

# PD2017 Lab10

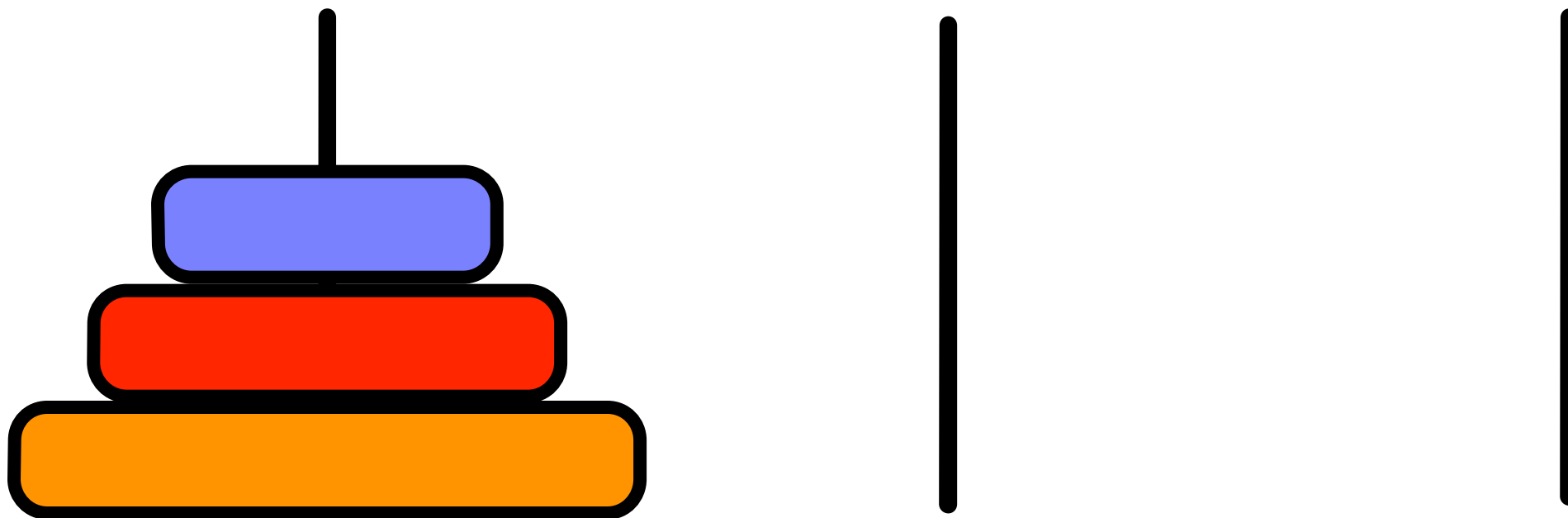
Jhih-Bang Hsieh

# Outline

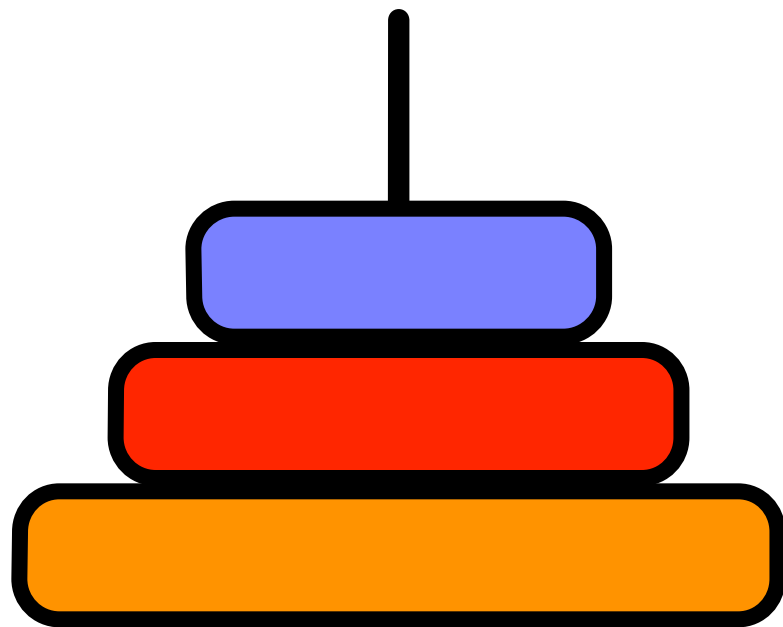
- Recursion
- Binary search
- Practice

# Hanoi Tower

- 有三個柱子，和三個碟子，碟子大小相異，且規定小碟子必須置放於大碟子之上。
- 初始狀態，三個碟子均置放於同一柱上，問將所有碟子搬移至另一柱上最少需要幾步？

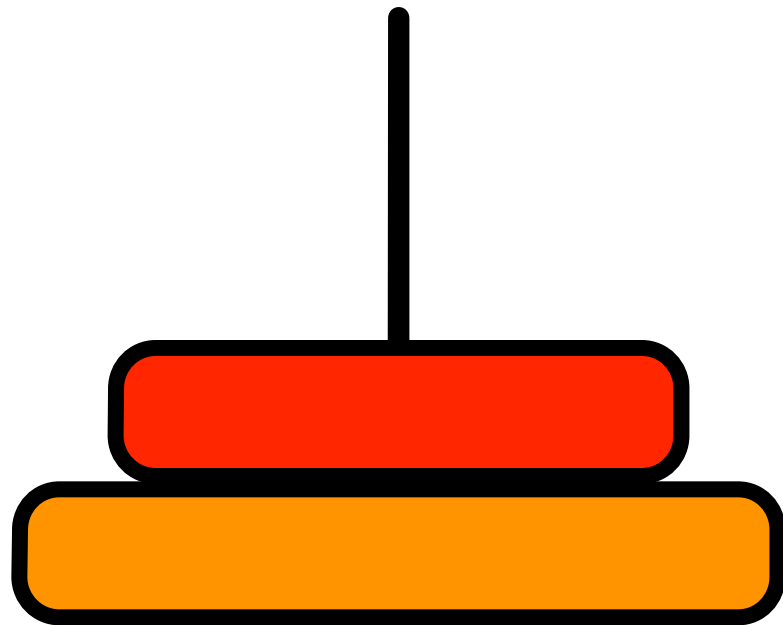


# Hanoi Tower



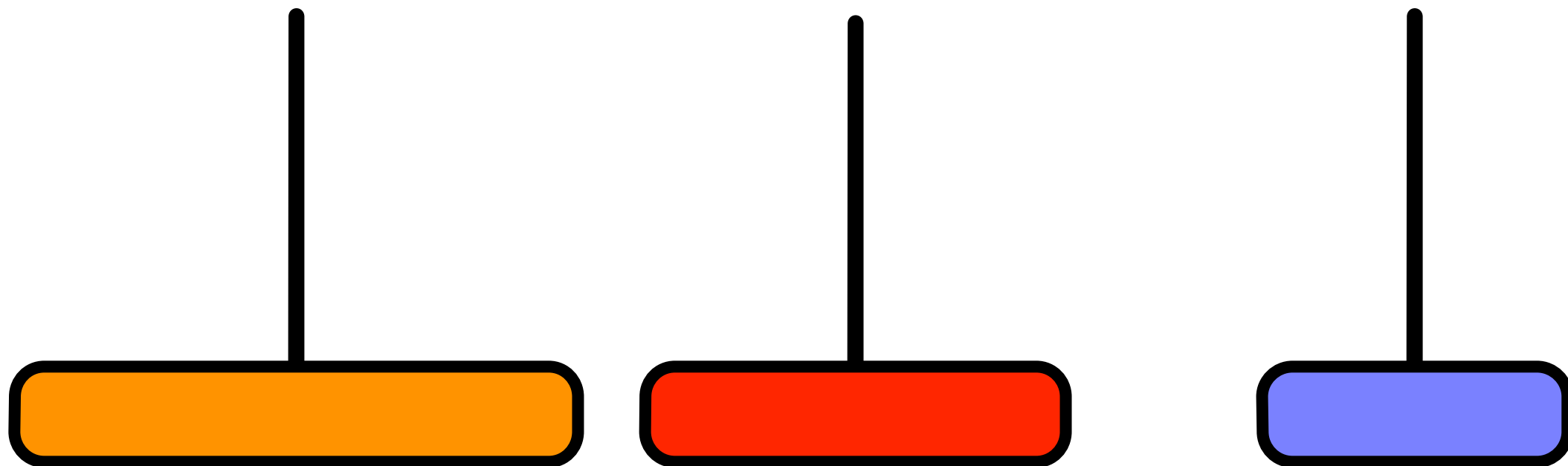
0

# Hanoi Tower



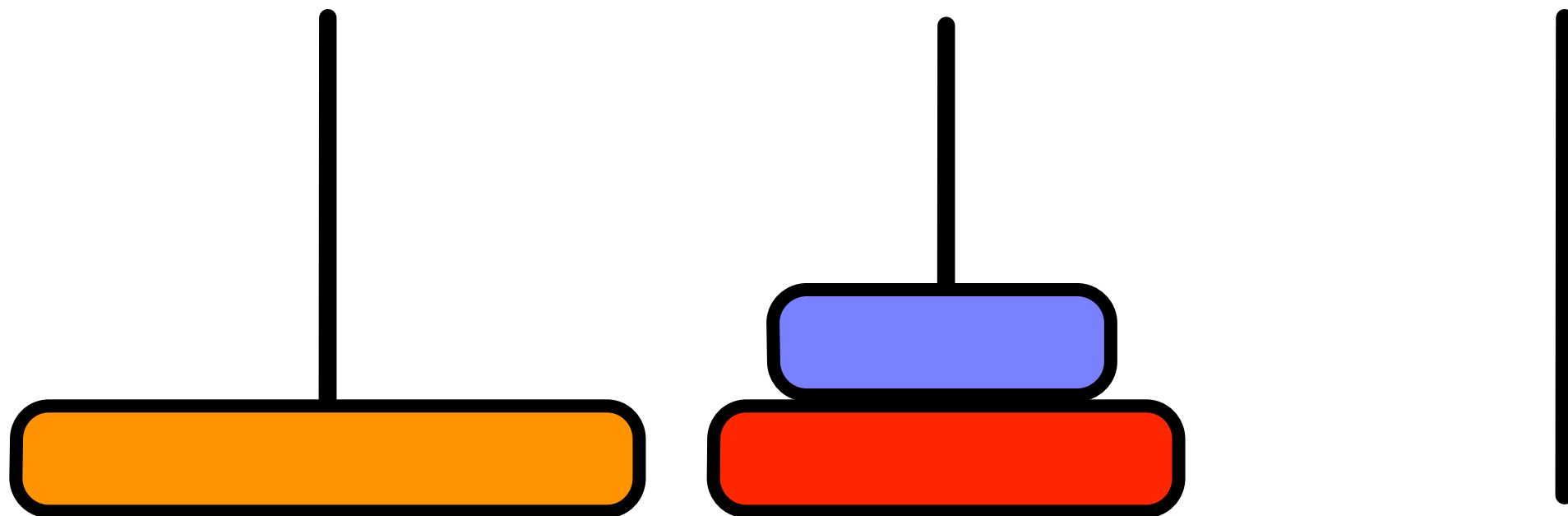
1

# Hanoi Tower



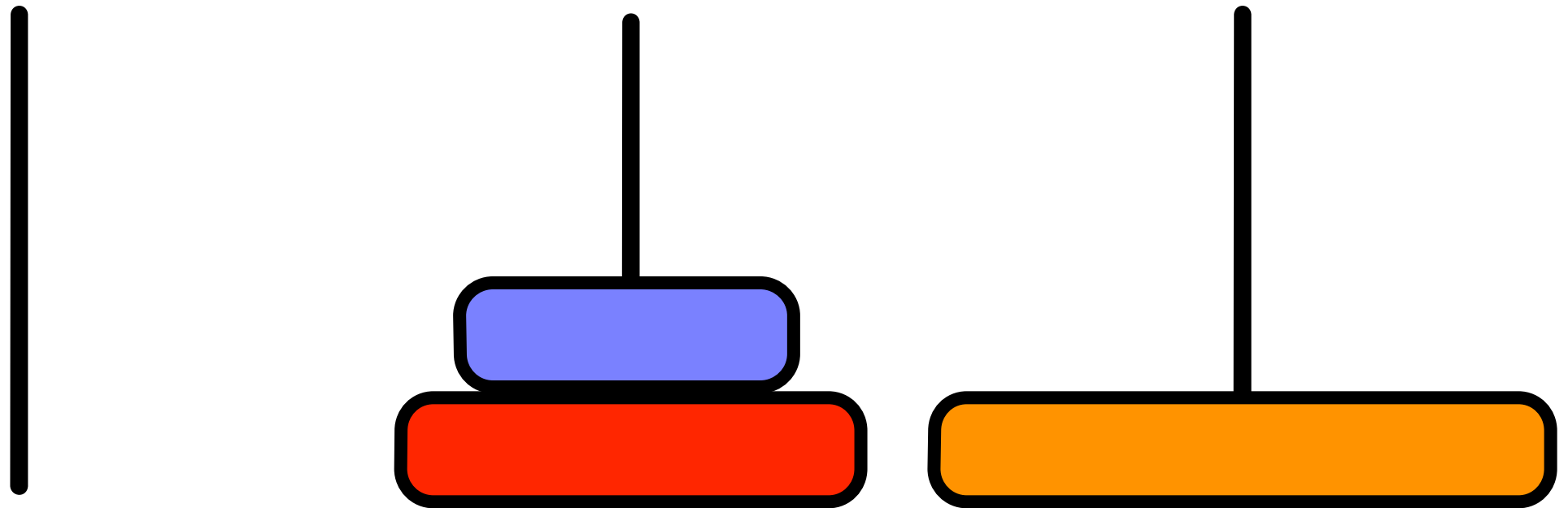
2

# Hanoi Tower



3

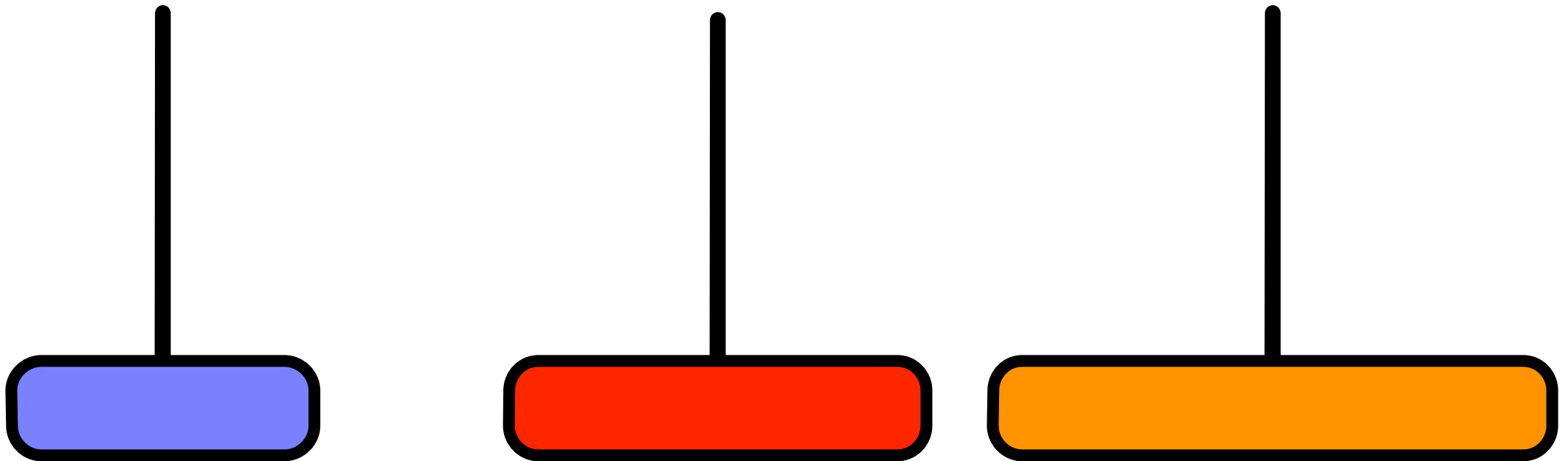
# Hanoi Tower



4

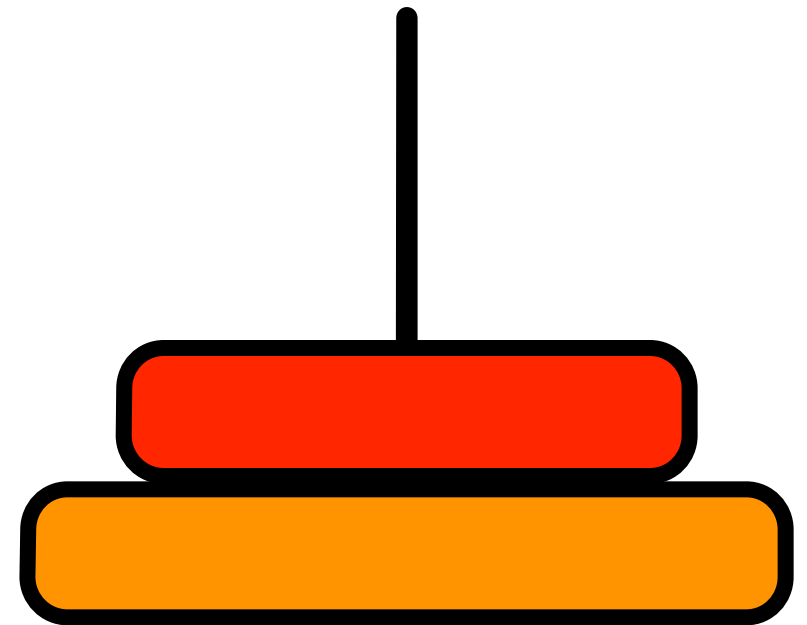
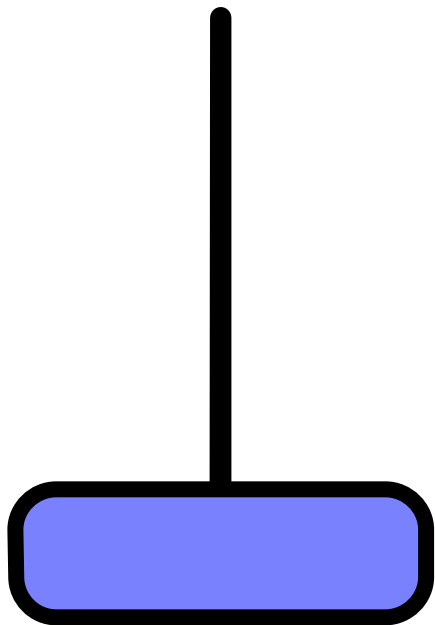


# Hanoi Tower



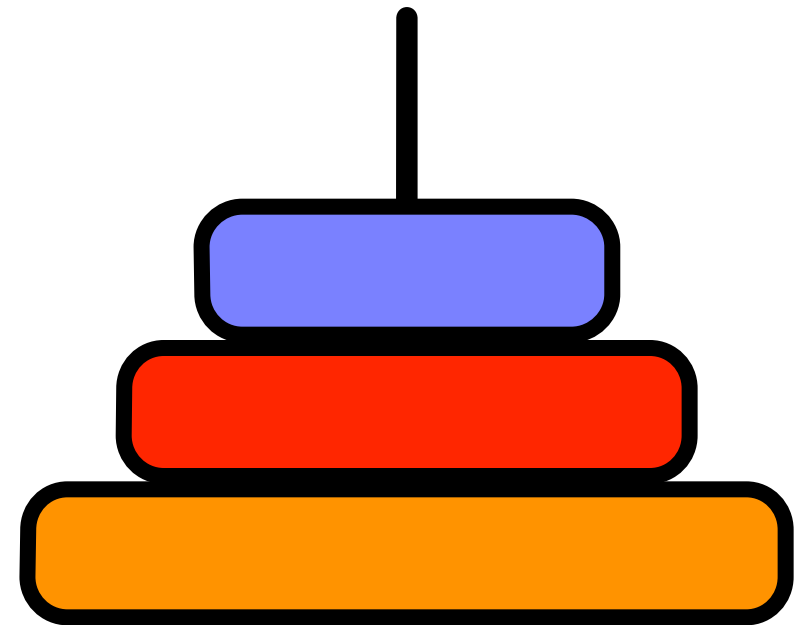
5

# Hanoi Tower



6

# Hanoi Tower

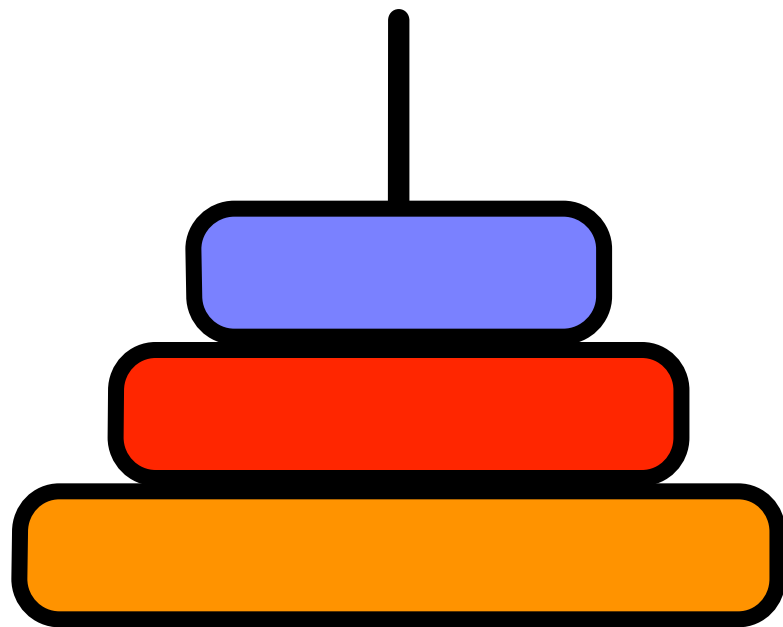


7

# Hanoi Tower

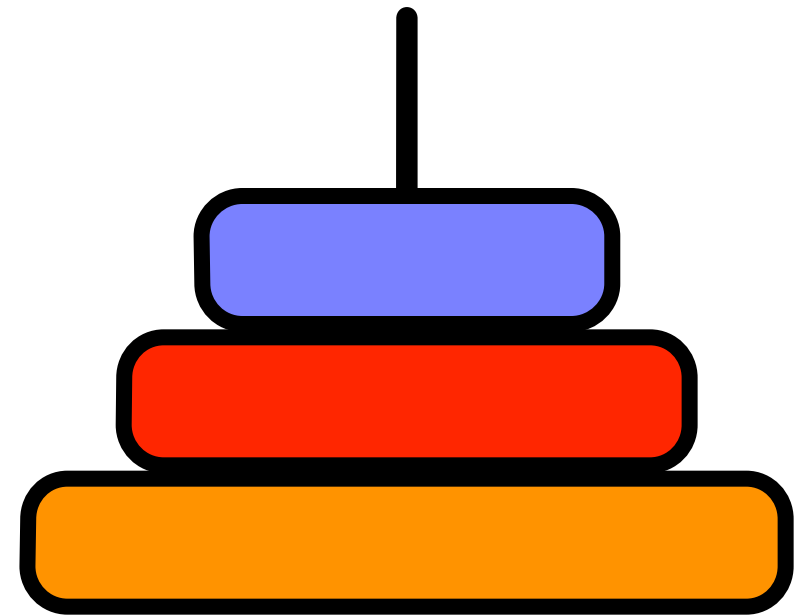
- 假設解決這個問題的函式是  $F(n)$ ， $n$  代表碟子數量。
- 我們需要的是  $F(3) = F(2) + F(1) + F(2)$ ！

# Hanoi Tower



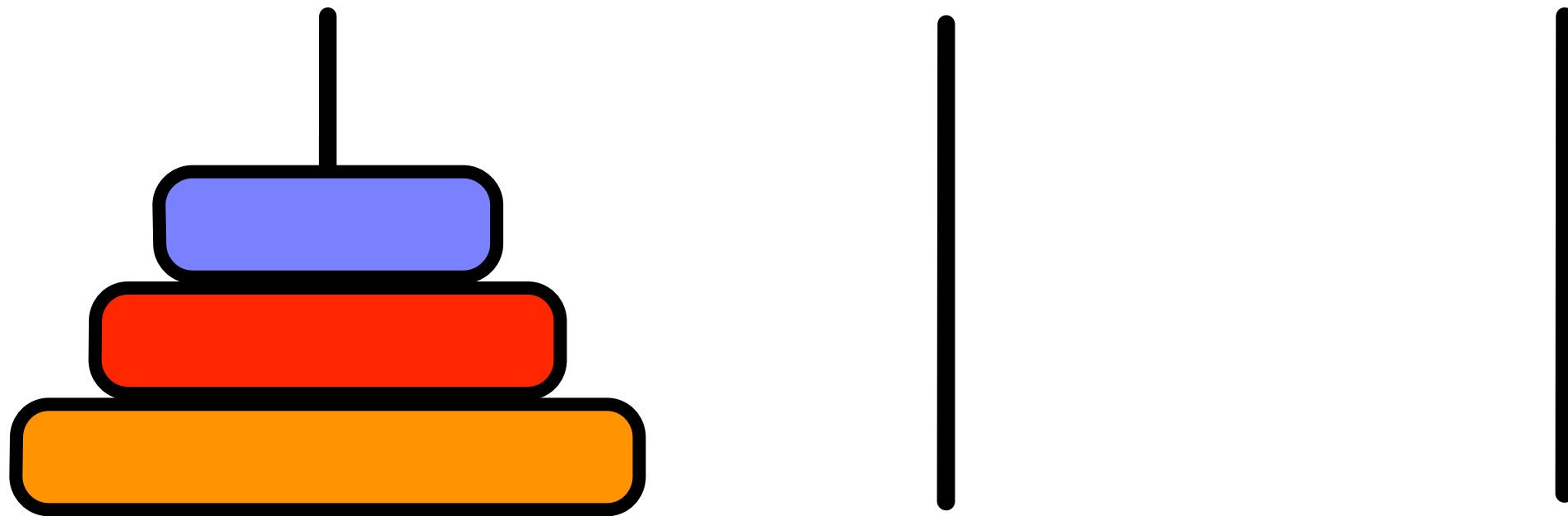
**F(3)**

# Hanoi Tower



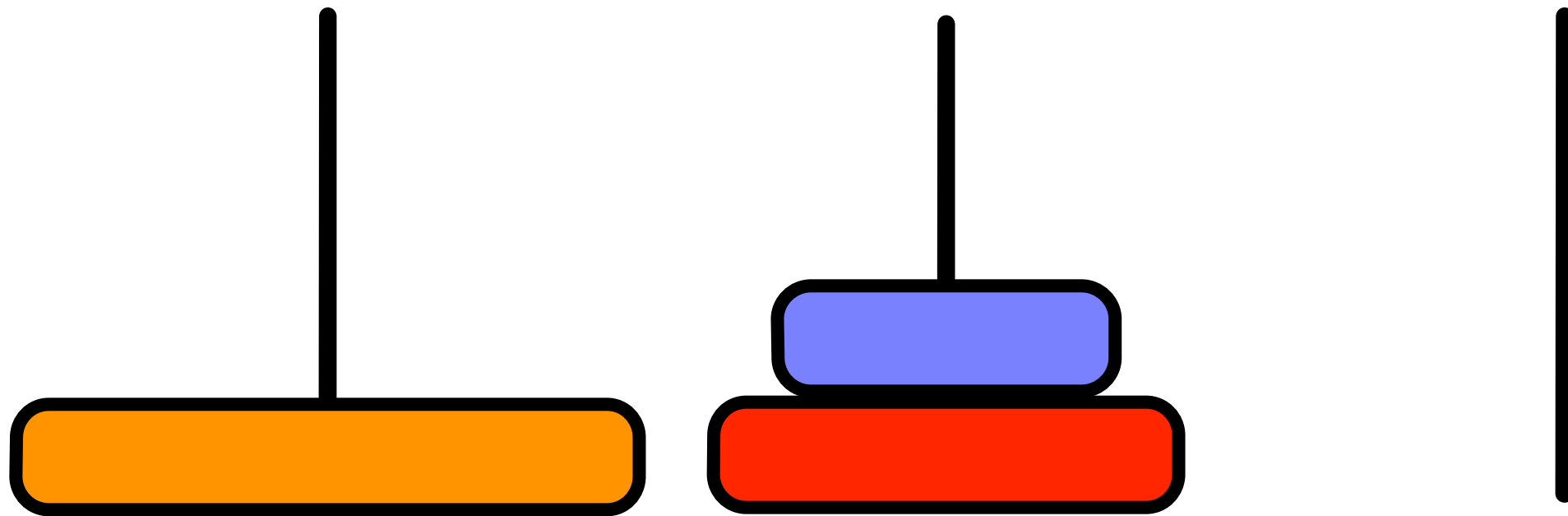
**F(3)**

# Hanoi Tower



$$F(3) = F(2) + F(1) + F(2)$$

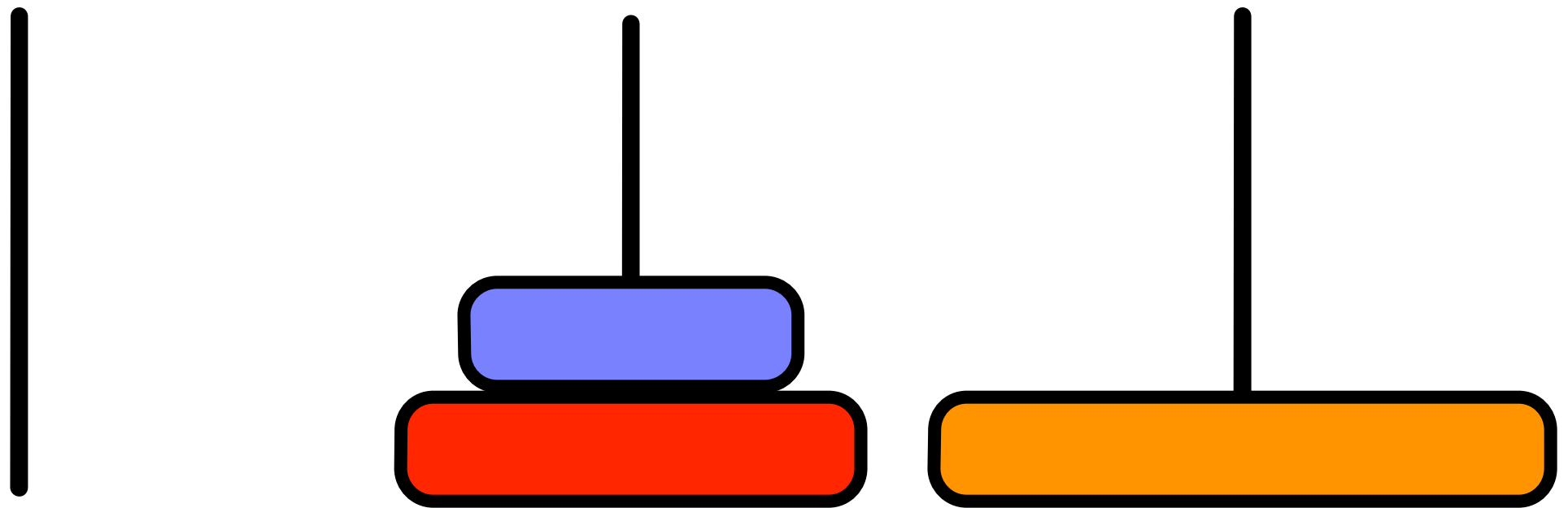
# Hanoi Tower



$$F(3) = F(2) + F(1) + F(2)$$

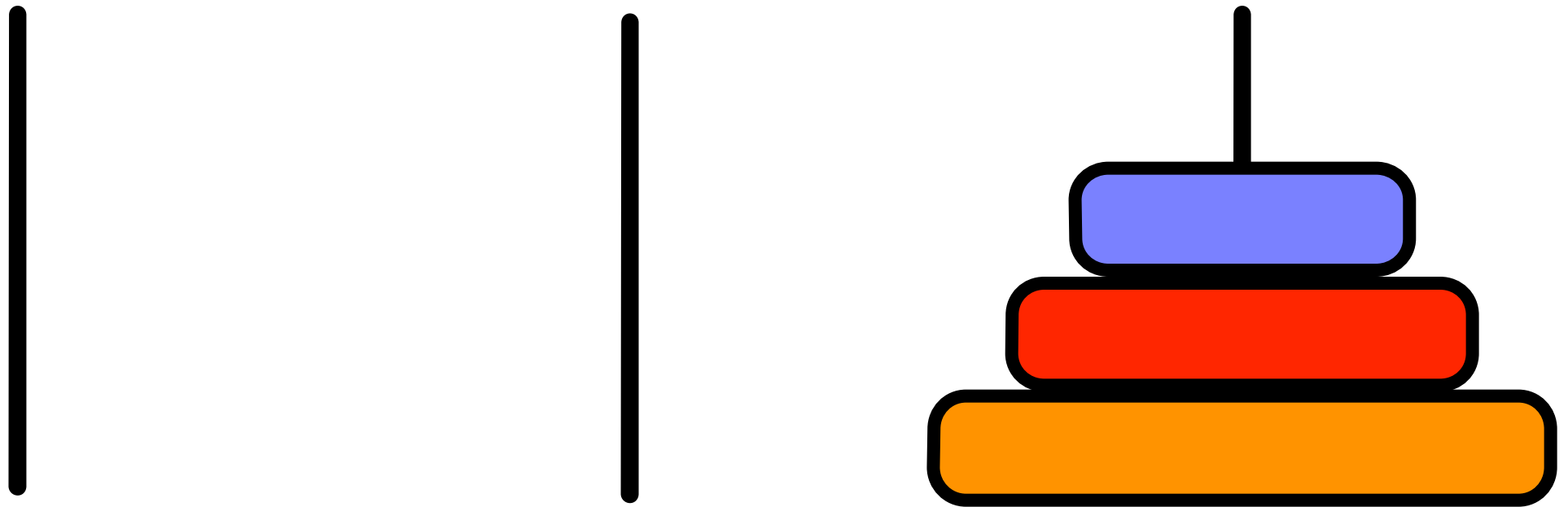


# Hanoi Tower



$$F(3) = F(2) + F(1) + F(2)$$

# Hanoi Tower



$$F(3) = F(2) + F(1) + F(2)$$

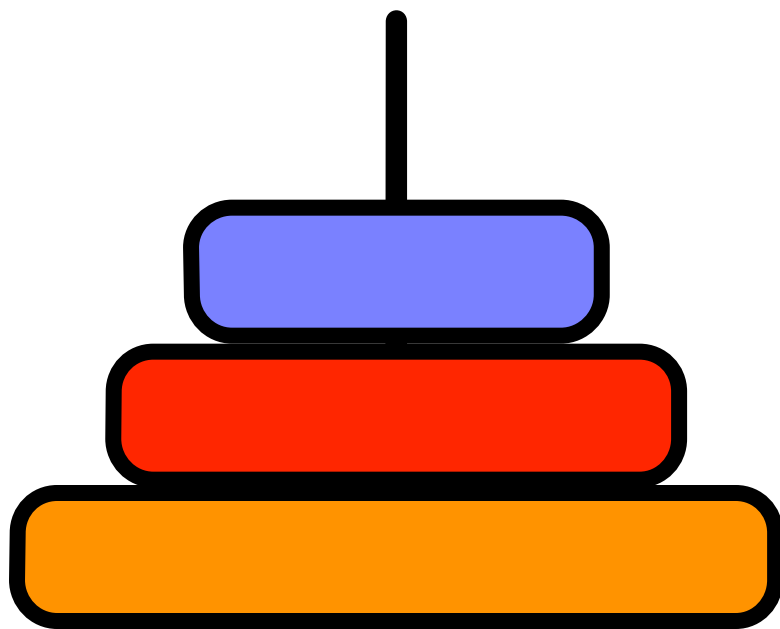
# Hanoi Tower

- 依照  $F(n) = F(n-1) + F(1) + F(n-1)$  的規律拆解。
- 其中  $F(1) = 1$ 。
- 寫程式時，遞迴的概念一模一樣！
- 如果將柱子編號為 1、2、3，該怎麼印出搬運的過程？

```
// move n plates from No.f to No.t
void mov(int n, int f, int t, int med) {
    if(n == 1) {
        cout << "Move from No." << f << " to No." << t << endl;
    } else {
        mov(n-1, f, med, t);
        cout << "Move from No." << f << " to No." << t << endl;
        mov(n-1, med, t, f);
    }
}
```

# 延伸

- 如果今天第一柱與第三柱之間不能移動，只能移動至相鄰的柱子上，則需要幾次操作？



# Binary search

- 中文叫做「二分搜尋法」。
- 顧名思義，每次都會做一次「二分」的動作。
- 假設一個大小為  $N$  的問題，逐步用「二分」拆解，則其複雜度為  $O(\log N)$ 。

# Binary search

- 想像一個「已排序」的數列。
- 我們想知道第一個大於等於某數的數字是多少？
- 例：第一個大於等於 7 的數字是多少？



# Binary search



- 每次將數列分成兩半



# Binary search



- 每次將數列分成兩半
- 問某一半邊有沒有可能大於等於該數

# Binary search



- 每次將數列分成兩半
- 問某一半邊有沒有可能大於等於該數
- 若左半邊可以，則忽略右半邊，否則忽略左半邊

# Binary search



- 每次將數列分成兩半
- 問某一半邊有沒有可能大於等於該數
- 若左半邊可以，則忽略右半邊，否則忽略左半邊
- 一次淘汰一半的候選數字！

# Practice

- 給一個排序好的數列和一個指定數字，問該數列內第一個大於等於該數字的數為何？

# Practice

- PDOGS Problem 11  
<https://pdogs.ntu.im/judge/problem/view/11>