

# Tower of Hanoi

## a. 題目分析與演算法

解這個題目時是使用 Recursion 來解決，假設有  $n$  個碟子（或盤子）放在最左邊（輸出表示成 1），要將他們一個一個放到最右邊的柱子（輸出表示成 3），最短的路徑是先把  $n-1$  個碟子放在中間的柱子（輸出表示成 2），接著把最大的碟子移到最右邊的柱子，再把  $n-1$  個碟子放到最右邊。

在過程中，用遞迴的方式把  $n$  一直拆，拆成 1，假如  $n$  等於 2，拆到  $n$  等於 1 時，起點為 1，最小的碟子會移到 2，然後下一個碟子會移到 3，最後把第一個碟子從 2 移到 3。

在  $n$  等於 2 的例子裡，在起始柱（程式碼的 `sta`）的第一個碟子的目標柱（程式碼的 `goal`）改變為暫存柱（程式碼的 `tmp`），再把最下面的碟子搬到目標柱後， $n-1$  個碟子所在的暫存柱變為起始柱，原本的起始柱則變成暫存柱，最後把  $n-1$  個碟子移到目標柱。



（圖片取自動物朋友的 komica wiki）

## b. 討論

如果把 1 到  $n$  層碟子改成尺寸 1 有 1 個碟子，尺寸 2 有 2 個碟子，以此類推，尺寸  $n$  有  $n$  個碟子，那又該如何移動，要移動多少次呢。

以下是題外話。

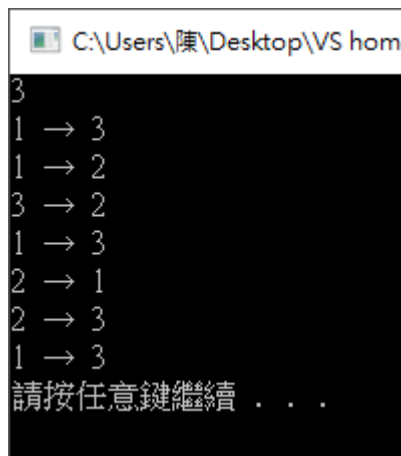
在寫報告前，我查過河內塔的由來，傳說是某間寺廟有三根銀棒，上面共有 64 個金盤，寺廟裡的僧侶依照一個預言來移動這些盤子；預言說盤子移動完

後世界就會滅亡，所以這個程式其實是在告訴僧侶如何用最短的路徑毀滅世界。

順帶一提，僧侶們至少要移動 2 的 64 次方減 1 次來毀滅世界。

### c.執行畫面

測試輸入的值是課本上舉例的 n 等於 3。



```
C:\Users\陳\Desktop\VS hom
3
1 → 3
1 → 2
3 → 2
1 → 3
2 → 1
2 → 3
1 → 3
請按任意鍵繼續 . . .
```

### d.程式碼

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

```
# define A 1
# define B 2
# define C 3
```

```
void hanoi(int x, int sta, int tmp, int goal);
```

```
int main(void){
```

```
    int n;
```

```
    scanf("%d", &n);
```

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

```
        system("pause");

    }

void hanoi(int x, int sta, int tmp, int goal){

    if (x == 1){

        printf("%d → %d\n", sta, goal);
    }

    else{

        hanoi(x - 1, sta, goal, tmp);

        printf("%d → %d\n", sta, goal);

        hanoi(x - 1, tmp, sta, goal);
    }

}
```

以下是程式碼截圖。

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

#define A 1
#define B 2
#define C 3

void hanoi(int x, int sta, int tmp, int goal);

int main(void){
    int n;

    scanf("%d", &n);

    hanoi(n, A, B, C);

    system("pause");
}
```

```
void hanoi(int x, int sta, int tmp, int goal){
    if (x == 1){
        printf("%d → %d\n", sta, goal);
    }
    else{
        hanoi(x - 1, sta, goal, tmp);
        printf("%d → %d\n", sta, goal);
        hanoi(x - 1, tmp, sta, goal);
    }
}
```