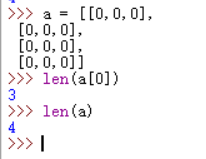
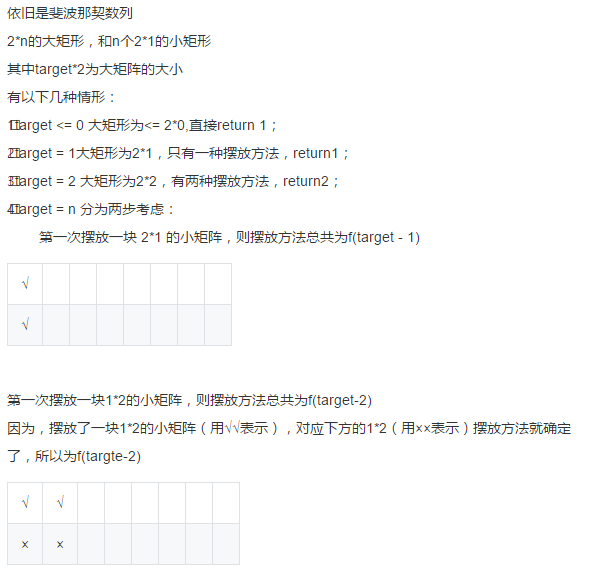
01：



10矩形覆盖：



n：摆放第一个小矩阵分两种情况：

因此：f(n) = f(n-1) + f(n-2)

011-二进制中1的个：

n = n & 0xffffffff原因:

对于负数，最高位为1，而负数在计算机是以补码存在的，往右移，符号位不变，符号位1往右移，最终可能会出现全1的情况，导致死循环。

与0xffffffff相与，就可以消除负数的影响:相当于对负数的补码再求了一次补码。变成不带符号的数了。

计算机中的负数是以其补码形式存在的 补码=原码取反+1  
一个字节有8位 可以表示的数值范围在 -128到+127  
用二进制表示也就是 10000000 - 01111111（注意：最高位表示符号）  
最高位是1的都是负数 最高位是0的都是正数  
如-7 原码是 10000111 然后取反（最高位是符合不用取反）得11111000  
加一 得11111001 那么-7的二进制数就是 11111001  
再如 -10 原码是 10001010 取反得 11110101 加一得 11110110  
那么-10的二进制数就是 11110110  
二进制数是逢二进一 只有0和1两个数字 没有2