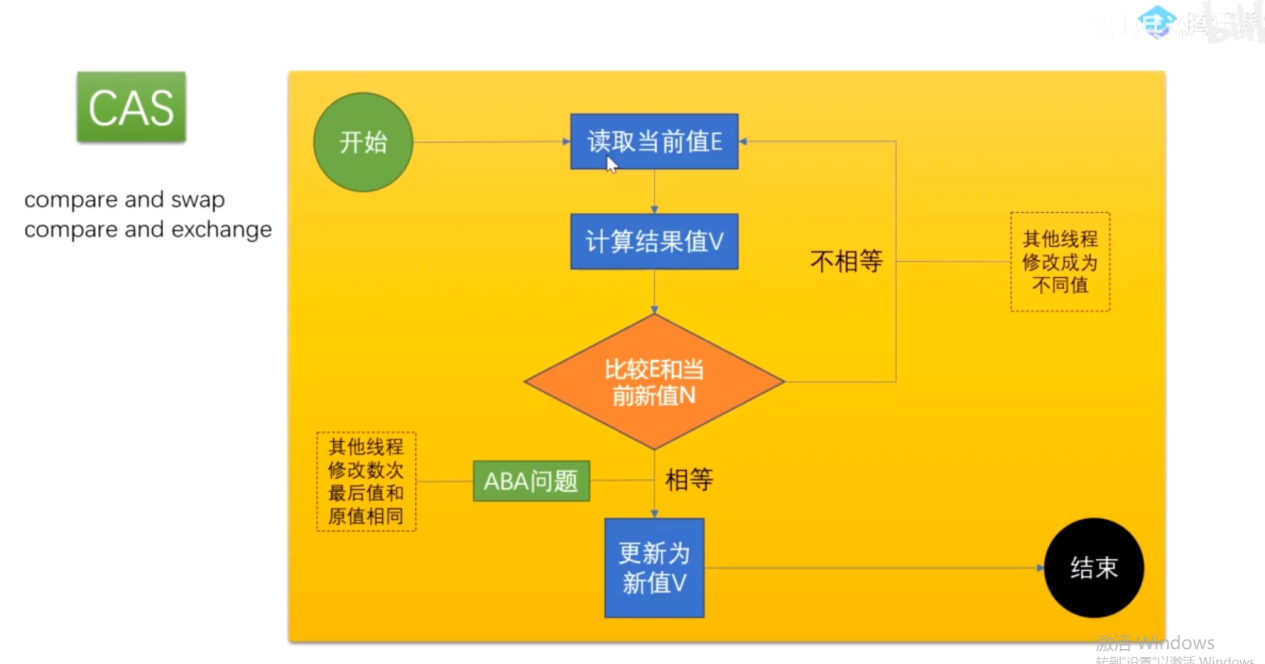


1. CAS（无锁优化、自旋锁、乐观锁：乐观的认为值不会被改变）

（1）原理：

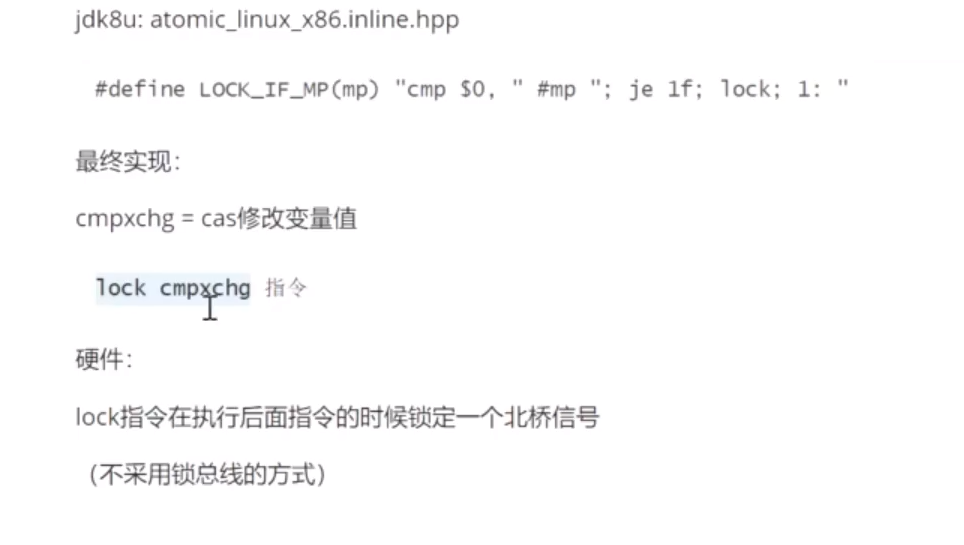
* 顶层：

从内存中读取值V到CPU中为E，进行运算得到N，然后写回内存，在此之前判断V与E是否还相等，相等（没被人改过）则写回内存，不相等（被人改过）重新读取值运算。



* 底层

lock cmpxchg指令



1. CAS问题：ABA

例子1： 假设有一个变量 A ，修改为B，然后又修改为了 A，实际已经修改过了，但 CAS 可能无法感知，造成了不合理的值修改操作。

例子2： 引用R指向一个对象，一个线程把R指向的对象改了，另外一个线程发现引用R没变，CAS就无法感知到R指向对象的变化。

如果是基本类型，无所谓

如果是引用类型，可能就会出现问题

解决方法：

1. 加版本号
2. 加标签