

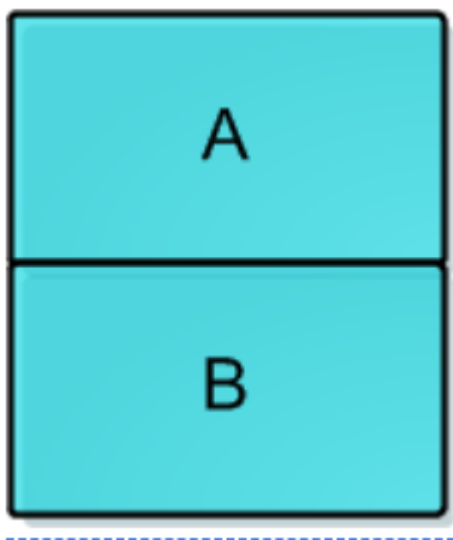
CAS看起来很好，但是会导致ABA问题出现。

CAS算法实现一个重要前提需要取出内存中某时刻的数据，而下时刻比较并替换，那么在这个时间差内会导致数据的变化。

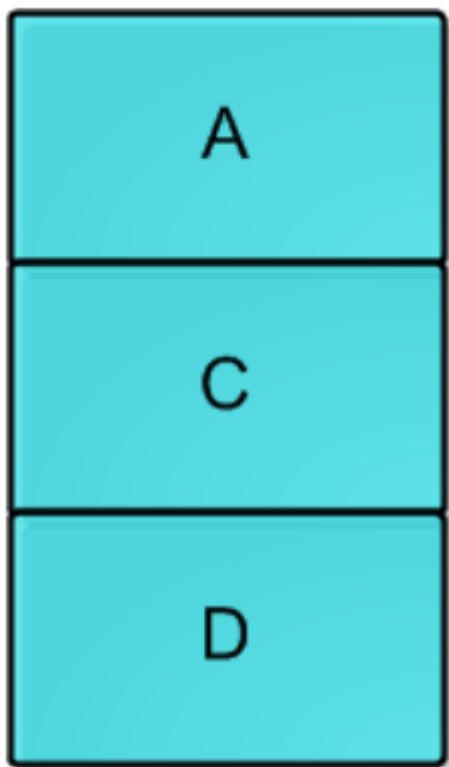
比如说一个线程one从内存位置V中取出A，这时候另一个线程two也从内存中取出A，并且two进行了一些操作变成了B，然后two又将V位置的数据变成A，这时候线程one进行CAS操作发现内存中仍然是A，然后one操作成功。尽管线程one的CAS操作成功，但是不代表这个过程就是没有问题的。

举例如下：

线程1准备用CAS将变量的值由A替换为B，在此之前，线程2将变量的值由A替换为C，又由C替换为A，然后线程1执行CAS时发现变量的值仍为A，所以CAS成功。但实际上这时的线程于最初不同了，尽管CAS成功，但可能存在潜藏的问题。



现有一个用单向链表实现的堆栈，栈顶为A，这时线程T1已经知道A.next为B，在T1执行上面的这条指令之前，线程T2介入，将A、B出栈，再push D、C、A，此时堆栈结构如下图，而对象B此时处于游离状态。



此时轮到线程T1执行CAS操作，检测发现栈顶仍为A，所以CAS成功，栈顶变为B，但实际上B.next为null，所以此时的情况变为其中栈顶中只有B一个元素，C和D组成的链表不再存在堆栈中，平白无故就把C、D丢掉了

