

18.1

如果2PL不能确保序列化，那么一定存在一个满足2PL协议的事务集合 T_0, T_1, \dots, T_{n-1} 产生了非序列化的计划，这意味着在前驱图中有一个环。我们可以证明2PL不会产生环。

假设前驱图中存在一个环： $T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow \dots \rightarrow T_{n-1} \rightarrow T_0$ ， α_i 为 T_i 获取它最后的锁的时间，因此，对于所有的成对事务 $T_i \rightarrow T_j$ ，有 $\alpha_i < \alpha_j$ 。因此对于该环有 $\alpha_0 < \alpha_1 < \dots < \alpha_{n-1} < \alpha_0$ ，显然 $\alpha_0 < \alpha_0$ 是不可能的，所以2PL无法产生非序列化的执行计划。因为对于所有如 $T_i \rightarrow T_j$ ， $\alpha_i < \alpha_j$ 的事务，事务的lock point 排序也是前驱图的拓扑排序，因此，事务可以按照lock point进行序列化。

18.7

a. 如果事务之间存在针对同一数据的 I 锁，这些自增操作的事务是可以互换共享的，就像读操作一样，但是如果有事务的操作互斥，那么按照 lock point 的先后顺序就可对事务进行排序，使其达到序列化。

b. 自增锁与自身可兼容，就允许了多个自增的同时获得锁，增加事务的并发性。如果没有自增锁，要想实现必须使用互斥锁，由于互斥，操作的锁等待时间就增加了，阻碍了并发的进行。

18.18

首先是实现较为简单，其次由于无级联调度，回滚的开销较低；最后它也允许一定的并发处理。