 1) {
    }
    else {
        hanoi(n-1, A, C, B);
        hanoi(1, A, B, C);
        hanoi(n-1, B, A, C);
    }
}

int main() {

clock_t start, end;

    int n;
    printf("請輸入盤數：");
    scanf("%d", &n);

    start = clock();

    hanoi(n, 'A', 'B', 'C');

end = clock();
double diff = end-start;
    printf(" %f  sec", diff / CLOCKS_PER_SEC );

return 0;
}
```



CPU : i7-8565U

我的文書機跑超慢又容易過熱(= =), 跑 40 盤的時候我的CPU差點過世，所以我最後決定紀錄38盤的秒數

,