

Tower of Hanoi recursive (J)

ฟังก์ชันส่งย้ายแผ่น

```
4 void move(char a,char b)
5 {
6     cout<<"move "<<a<<" to "<<b<<endl;
7 }
8
```

Base case : มีแผ่น 1 แผ่นในเสาเริ่ม ให้ย้ายไปยังเสาเป้าหมายเลย

```
8 //function hanoi n = how many disk , s = start tower, c = temp tower,t = destination tower
9 void hanoi(int n,char s,char c,char t)
10 {
11     // base case when have 1 disk move form start to destination
12     if(n == 1)
13     {
14         move(s,t);
15         return;
16     }
17 }
```

Case มี 2 แผ่นในเสาเริ่มต้องย้ายแผ่นแรกไปยังเสาพักก่อน จึงทำการเพิ่มโค้ดการย้ายไปยังเสาพัก

```
9 //function hanoi n = how many disk , s = start tower, c = temp tower,t = destination tower
10 void hanoi(int n,char s,char c,char t)
11 {
12     // base case when have 1 disk move form start to destination
13     if(n == 1)
14     {
15         move(s,t);
16         return;
17     }
18     //move disk form start tower to temp tower using destination tower as temp tower
19     hanoi(n-1,s,t,c);
20 }
```

หลังจากนั้นจึงค่อยย้ายแผ่นที่สองไปยังเสาเป้าหมาย

```
8 //function hanoi n = how many disk , s = start tower, c = temp tower,t = destination tower
9 void hanoi(int n,char s,char c,char t)
10 {
11     // base case when have 1 disk move form start to destination
12     if(n == 1)
13     {
14         move(s,t);
15         return;
16     }
17     //move disk form start tower to temp tower using destination tower as temp tower
18     hanoi(n-1,s,t,c);
19     move(s,t);
20 }
```

พอย้ายแผ่นที่สองไปยังเป้าหมายแล้ว เราก็ต้องนำแผ่นแรกจากเสาพักไปไว้ที่เสาเป้าหมาย เป็นอันเสร็จ

```
9 //function hanoi n = how many disk , s = start tower, c = temp tower,t = destination tower
10 void hanoi(int n,char s,char c,char t)
11 {
12     // base case when have 1 disk move form start to destination
13     if(n == 1)
14     {
15         move(s,t);
16         return;
17     }
18     //move disk form start tower to temp tower using destination tower as temp tower
19     hanoi(n-1,s,t,c);
20     move(s,t);
21     //move disk form temp tower to destination tower using start tower as temp tower
22     hanoi(n-1,c,s,t);
23 }
```

สรุป หากมีดิสเหลือดิสเดียวในเสานั้นสามารถย้ายดิสบนสุดได้เลย แต่ถ้าหากมีสองดิสหรือมากกว่า เราจำเป็นต้องย้ายดิสที่อยู่รองลงมาเป็นลำดับที่สองก่อนเพื่อไม่ให้ดิสที่ใหญ่กว่ามาทับดิสที่เล็กกว่า โดยในกาย้ายนั้นเราจำเป็นต้องย้ายดิสบนสุดไปยังเสาพักแล้วจึงค่อยย้ายดิสที่สองไปยังเสาเป้าหมาย ทำซ้ำเช่นนี้ไปเรื่อยๆ ก็จะสามารถย้ายทั้งหมดไปยังเสาเป้าหมายได้

```
void move(char a,char b)
{
    cout<<"move "<<a<<" to "<<b<<endl;
}
//function hanoi n = how many disk , s = start tower, c = temp tower,t = destination tower
void hanoi(int n,char s,char c,char t)
{
    // base case when have 1 disk move form start to destination
    if(n == 1)
    {
        move(s,t);
        return;
    }
    //move disk form start tower to temp tower using destination tower as temp tower
    hanoi(n-1,s,t,c);
    move(s,t);
    //move disk form temp tower to destination tower using start tower as temp tower
    hanoi(n-1,c,s,t);
}
```