编程题:有n步台阶,一次只能上1步或2步,共有多少种走法?

- 1. 递归
- 2. 循环迭代

高频面试题系列

->一步

递归

• n=1

(2) 先到达f(x-1), 然后从f(x-1) 跨1步

->f(1) = 1

->f(x) = f(x-2) + f(x-1)

高频面试题系列

循环迭代

one保存最后走一步 two保存最后走两步

• n=2

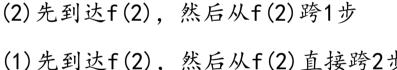
• n=3

• n=4

• n=x

->(1) 先到达f(x-2), 然后从f(x-2) 直接跨2步

(2) 先到达f(x-1), 然后从f(x-1) 跨1步



$$->f(2) = 2$$

 $->f(3) = t$
 $f(3) = f$

->f(1) = 1

$$->f(3) = two + one$$

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

->f(4) = two + one

->f(x) = two + one

f(4) = f(2) + f(3)

two = f(2); one = f(3)

f(x) = f(x-2) + f(x-1)

two = f(x-2); one = f(x-1)

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

two = f(1); one = f(2)

小结

- 方法调用自身称为递归, 利用变量的原值推出新值称为迭代。
- 递归
 - 优点: 大问题转化为小问题, 可以减少代码量, 同时代码精简, 可读性好;
 - 缺点: 递归调用浪费了空间, 而且递归太深容易造成堆栈的溢出。

• 迭代

- 优点: 代码运行效率好, 因为时间只因循环次数增加而增加, 而且没有额外的空间开销;
- 缺点: 代码不如递归简洁, 可读性好