const ary = (fn, n) => (args) => fn(...args.slice(0, n))

// 多个箭头函数其实很简单，就是可以接收多个参数，执行的时候依次传入参数

Reduce函数：

arr.reduce(callback(accumulator, currentValue[, index[, array]])[, initialValue])

观察者模式：

Subject：维护他的依赖数组

Observer：

观察者模式实现异步

Thunk函数：

编译器的传名调用实现，往往就是将参数放到一个临时函数之中，再将这个临时函数转入函数体，这个临时函数就叫做Thunk函数。