

## HW3 河內塔問題

B1029016 黃靖庭

河內塔問題，指有三根杆子，其中一根放入若干個(不為零)穿孔圓盤，在遵守大盤不能放在小盤上、一次只能移動一個盤子等規則，把原本在第一根杆子的圓盤以相同的擺放方式放到第三根杆子並且計算如何擺放才是最有效率並且計算移動的次數。

最早發明這個問題的人是法國數學家愛德華·盧卡斯。傳說越南河內某間寺院有三根銀棒，上串 64 個金盤。寺院裡的僧侶依照一個古老的預言，以上述規則移動這些盤子；預言說當這些盤子移動完畢，世界就會滅亡。這個傳說叫做梵天寺之塔問題（Tower of Brahma puzzle）。但不知道是盧卡斯自創的這個傳說，還是他受他人啟發。

經過計算此傳說最少要移動 $2^{64}-1$ 次才能成功移完，而這數字大到就算一秒移動一次，也要5849.42億年才有可能移動完，而依照大爆炸理論，地球至今的年齡也才差不多137億年而已。

河內塔問題其實與數學的遞迴關係匪淺，遞迴數列中，就是利用不斷重複的特性，達到在有限條件內，讓原本很龐大複雜的難以計算的問題化簡成重複可計算的數學式子。

河內塔問題的規則提到，大盤不能放在小盤上面，使得在解決問題時要一直重複不斷地，把小環拿出放到暫存的杆子上，這種來回的擺放方式因為過程太過於繁瑣，索性直接寫成程式讓電腦去計算，這時程式的設計就變得很重要。

```
#include <stdio.h>
```

```
#include <time.h>
```

```
#include <intrin.h>
```

```
void getCpuInfo()
```

```
{  
  
    int cpulInfo[4] = {-1};  
  
    char cpu_manufacture[32] = { 0 };  
  
    char cpu_type[32] = { 0 };  
  
    char cpu_freq[32] = { 0 };  
  
    __cpuid(cpulInfo, 0x80000002);  
  
    memcpy(cpu_manufacture, cpulInfo, sizeof(cpulInfo));  
  
  
  
    __cpuid(cpulInfo, 0x80000003);  
  
    memcpy(cpu_type, cpulInfo, sizeof(cpulInfo));  
  
  
  
    __cpuid(cpulInfo, 0x80000004);  
  
    memcpy(cpu_freq, cpulInfo, sizeof(cpulInfo));  
  
  
  
    printf("\nCPU manufacture: %s",cpu_manufacture);  
  
    printf("CPU type: %s",cpu_type);  
  
    printf("CPU main frequency: %s\n",cpu_freq);  
  
}  
  
void hanoi(int Height, char A, char B, char C) {  
  
    if(Height == 1) {  
  
        printf("Move sheet from %c to %c\n", A, C);  
  
    }  
}
```

```
else {  
    hanoi(Height-1, A, C, B);  
  
    hanoi(1, A, B, C);  
  
    hanoi(Height-1, B, A, C);  
}  
}  
  
int main()  
{  
    int Height;  
  
    clock_t start,end;  
  
    start=clock();  
  
    scanf("%d",&Height);  
  
    hanoi(Height,'A','B','C');  
  
    getCpuInfo();  
  
    end=clock();  
  
    double diff = end - start;  
  
    printf(" using %f sec", diff / CLOCKS_PER_SEC );  
  
    return 0;  
}
```

此程式包含了顯示程式執行時間及河內塔問題的計算程式。

資料來源 <https://zh.wikipedia.org/wiki/%E6%B1%89%E8%AF%BA%E5%A1%94>