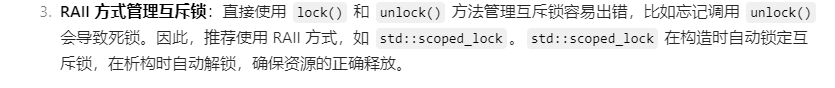
文本

AI 生成的内容可能不正确。

在并发编程中，临界区（Critical Section）是指一段代码，在这段代码中，程序会访问和操作共享资源（如共享内存、文件、数据库连接等），而这些操作在多线程或多进程环境下如果不加以控制，就可能会引发数据竞争（Data Race）和其他并发问题。

当使用互斥锁来保护共享资源时，存在陷入称为死锁的状态的风险。当两个线程都在等待对方释放锁时，就会发生死锁。两个线程都无法继续，它们陷入了死锁状态。发生死锁需要满足的一个条件是，一个已经持有锁的线程试图获取另一个锁。当系统变得越来越大时，跟踪系统中运行的所有线程可能使用的所有锁变得越来越困难。这是始终尽量减少共享资源使用的原因之一，这也说明了排他锁的必要性。



手机屏幕截图

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。