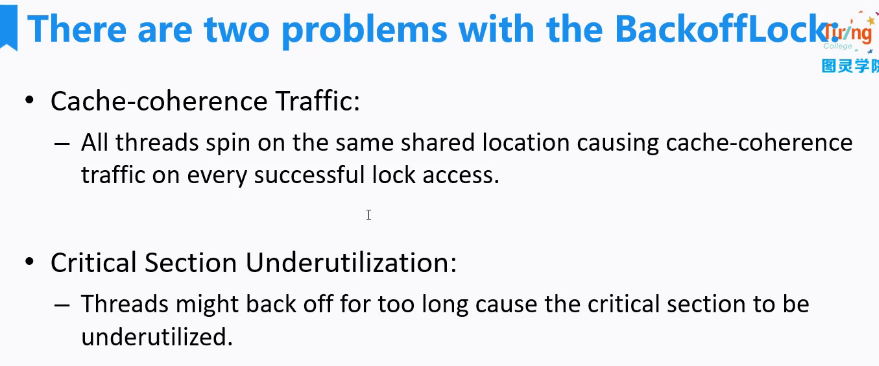
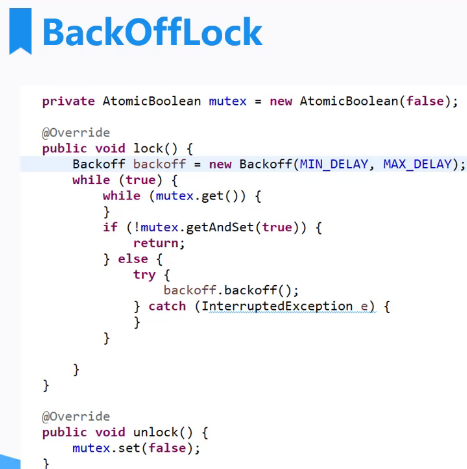
# TTAS锁：

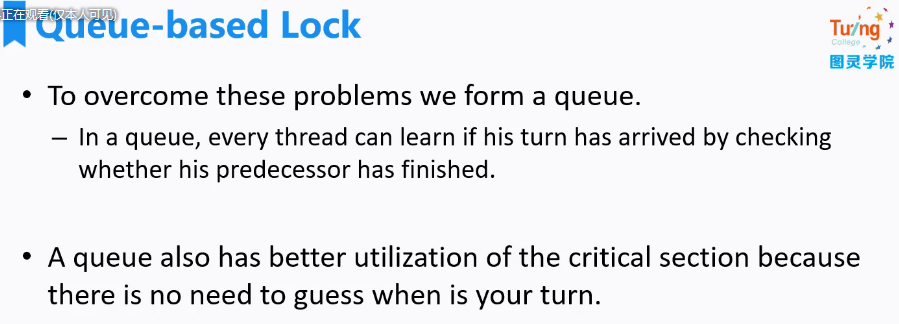
# BackoffLock锁的问题：

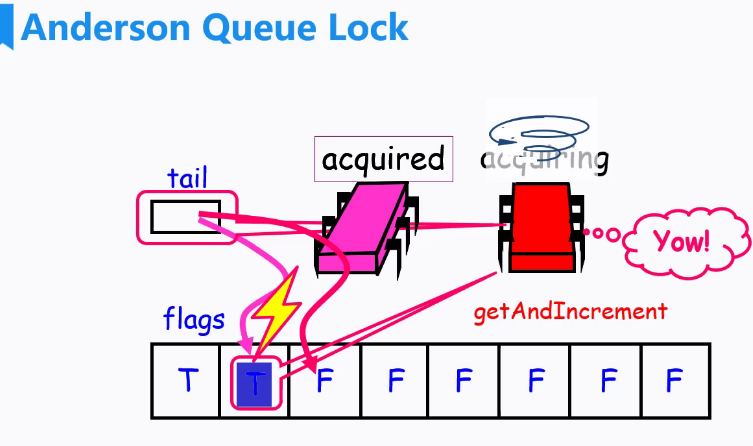


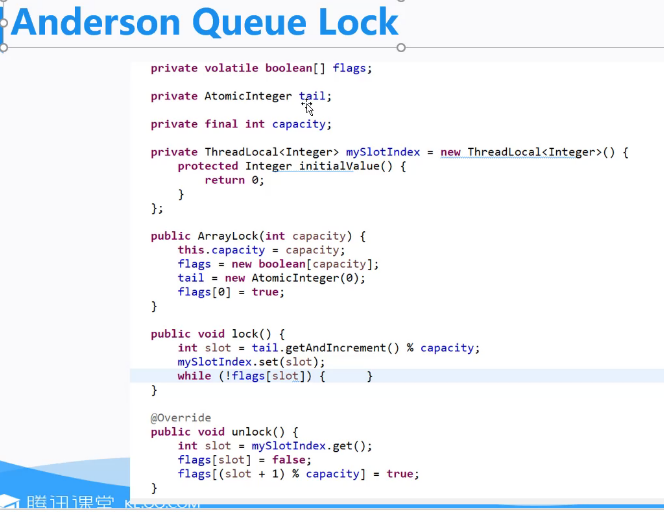


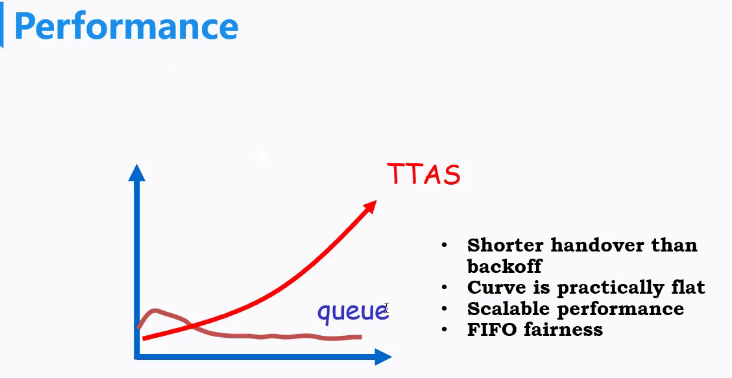
# Queue-based Lock：

JDK中使用的就是Queue-based Lock锁。

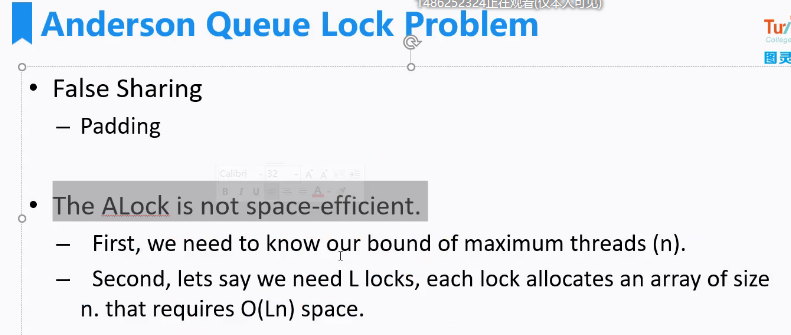






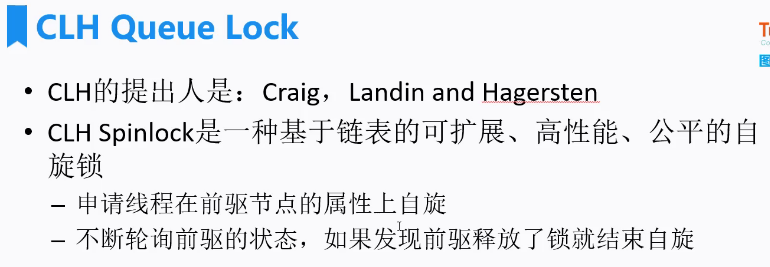


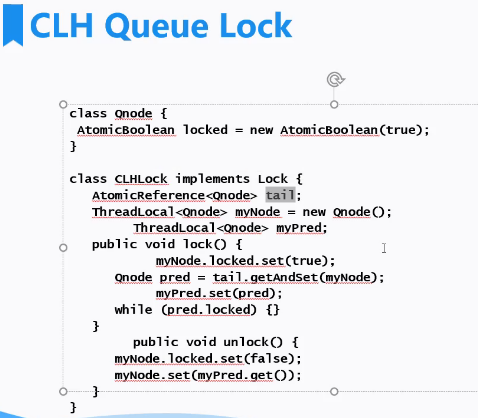
空间受限。



# CLH Queue Lock

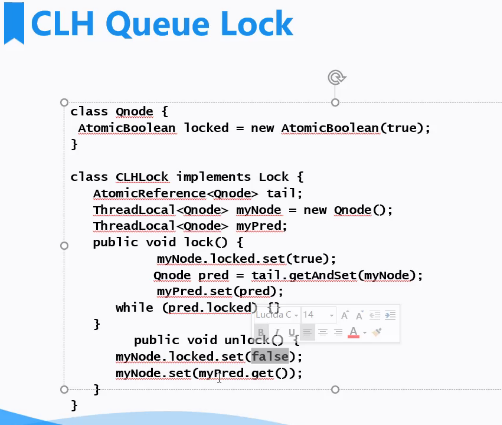
基于链表实现

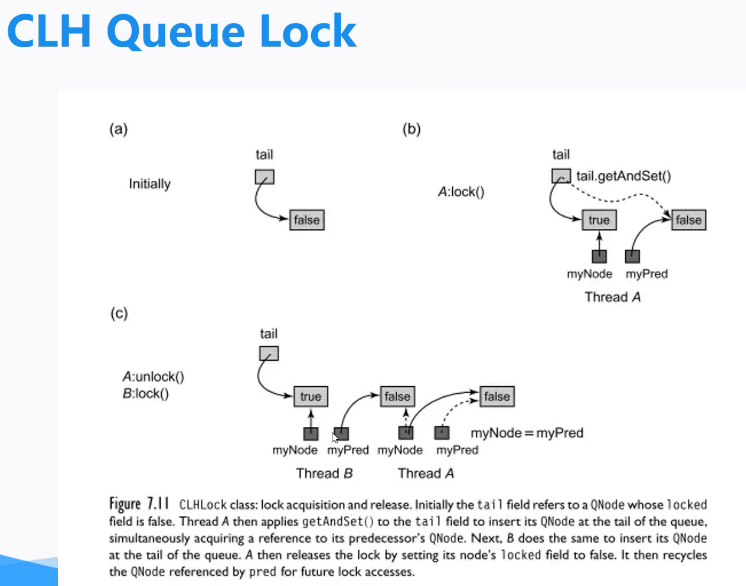




伪链表：

不是通过pre或next实现的，而是通过ThreadLocal保存两个变量，一个指向当前的Node，一个指向preNode。

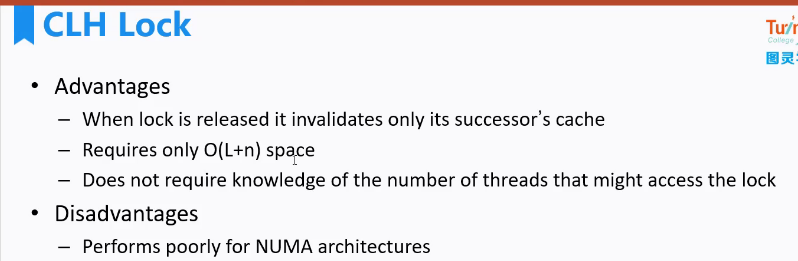




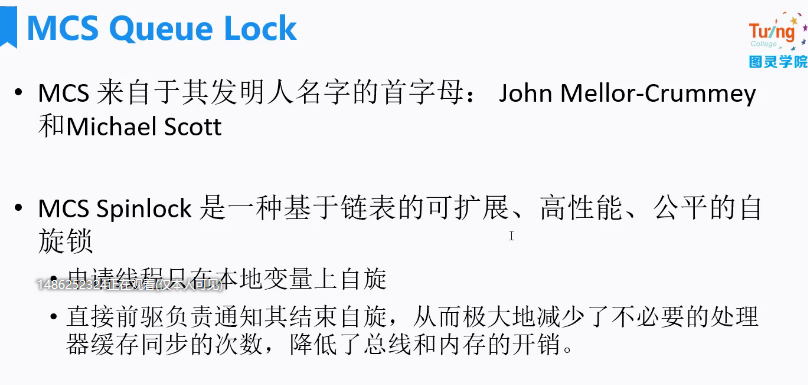
抢夺Tail：利用AtomicReference。

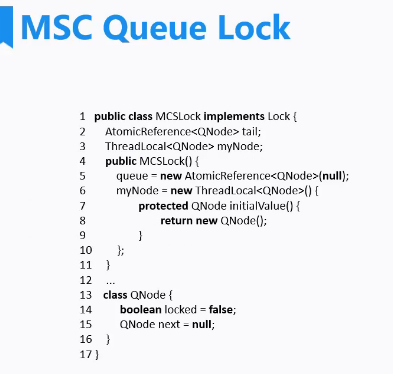
抢夺了Tail之后，就可以在尾部插入自己对应的true或false。

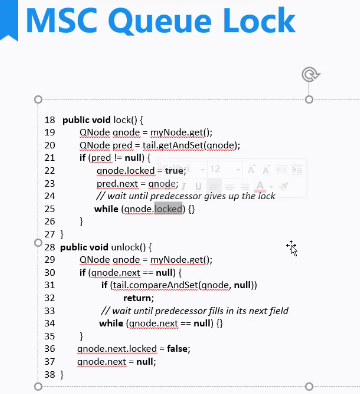
**始终轮询查看前一个节点的状态**。

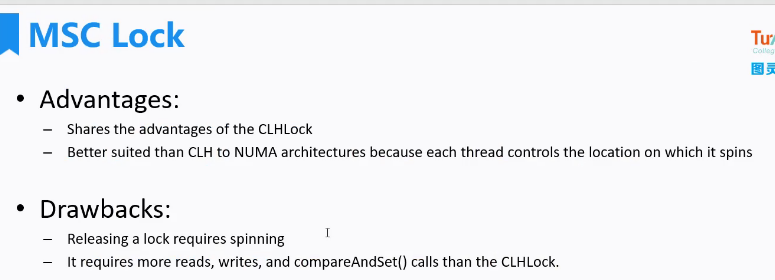


# MCS Queue Lock



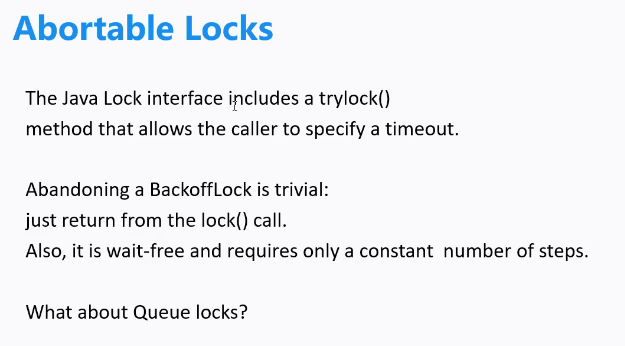




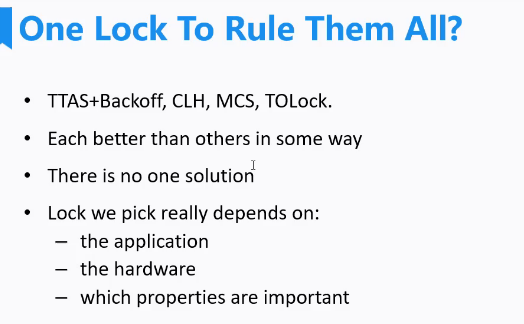


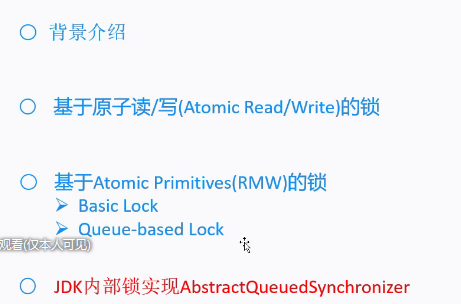
SMP

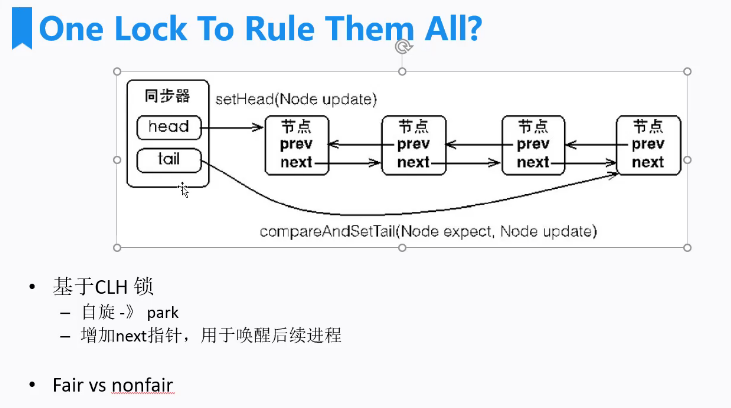
# Abortable Locks：可放弃的Lock，如tryLock方法



# 没有最好的！







**不公平实现原理**：在加入队尾之前，与队头的先抢一次，如果抢到就执行，否则就老老实实插入到队尾。

