

计算机二级考试java辅导:Java同步机制浅谈计算机二级考试

PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/581/2021_2022__E8_AE_A1_

[E7_AE_97_E6_9C_BA_E4_c97_581990.htm](https://www.100test.com/kao_ti2020/581/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E4_c97_581990.htm) 2009年下半年全国计算机等级考试你准备好了没?考计算机等级考试的朋友,2009年下半年全国计算机等级考试时间是2009年9月19日至23日。

更多优质资料尽在百考试题论坛 百考试题在线题库 Java对多线程的支持与同步机制深受大家的喜爱，似乎看起来使用

了synchronized关键字就可以轻松地解决多线程共享数据同步问题。到底如何？——还得对synchronized关键字的作用进行

深入了解才可定论。总的说来，synchronized关键字可以作为函数的修饰符，也可作为函数内的语句，也就是平时说的同步方法和同步语句块。如果再细的分类，synchronized可作用于instance变量、object reference（对象引用）、static函数

和class literals(类名称字面常量)身上。A.无论synchronized关键字加在方法上还是对象上，它取得的锁都是对象，而不是把一段代码或函数当作锁——而且同步方法很可能还会被其他线程的对象访问。B.每个对象只有一个锁（lock）与之相关联。C.实现同步是要很大的系统开销作为代价的，甚至可能造成死锁，所以尽量避免无谓的同步控制。接着来讨

论synchronized用到不同地方对代码产生的影响：假设P1、P2是同一个类的不同对象，这个类中定义了以下几种情况的同步块或同步方法，P1、P2就都可以调用它们。1.

把synchronized当作函数修饰符时，示例代码如下：Public synchronized void methodAAA() { //.... } 这也就是同步方法，那这时synchronized锁定的是哪个对象呢？它锁定的是调用这

个同步方法对象。也就是说，当一个对象P1在不同的线程中执行这个同步方法时，它们之间会形成互斥，达到同步的效果。但是这个对象所属的Class所产生的另一对象P2却可以任意调用这个被加了synchronized关键字的方法。上边的示例代码等同于如下代码：`public void methodAAA() { synchronized (this) // (1) { //..... } }`(1)处的this指的是什么呢？它指的就是调用这个方法的对象，如P1。可见同步方法实质是将synchronized作用于object reference。——那个拿到了P1对象锁的线程，才可以调用P1的同步方法，而对P2而言，P1这个锁与它毫不相干，程序也可能在这种情形下摆脱同步机制的控制，造成数据混乱。

2. 同步块，示例代码如下：`public void method3(SomeObject so) { synchronized(so) { //..... } }`这时，锁就是so这个对象，谁拿到这个锁谁就可以运行它所控制的那段代码。当有一个明确的对象作为锁时，就可以这样写程序，但当没有明确的对象作为锁，只是想让一段代码同步时，可以创建一个特殊的instance变量（它得是一个对象）来充当锁：`class Foo implements Runnable { private byte[] lock = new byte[0]. // 特殊的instance变量 Public void methodA() { synchronized(lock) { //... } } //..... }`注：零长度的byte数组对象创建起来将比任何对象都经济——查看编译后的字节码：生成零长度的byte[]对象只需3条操作码，而Object lock = new Object()则需要7行操作码。

3. 将synchronized作用于static函数，示例代码如下：`Class Foo { public synchronized static void methodAAA() // 同步的static 函数 { //.... } public void methodBBB() { synchronized(Foo.class) // class literal(类名称字面常量) } }`代码中的methodBBB()方法是把class literal作为锁的情

况，它和同步的static函数产生的效果是一样的，取得的锁很特别，是当前调用这个方法的对象所属的类（Class，而不再是由这个Class产生的某个具体对象了）。记得在《Effective Java》一书中看到过将 Foo.class和 P1.getClass()用于作同步锁还不一样，不能用P1.getClass()来达到锁这个Class的目的。P1指的是由Foo类产生的对象。可以推断：如果一个类中定义了一个synchronized的static函数A，也定义了一个synchronized的instance函数B，那么这个类的同一对象Obj在多线程中分别访问A和B两个方法时，不会构成同步，因为它们的锁都不一样。B方法的锁是Obj这个对象，而A的锁是Obj所属的那个Class。小结如下：搞清楚synchronized锁定的是哪个对象，就能帮助我们设计更安全的多线程程序。还有一些技巧可以让我们对共享资源的同步访问更加安全：1. 定义private的instance变量它的get方法，而不要定义public/protected的instance变量。百考试题提示如果将变量定义为public，对象在外界可以绕过同步方法的控制而直接取得它，并改动它。这也是JavaBean的标准实现方式之一。2. 如果instance变量是一个可变对象，特别数组或ArrayList什么的，那上述方法仍然不安全，因为当外界对象通过get方法拿到这个instance对象的引用后，又将其指向另一个对象，那么这个private变量也就变了，岂不是很危险。这个时候就需要将get方法也加上synchronized同步，并且，只返回这个private对象的clone()——这样，调用端得到的就是对象副本的引用了。或者也可以返回一个instance变量的副本引用（局部变量）。2009年上半年全国计算机等级考试参考答案请进入计算机考试论坛
2009年全国计算机等级考试报名信息汇总 2009年NCRE考试有

新变化 2009年全国计算机等级考试大纲 2009年上半年全国计算机二级考试试题及答案 2009年上半年全国计算机等级考试试题答案汇总 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com