递　归

周云晓

2017.3.9

内容

- 递归定义
- 递归例子
  - 斐波那契数列
  - 计算阶乘
  - 汉诺塔

定义

- 语言例子：

从前有座山，山里有座庙，庙里有个老和尚，正在给小和尚讲故事呢！故事是什么呢？

“从前有座山，山里有座庙，庙里有个老和尚，正在给小和尚讲故事呢！故事是什么呢？

‘从前有座山，山里有座庙，庙里有个老和尚，正在给小和尚讲故事呢！故事是什么呢？……’”

定义

- 递归算法
  - 一种直接或间接调用自身函数或方法的算法
- 实质
  - 把一个大的复杂的问题分解为规模缩小的和原问题相似的问题来求解

例子 -- 1

- 斐波那契数列 (Fibonacci sequence)
  - 递归定义：

$$F_0 = 0$$
$$F_1 = 1$$

出口条件

$$F_n = F_{n-1} + F_{n-2} \quad (n \geq 2)$$

递归方程

例子 -- 斐波那契数列

- 伪代码

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2} \quad (n \geq 2)$$

Define Fib(n):

n = 0时， return 0

n = 1时， return 1

n > 1时， return Fib(n-1) + Fib(n-2)

例子 -- 斐波那契数列

- Java实现

Define Fib(n):
n = 0时，return 0
n = 1时，return 1
n > 1时，return Fib(n-1) + Fib(n-2)

```
public int Fib (int n){
    if (n <= 1)
        return n;
    else
        return Fib(n-1) + Fib(n-2);
}
```

出口条件

例子 -- 斐波那契数列

- Java实现

Fibonacci.java

```
1  public class Fibonacci{
2
3      public int Fib (int n){
4          if (n <= 1)
5              return n;
6          else
7              return Fib(n-1) + Fib(n-2);
8      }
9
10     public static void main(String args[]){
11         Fibonacci fibonacci = new Fibonacci();
12         int result = fibonacci.Fib(10);
13         System.out.println(result);
14     }
15 }
16
17
```

例子 -- 2

- 计算阶乘
  - 1! = 1
  - 2! = 2 * 1
  - 3! = 3 * 2 * 1
  - 4! = 4 * 3 * 2 * 1

  - n! = n * (n-1) * …* 2 * 1

例子 -- 计算阶乘

- Java实现：
  - 设 F_n 为n的阶乘：

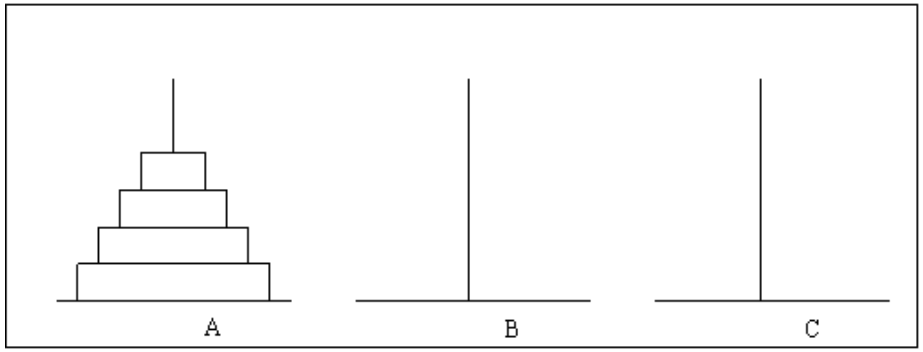
$$F_1 = 1$$

$$F_n = n * F_{n-1} \quad (n > 1)$$

```
public int fac(int n){
    if (n == 1)
        return 1;
    else
        return n * fac(n-1);
}
```

例子 -- 3

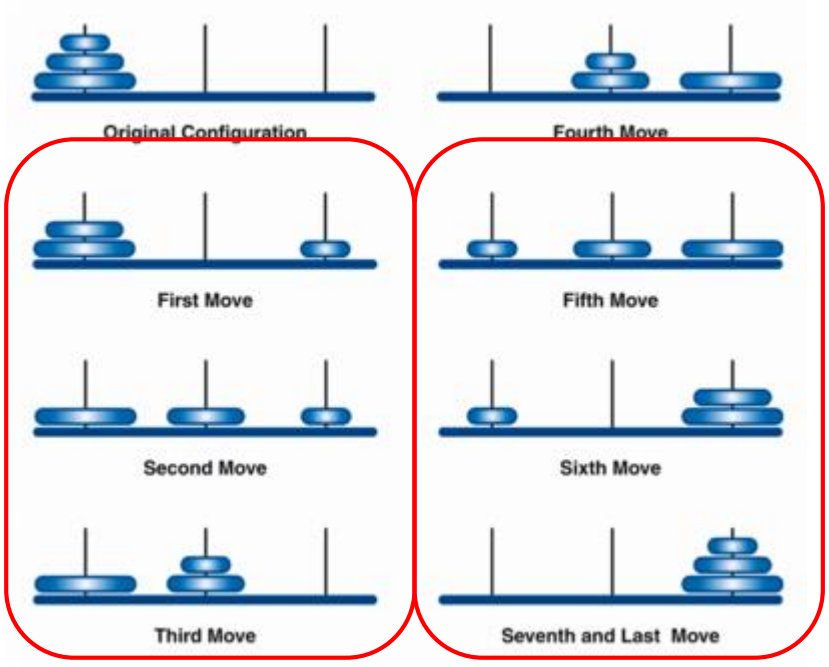
- 汉诺塔 (Hanoi Tower)
  - 有三根杆子A, B, C。A杆上有N个（N>0）圆盘，盘的尺寸由上到下依次变大。要求按下列规则将所有圆盘移至C杆：
    - 每次只能移动一个圆盘
    - 大盘不能叠在小盘上面
  - 求：最少移动多少次？



例子 -- 汉诺塔

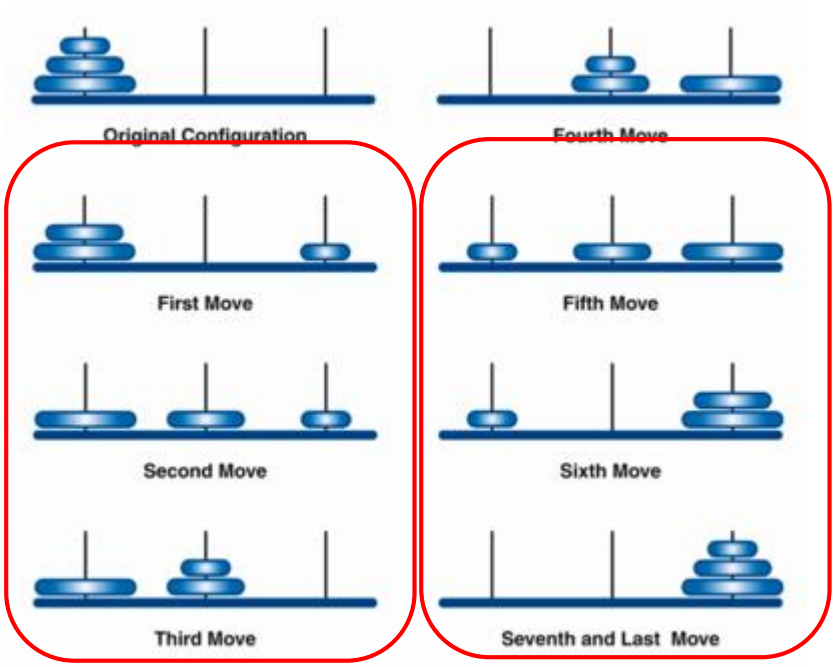
- 1个盘子： 1次
- 2个盘子： 3次
- 3个盘子： 7次
- …

- n个盘子： T_n次



例子 -- 汉诺塔

- n个盘子： T_n次
  - (n-1)个盘子从 A → B
  - 最大盘子从 A → C
  - (n-1)个盘子从 B → C



- $T_n = T_{n-1} + 1 + T_{n-1}$

  $= 2* Tn-1 + 1 \quad (n > 1)$

例子 -- 汉诺塔

- Java实现

$$T_1 = 1$$

$$T_n = 2 * T_{n-1} + 1 \quad (n > 1)$$

```
public int hanoi(int n){
    if (n == 1)
        return 1;
    else
        return 2*hanoi(n-1) + 1;
}
```

例子 -- 汉诺塔

- Java实现

Hanoi.java

```
1  public class Hanoi{
2
3      public int hanoi(int n){
4          if (n == 1)
5              return 1;
6          else
7              return 2*hanoi(n-1) + 1;
8      }
9
10     public static void main(String args[]){
11         Hanoi hanoi_1 = new Hanoi();
12         int result = hanoi_1.hanoi(6);
13         System.out.println(result);
14     }
15 }
16
17
```

练习

- 母牛生小牛问题
  - 设有一头小母牛，从出生第四年起每年生一头小母牛，按此规律，第N年时有几头母牛？

```
public int func(int year){
    if (year < 4)
        return 1;
    else
        return func(year - 3) + func(year - 1);
}
```