Subhajit Sahu
2018801013

26.10.2020

TUTORIAL

10.1 Explain why the fine-grained locking algorithm is not subject to deadlock.

A both add() and remove() methods aqquire node locks in one-after the other order (max. 2 at a time). hence, no deadlock. if however the add() and remove() acquired locks in different order, there could be deadlocks.

10.3 explain why the optimistic and lazy locking algorithms are not subject to deadlock.

A optimistic list delays the acquiring of locks until it finds the 2 nodes (adjacent) it wants to work with. then it locks them, and verifies that they are still part of the list. since it still aquires the 2 locks in one-after-other order, there is no deadlock.

lazy list is very similar, except it has another "marked" field to indicate nodes marked for removal. acquiring of locks is again in one-after-other order. hence, no deadlock.