

1. 介紹 Tower of Hanoi (河內塔) 問題

河內塔傳說是由 A、B、C 三個塔與 N 個圓盤組成的一個遞迴邏輯，假設 A 塔中具有 N 個圓盤從下而上、由大到小排列，在每次只能移動一個圓盤，並且大圓盤不能置於小圓盤之上的情況下，完成圓盤成功轉移到另一根柱子的任務；這個問題的解法有許多種，但最快速且簡短的做法是 $2^n - 1$ 次(至少目前找不到更為快速的解法，應該也沒更快的次數了)，其作法是 A 塔有 N 塊盤，目標是把這些盤全部移到 C 塔。那麼先把 A 塔頂部的 N-1 塊盤移動到 B 塔，再把 A 塔剩下的大盤移到 C，最後把 B 塔的 N-1 塊盤移到 C。如此遞迴地使用下去，就可以求解。

2. Tower of Hanoi (C code)

由第一點可知，我們需要使用 recursion 去重複呼叫 hanoi function，從第 N 層直到最後 1 層，因此程式如下圖所示。

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  static int move = 0;
6  static char Time1[32], Time2[32];
7
8  void hanoi(int n, char A, char B, char C) {
9      if(n == 1) {
10         move++;
11     }
12     else {
13         hanoi(n-1, A, C, B);
14         hanoi(1, A, B, C);
15         hanoi(n-1, B, A, C);
16     }
17 }
18
19 int main() {
20     clock_t timeset1, timeset2;
21     int n;
22     printf("請輸入盤數: ");
23     scanf("%d", &n);
24     timeset1 = clock();
25     hanoi(n, 'A', 'B', 'C');
26     timeset2 = clock();
27     double tot = timeset2 - timeset1;
28     printf(" %f seconds", tot);
29     return 0;
30 }
```

由上圖可知，在第十三行到第十五行用了三行的 recursion，完成圓盤依照規則移動的動作。

3. CPU 規格與最多能跑到的層數

CPU : Intel i5-1125G7

CJSCOPE 喜傑獅® 2020、2021 Series

裝置名稱	DESKTOP-F9RI83B
處理器	11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz
已安裝記憶體(RAM)	16.0 GB (15.8 GB 可用)
裝置識別碼	612B2B48-B194-4071-A051-DCCA4D1A7165
產品識別碼	00325-96744-97998-AAOEM
系統類型	64 位元作業系統，x64 型處理器
手寫筆與觸控	此顯示器不提供手寫筆或觸控式輸入功能

複製

最多能跑到的層數：40 層

C:\Users\emily\OneDrive\桌面\計概作業.exe

```
請輸入盤數：40
3429358.000000 seconds
-----
Process exited after 3435 seconds with return value 0
請按任意鍵繼續 . . .
```