

Locks (Ch 28)

Discussion Questions

- Why do we need locks?
- What are the possible states of a lock?
- What are the semantics of a call to lock()?
- What are the semantics of a call to unlock()?
- What three metrics do we use to measure the quality of a lock?
 - correctness
 - fairness
 - performance
- What are the drawbacks of implementing locks by disabling interrupts?
 - OS loses control
 - does not work with multiprocessors
 - slow to disable interrupts

- What is spin waiting?
- How does a spin lock work?
 - What does it require from the hardware?
 - test-and-set or compare-and-swap
 - What are the drawbacks?
 - starvation possible
 - wasted CPU resources
- How do test-and-set and compare-and-swap differ?
- What is the advantage of ticket locks over simple spin-locks?
 - no starvation
- Why does the code in Fig. 28.8 (locks with queues) test-and-set *guard* instead of *flag*?
- Why is spin waiting in the QueueLock better than spin waiting in general?

- limited to spin waiting on lock that guards the
queue, not the general lock
- What does park & unpark do in Solaris?
- What would happen if we released the guard lock after
the park(...) in QueueLock?
- Could we use a compare-and-swap in QueueLocks
instead of test-and-set?
- What is the advantage of two-phase locks?