### 1.4斐波那契数列最终版

// 创建缓存容器

// function createCache(){

// var cache = {};

// return function (key, value) {

// //如果传了值，就说名是设置值

// if(value !== undefined){

// cache[key] = value;

// return cache[key];

// }

// //如果没有传值，只穿了键，那就是获取值

// else{

// return cache[key];

// }

// }

// }

var count =0 ;

function createFib(){

var fibCache = createCache();

function fib(n){

count ++;

//1.从cache中获取数据

if(fibCache(n) !== undefined){

//如果缓存中有 直接返回

return fibCache(n) ;

}

//如果缓存中没有 就计算

if(n <= 2){

//把计算结果存入数组

fibCache(n , 1) ;

return 1;

}

var temp = fib(n - 1) + fib(n - 2);

//把计算结果存入数组

fibCache(n, temp) ;

return temp;

}

return fib;

}