

J DJ J

博客园 | 首页 | 发新随笔 | 发新文章 | 联系 | 订阅 XML | 管理

随笔：21 文章：0 评论：0 引用：0

可重入锁和不可重入锁

锁的简单应用

用lock来保证原子性（this.count++这段代码称为临界区）

什么是原子性，就是不可分，从头执行到尾，不能被其他线程同时执行。

可通过CAS来实现原子操作

CAS(Compare and Swap):

CAS操作需要输入两个数值，一个旧值（期望操作前的值）和一个新值，在操作期间先比较下旧值有没有发生变化，如果没有发生变化，才交换成新值，发生了变化则不交换。

CAS主要通过compareAndSwapXXX()方法来实现，而这个方法的实现需要涉及底层的unsafe类

unsafe类：java不能直接访问操作系统底层，而是通过本地方法来访问。Unsafe类提供了硬件级别的原子操作

这里有个介绍原子操作的博客

<https://my.oschina.net/xinxingegeya/blog/499223>

还有对unsafe类详解的博客

<http://www.cnblogs.com/mickole/articles/3757278.html>



```
1 public class Counter{
2     private Lock lock = new Lock();
3     private int count = 0;
4     public int inc(){
5         lock.lock();
6         this.count++;
7         lock.unlock();
8         return count;
9     }
10 }
```

<	2017年4月						>
日	一	二	三	四	五	六	
26	27	28	29	30	31	1	
2	3	4	5	6	7	8	
9	10	11	12	13	14	15	
16	17	18	19	20	21	22	
23	24	25	26	27	28	29	
30	1	2	3	4	5	6	

公告

昵称：smile_victor

园龄：2年5个月

粉丝：2

关注：0

+加关注

搜索

<input type="text"/>	找找看
<input type="text"/>	谷歌搜索

常用链接

- 我的随笔
- 我的评论
- 我的参与
- 最新评论
- 我的标签

随笔分类

- ios(1)
- javaWeb(2)
- linux(2)
- 基础应用(4)
- 算法(8)
- 一些项目开发的配置(3)

随笔档案

```
9     }  
10 }
```



不可重入锁

先来设计一种锁



```
1 public class Lock{  
2     private boolean isLocked = false;  
3     public synchronized void lock() throws InterruptedException{  
4         while(isLocked){  
5             wait();  
6         }  
7         isLocked = true;  
8     }  
9     public synchronized void unlock(){  
10        isLocked = false;  
11        notify();  
12    }  
13 }
```



这其实是个不可重入锁，举个例子



```
1 public class Count{  
2     Lock lock = new Lock();  
3     public void print(){  
4         lock.lock();  
5         doAdd();  
6         lock.unlock();  
7     }  
8     public void doAdd(){  
9         lock.lock();  
10        //do something  
11        lock.unlock();  
12    }  
13 }
```



当调用print()方法时，获得了锁，这时就无法再调用doAdd()方法，这时必须先释放锁才能调用，所以称这种锁为不可重入锁，也叫自旋锁。

可重入锁

设计如下:

- 2017年4月 (1)
- 2017年3月 (2)
- 2017年1月 (1)
- 2016年11月 (2)
- 2016年10月 (4)
- 2016年1月 (1)
- 2015年11月 (1)
- 2015年10月 (6)
- 2015年9月 (1)
- 2015年4月 (2)

阅读排行榜

- 1. html选择图片后预览，保存并上传(1894)
- 2. 可重入锁和不可重入锁(239)
- 3. map的应用(39)
- 4. 区间更新 zoj3911(28)
- 5. 线段树 hdu4046(28)

Powered by: 博客园

模板提供：沪江博客

Copyright ©2017 smile_victor



```
1 public class Lock{
2     boolean isLocked = false;
3     Thread   lockedBy = null;
4     int lockedCount = 0;
5     public synchronized void lock()
6         throws InterruptedException{
7         Thread thread = Thread.currentThread();
8         while(isLocked && lockedBy != thread){
9             wait();
10        }
11        isLocked = true;
12        lockedCount++;
13        lockedBy = thread;
14    }
15    public synchronized void unlock(){
16        if(Thread.currentThread() == this.lockedBy){
17            lockedCount--;
18            if(lockedCount == 0){
19                isLocked = false;
20                notify();
21            }
22        }
23    }
24 }
```



相对来说，可重入就意味着：线程可以进入任何一个它已经拥有的锁所同步着的代码块。

第一个线程执行print()方法，得到了锁，使lockedBy等于当前线程，也就是说，执行的这个方法的线程获得了这个锁，执行add()方法时，同样要先获得锁，因不满足while循环的条件，也就是不等待，继续进行，将此时的lockedCount变量，也就是当前获得锁的数量加一，当释放了所有的锁，才执行notify()。如果在执行这个方法时，有第二个线程想要执行这个方法，因为lockedBy不等于第二个线程，导致这个线程进入了循环，也就是等待，不断执行wait()方法。只有当第一个线程释放了所有的锁，执行了notify()方法，第二个线程才得以跳出循环，继续执行。

这就是可重入锁的特点。

java中常用的可重入锁

synchronized

java.util.concurrent.locks.ReentrantLock

ps:顺便记录下java中实现原子操作的类（记录至

<http://blog.csdn.net/huzhigenlaohu/article/details/51646455>）

- AtomicIntegerFieldUpdater:原子更新整型的字段的更新器
- AtomicLongFieldUpdater：原子更新长整型字段的更新器
- AtomicStampedReference:原子更新带有版本号的引用类型。该类将整型数值与引用关联起来，可用于原子的更新数据和数据的版本号，可以解决使用CAS进行原子更新时可能出现的ABA问题。

- AtomicReference : 原子更新引用类型
- AtomicReferenceFieldUpdater : 原子更新引用类型里的字段
- AtomicMarkableReference : 原子更新带有标记位的引用类型。可以原子更新一个布尔类型的标记位和应用类型
- AtomicIntegerArray : 原子更新整型数组里的元素
- AtomicLongArray : 原子更新长整型数组里的元素
- AtomicReferenceArray : 原子更新引用类型数组的元素
- AtomicBooleanArray : 原子更新布尔类型数组的元素
- AtomicBoolean : 原子更新布尔类型
- AtomicInteger : 原子更新整型
- AtomicLong : 原子更新长整型

好文要顶

关注我

收藏该文



smile_victor

关注 - 0

粉丝 - 2

0

0

+加关注

« 上一篇: 在Centos7.2(64位)下搭建Web服务器

» 下一篇: React.js开发的基本配置 (配了两天)

发表于 2017-03-19 15:54 SMILE_VICTOR 阅读(238) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【推荐】群英云服务器性价比王，2核4G5M BGP带宽 68元首月！

【福利】阿里云免费套餐升级，更多产品，更久时长



最新IT新闻:

- NASA打造“太空农场”：未来火星上或将花果飘香
- HUAWEI WATCH 2测评：独立通话 运动更自由

- 资本赏味下的网红美食如何避免昙花一现
- 风暴后的乐视体育：从高调激进到“埋头做事”
- 智能房屋充当全天候护士 实时检测人体健康水平
- » 更多新闻...



最新知识库文章:

- 唱吧DevOps的落地，微服务CI/CD的范本技术解读
- 程序员，如何从平庸走向理想？
- 我为什么鼓励工程师写blog
- 怎么轻松学习JavaScript
- 如何打好前端游击战
- » 更多知识库文章...