

計概hw3 B1029029林郁庭

假設解n個圓盤的河內塔，執行次數為 $T(n)$ ，一個圓盤用一步就結束了，所以 $T(n)=1$ ， n 個圓盤時，首先要將上面的 $n-1$ 個圓盤從A移動到B，為 $T(n-1)$ 步，把最大的圓盤移動到C是一步，把位在B的 $n-1$ 個圓盤移動到C是 $T(n-1)$ 步，所以 $T(n)=2T(n-1)+1$ ，可得 $T(n)=(2^{**n})-1$ ，執行次數比這個還少的解法並不存在

如取 $N=64$ ，最少需移動 $2^{64}-1$ 次。即如果一秒鐘能移動一塊圓盤，仍將需5849.42億年。目前按照[宇宙大爆炸理論](#)的推測，宇宙的年齡僅為 137 億年。

在真實玩具中，一般 $N=8$ ；最少需移動 255 次。如果 $N=10$ ，最少需移動 1023 次。如果 $N=15$ ，最少需移動 32767 次；這就是說，如果一個人從 3 歲到 99 歲，每天移動一塊圓盤，他最多僅能移動 15 塊。如果 $N=20$ ，最少需移動 1048575 次，即超過了一百萬次。

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int moved=0;

void hanoi(int n, char A, char B, char C) {
    if(n == 1) {
        printf("Move sheet from %c to %c\n", A, C);
        moved++;
    }
    else {
        hanoi(n-1, A, C, B);
        hanoi(1, A, B, C);
        hanoi(n-1, B, A, C);
    }
}

int main() {
```

```
int n;
clock_t begin,end;
double cost;
begin =clock();
printf("Please key-in number of sheet : ");
scanf("%d", &n);

hanoi(n, 'A', 'B', 'C');
printf("\nTotal moved %d steps\n\n",moved);
end=clock();
cost = (double)(end - begin)/CLOCKS_PER_SEC;
printf("constant CLOCKS_PER_SEC is: %ld, time coast is %lf
secs",CLOCKS_PER_SEC,cost);

return 0;
}
```