

第三单元 第三节

# 递归与汉诺塔问题



01

什么是递归？

从前有个学校  
学校里有个老师  
老师给同学们讲故事

从  
学校里有个老师  
老师给同学们讲故事

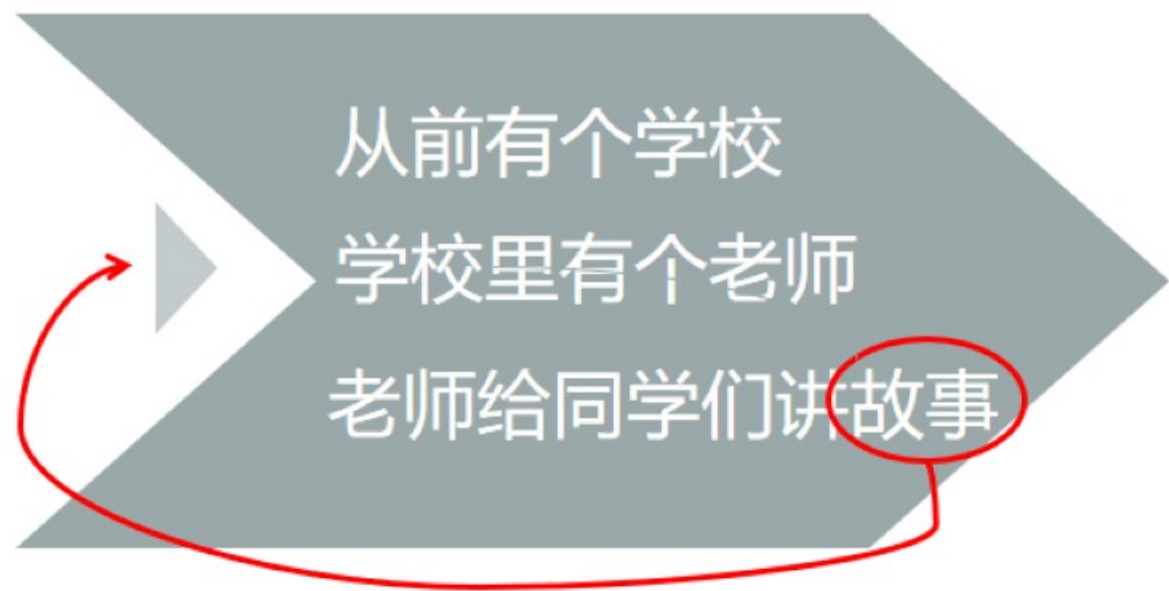
学校里有个老师  
老师给同学们讲故事

从前有个学校  
学校里有个老师  
老师给同学们讲故事

从前有个学校  
学校里有个老师  
老师给同学们讲故事

故事的特点?

# 递归



故事中提到同样的故事

函数A



调用

函数B

函数A



函数在运行时调用自己



02

## 计算阶乘



## 问题分析

阶乘通常定义为：

$$f(n) = n! = n \times (n-1) \times (n-2) \times \cdots \times 3 \times 2 \times 1$$

也可定义为：

$$f(n) = n \times f(n-1) \quad \longrightarrow \quad \text{递归定义}$$

阶乘函数可定义为：

$$f(n) = \begin{cases} 1 & n=0 \\ n \times f(n-1) & n>0 \end{cases} \quad \longrightarrow \quad \text{边界条件}$$

设  $f(n) = n!$  //  $n$  为自然数

如果  $n=0$ ，则  $f(n)=1$

否则， $f(n) = n \times f(n-1)$



03

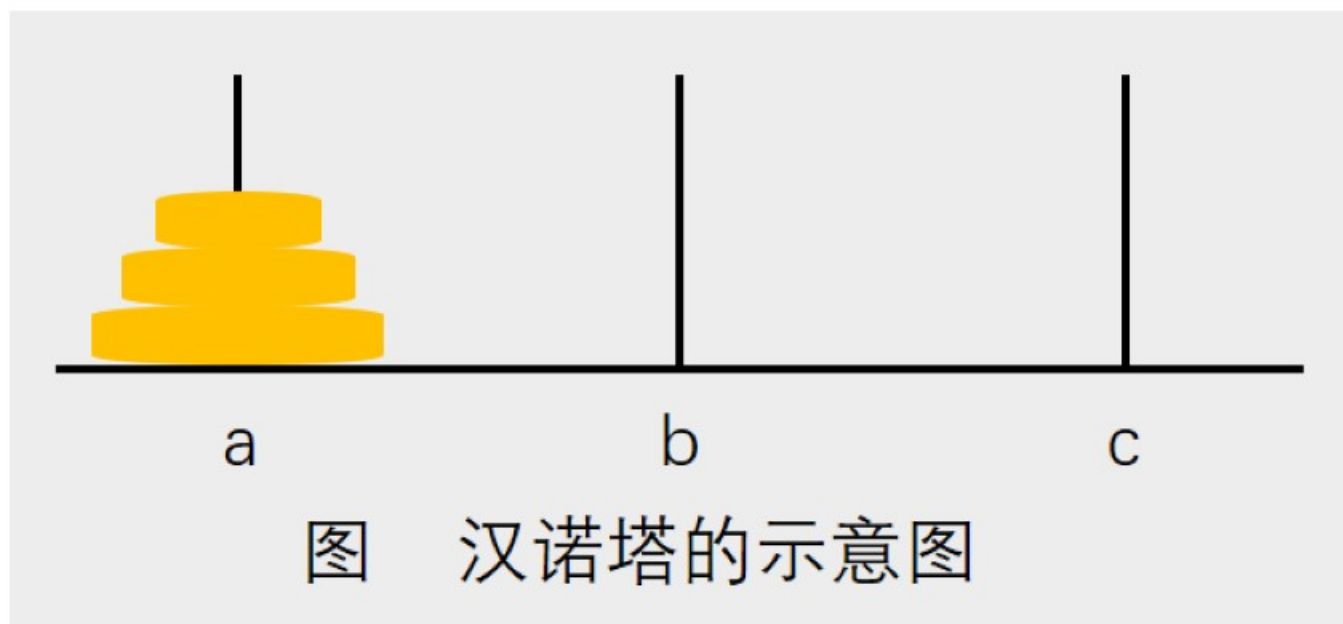
## 汉诺塔问题





由从小到大的圆盘和三根柱子组成

**规则：**一次只能移动一个圆盘，小圆盘只能叠放在大圆盘上。通过移动圆盘，将所有圆盘移动到另一根目标柱上



**递归法：**将规模繁杂的问题“递”减到规模较小较简单的问题，通过简单的问题找到其规律，再将规律“归”纳到原本繁杂的问题上来解决原来的问题



一块 ————— 1次

两块 ————— 3次

三块 ————— 7次

奇数层的汉诺塔:

要将第1、3、5……层, 放到目标柱上

偶数层的汉诺塔:

要将第2、4、6……层, 放到目标柱上

六十四块 ————— 世界末日会来嘛?

```
void Hanoi(int n,char A,char B,char C){  
if(n==1)  
printf("\t%c->%c\n",a,c); //当n只有1个的时候直接从a移动到c  
else  
{  
move(n-1,a,c,b); //前n-1个要从a通过c移动到b  
printf("\t%c->%c\n",a,c);  
move(n-1,b,a,c); //n-1个移动过来之后b变开始盘, b通过a移动到c  
}  
}
```





04

## 课堂任务


1、递归算法



2、编程实现



两个遗留问题



感谢观看  
还请指正