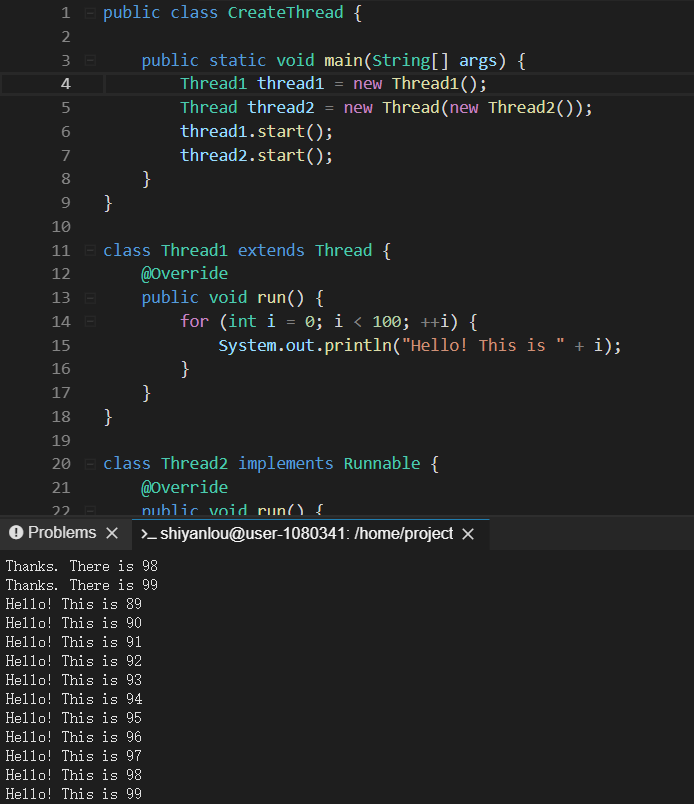
20190602学习心得

1. 一条线程指的是进程中一个单一顺序的控制流，一个进程中可以并发多个线程，每条线程并行执行不同的任务。多线程是多任务的一种特别的形式，但多线程使用了更小的资源开销。多线程能满足程序员编写高效率的程序来达到充分利用 CPU 的目的
2. 线程

程序执行流的最小单元。它是进程内一个相对独立的、可调度的执行单元，是系统独立调度和分派 CPU 的基本单位

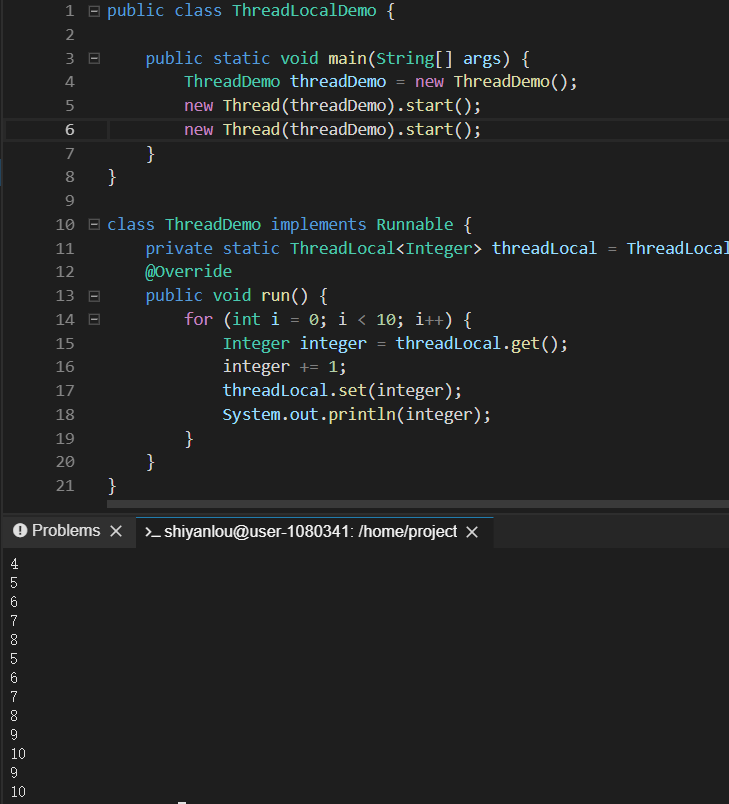
1. 创建多线程
2. 继承 Thread 类并重写它的run()方法，然后用这个子类来创建对象并调用start()方法
3. 定义一个类并实现 Runnable 接口，实现run()方法



1. 线程变量

ThreadLocal，即线程变量，是一个以 ThreadLocal 对象为键、任意对象为值的存储结构。这个结构被附带在线程上，也就是说一个线程可以根据一个 ThreadLocal 对象查询到绑定在这个线程上的一个值

可以通过 set(T)方法来设置一个值，在当前线程下再通过 get()方法获取到原先设置的值

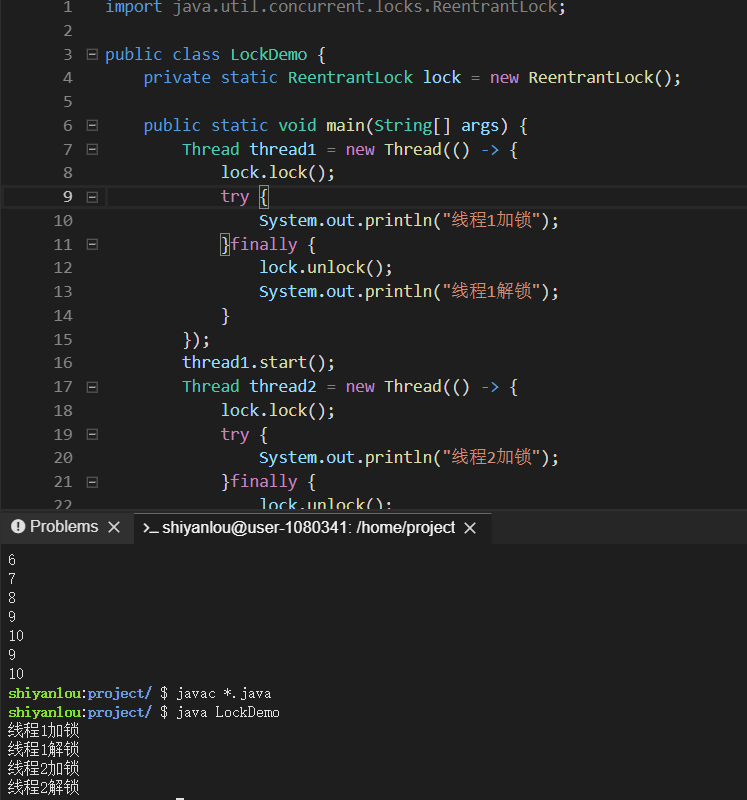


1. 线程同步

线程同步可以保证在同一个时刻该对象只被一个线程访问

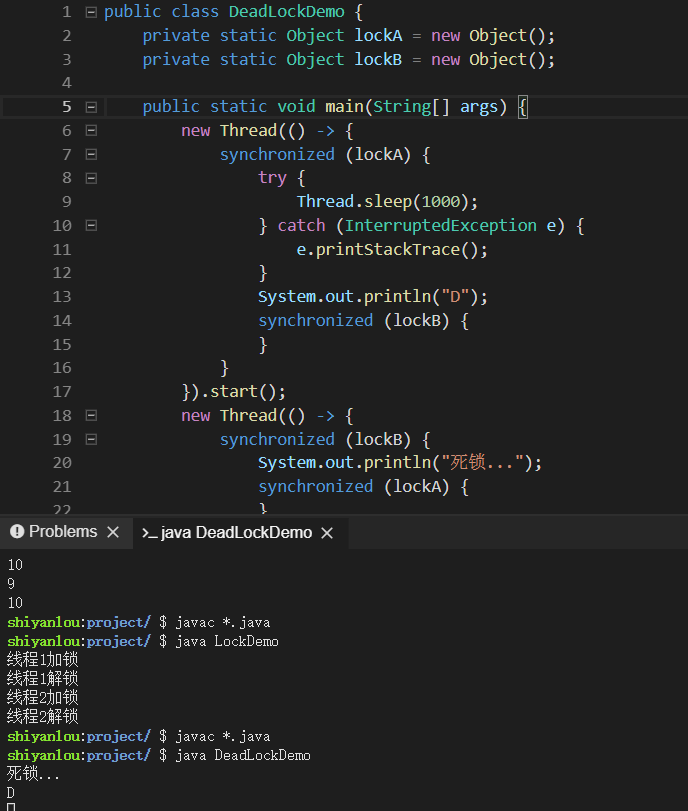
关键字 synchronized 可以修饰方法或者以同步块的形式来进行使用，它确保多个线程在同一个时刻，只能有一个线程处于方法或者同步块中，保证了线程对变量访问的可见性和排他性

1. java.util.concurrent 包是 java5 开始引入的并发类库，提供了多种在并发编程中的适用工具类。包括原子操作类，线程池，阻塞队列，Fork/Join 框架，并发集合，线程同步锁等
2. Lock和Unlock



1. 死锁

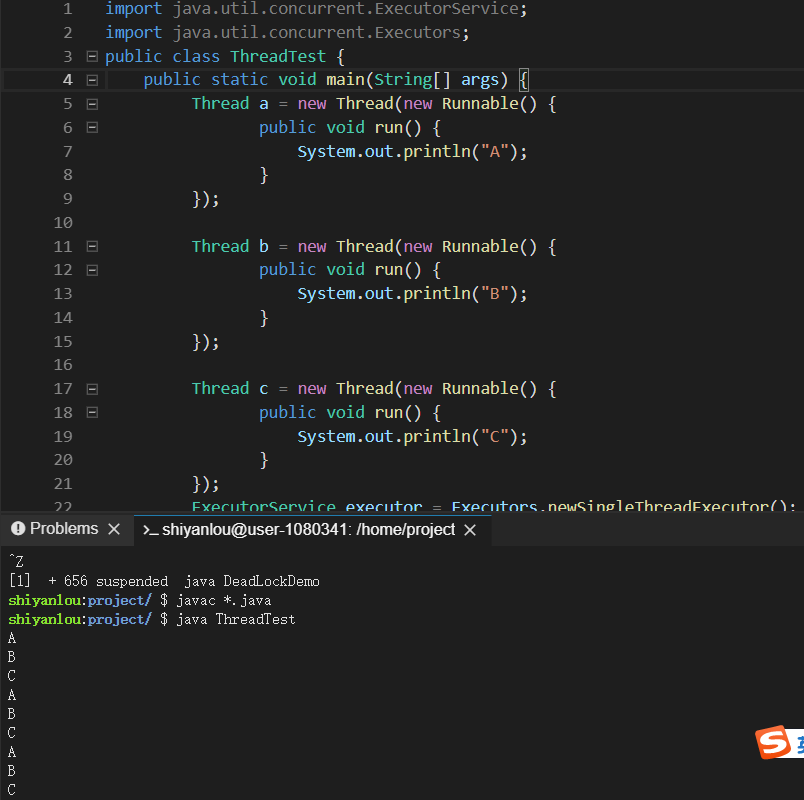
线程 1 获取了 lockA 的锁后再去获取 lockB 的锁，而此时 lockB 已经被线程 2 获取，同时线程 2 也想获取 lockA，两个线程进这样僵持了下去，谁也不让，造成了死锁



1. 饥饿

饥饿是指一个可运行的进程尽管能继续执行，但被调度器无限期地忽视，而不能被调度执行的情况

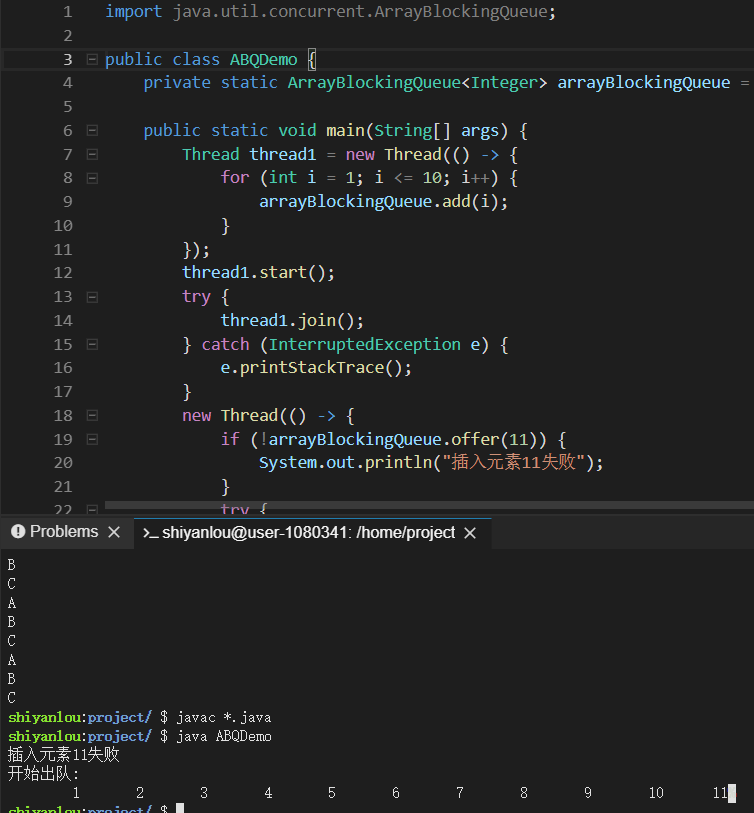
1. 顺序打印字母练习



1. ArrayBlockingQueue 是由数组支持的有界阻塞队列

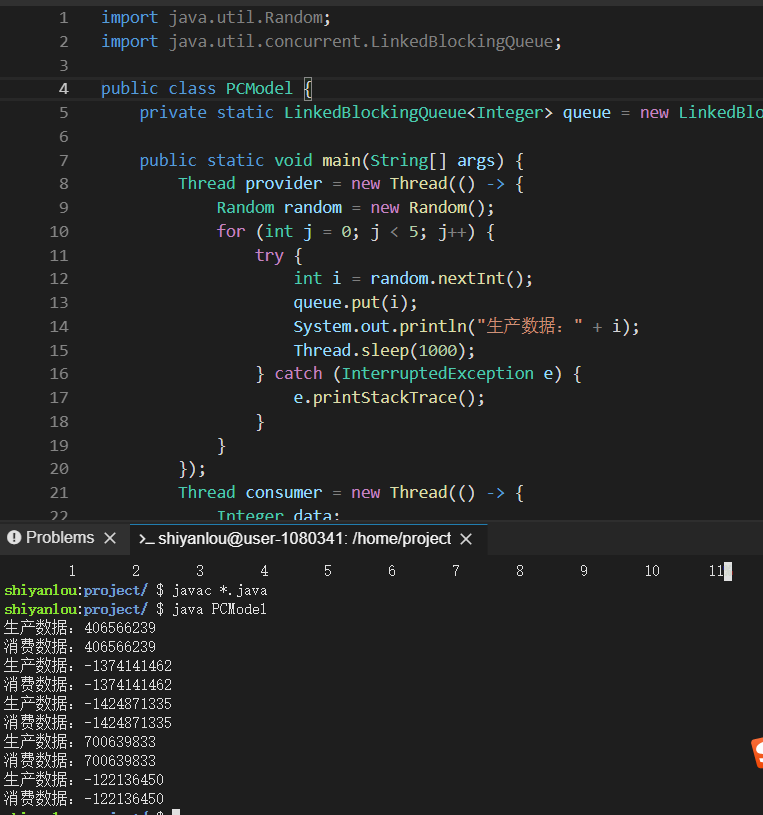




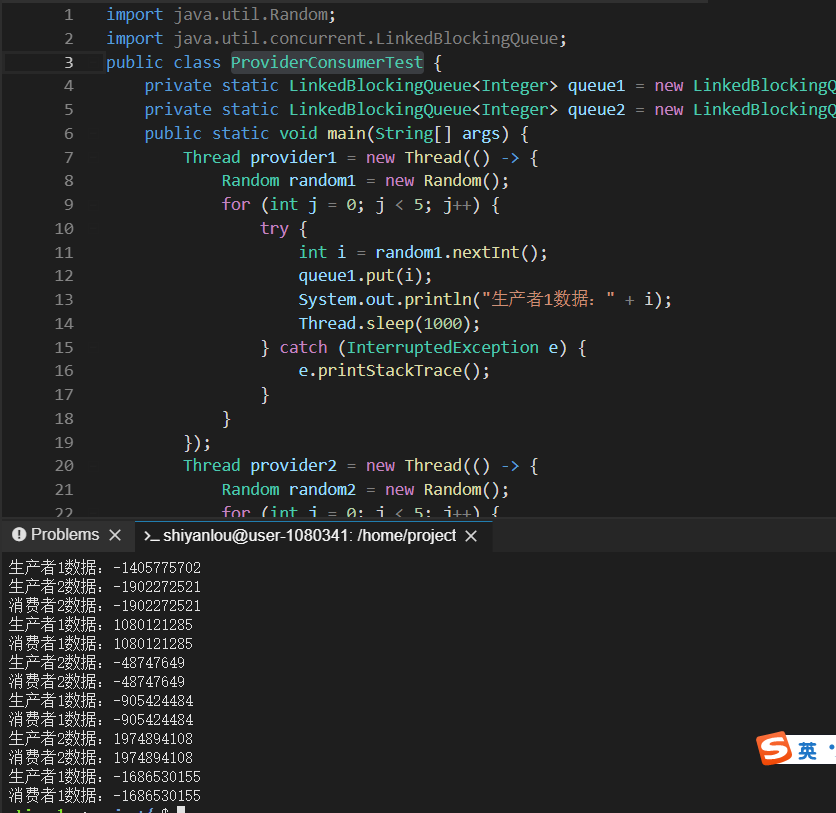


1. 生产者消费者模式

生产者消费者模式是多线程编程中非常重要的设计模式，生产者负责生产数据，消费者负责消费数据。生产者消费者模式中间通常还有一个缓冲区，用于存放生产者生产的数据，而消费者则从缓冲区中获取，这样可以降低生产者和消费者之间的耦合度



1. 多个生产者和消费者生成和消费数据练习



1. 线程池

一种线程使用模式。线程过多会带来调度开销，进而影响缓存局部性和整体性能。而线程池维护着多个线程，等待着监督管理者分配可并发执行的任务。这避免了在处理短时间任务时创建与销毁线程的代价

明日计划JDBC学习