```
import Foundation
// 剑指 Offer 10-I. 斐波那契数列
// 写一个函数,输入 n , 求斐波那契 (Fibonacci) 数列的第n项 (即 F(N))。
// 答案需要取模 1e9+7 (1000000007), 如计算初始结果为: 1000000008, 请返回 1。
class Solution {
   func fibSolution1(_ n: Int) -> Int {
       if n < 2 { return n }
       var a = 0
       var b = 1
       for _ in 0..<n-1 {
           (a, b) = (b, (a + b) \% 1000000007)
       }
       return b
   }
   func fib(_ n: Int) -> Int {
       if n < 2 { return n }
       let result = matrixPow([[1,1], [1,0]], n: n-1-1)
       return result[0][0]
   }
   /// 快速幂算法
   func matrixPow(_ matrix: [[Int]], n: Int) -> [[Int]] {
       var _matrix = matrix
       var_n = n
       var result = [[1,1], [1,0]]
       while _n > 0 {
           /// 判断n的奇偶性, &为按位与运算符
           /// 如果是奇数则需要多乘一次矩阵
           if (_n & 1 == 1) { result = multiply(a: result, b: _matrix) }
           _n >>= 1
           matrix = multiply(a: matrix, b: matrix)
       return result
   }
   func multiply(a: [[Int]], b: [[Int]]) -> [[Int]] {
       var c = [[0,0], [0,0]]
       for i in 0..<2 {
           for j in 0..<2 {
               c[i][j] = (a[i][0] * b[0][j] + a[i][1] * b[1][j]) % 1000000007
           }
       }
       return c
   }
}
```