

## 第 11 章

1. 利用分时的方法同时处理多个事务，称为事务的并发调度
2. 当多个事务并发执行的结果与多个事务按任意次序串行执行的结果相同，就认为该并发操作是正确的
3. 封锁粒度越大，可以同时进行的并发操作越少，系统的并发程度越低。
4. 并发事务管理模块负责控制并发事务的交替执行，防止并发执行可能导致破坏数据库的一致性。 1
5. 事务 T1 按一定条件从数据库中读取了某些数据记录后，事务 T2 删除了其中部分记录，当 T1 再次按相同条件读取数据时，发现某些记录消失了。这种现象被称为读“脏”数据。
6. 若事务 T 对数据对象 A 加上 X 锁，则其它任何事务都不能再对 A 加任何类型的锁。
7. 若事务 T 对数据对象 A 加上 S 锁，则其它任何事务都不能再对 A 加任何类型的锁。 2
8. 封锁技术可以有效地解决并行操作的一致性问题，但也可能造成死锁。
9. 当一个事务申请在一个数据对象上加写锁时，总是有读锁和新的读锁的加入。导致总是加不上，这种现象称为死锁。
10. 顺序封锁法就是一次封锁法，只是名称不同。
11. 对于同一个事务集，若存在调度 S1 和 S2，在数据库的任何初始状态下，所有读出的数据都是一样的，留给数据库的最终状态也是一样的，则称 s1 和 s2 是等价的。
12. 并发控制最基本的正确性原则是可串行化。
13. 如果调度 S 通过调换冲突操作的次序得到调度 S'，称 S 和 S' 是冲突等价。
- 一个调度是冲突可串行化，则一定是可串行化的调度。
14. 一个可串行化的调度，一定是冲突可串行化。
15. 任何冲突可串行化调度都是视图可串行化调度。
16. 任何视图可串行化调度都是冲突可串行化调度。
17. 使用两段封锁协议产生的调度是可串行化调度。
18. 使用两段封锁协议产生的调度不会出现死锁现象。
19. 若并发事务的一个调度是可串行化的，则所有事务都符合两段锁协议。
20. 在多粒度封锁中一个数据对象可能以两种方式封锁：显式封锁和隐式封锁。
21. 系统检查封锁冲突时，只要检查显式封锁，不需要检查隐式封锁。
22. 引进意向锁的目的是提高对某个数据对象加锁时系统的检查效率。
23. 引进意向锁的目的是提高对某个数据对象加锁的效率。
24. 如果对一个数据对象加 IX 锁，表示它的上层结点拟（意向）加 X 锁。
25. 如果对一个数据对象加 SIX 锁，表示对它的下层节点加 S 锁和 X 锁。
26. 数据的一致性也取决于并发控制的程度。施加的并发控制愈多，数据的一致性往往愈好。