**时间复杂度**：用来评估算法运行效率的式子 O(n)

Print()

Print() O(1)

For I in range(n):

Print()

For I in range(n):

Print() O(n²)

While n>1:

Print(n) 64 = 2的六次方 (log2 64 = 6)

N = n//2 O(log n)

1. 确定 问题规模n
2. 循环减半过程 log n
3. K层关于n的循环 n
4. 根据算法的执行过程判断

**空间复杂度**：评估算法内存占用大小的式子

(1)算法使用几个变量 O(1)

(2)算法使用长度为n的一维列表 O(n)

(3)算法使用m行n列的二位列表 O(mn)

**递归**：

调用自身，结束条件

Def func3(x):

If x>0:

Print(x)

Func3(x-1) 先打印后递归

Def func4(x):

If x>0:

Func4(x-1)

Print(x) 先递归后打印

1. 汉诺塔

1//把n-1个盘子从A经过C移动到B

2//把第n个盘子从A移动到C

3//把n-1个盘子从B经过A移动到C

H(x) = 2 h(x-1) + 1