Exercise for Process Synchronization

3180103570 卢佳盈

6.4 Explain why spinlocks are not appropriate for single-processor systems yet are often used in multiprocessor systems.

Only executing a different process can create the condition to break the original process out of the spinlock, so the spinlock can not be appropriated for single-processor systems. In single-processor, the process cannot relinquish the processor with spinlocks, then other processes cannot get the opportunity to set the condition to set such program condition for the original process to make any progress.

But in multiprocessor, other processes can execute on other processors, which can modify the program state and release the original process from the spinlock.

6.5 Explain why implementing synchronization primitives by disabling interrupts is not appropriate in a single-processor system if the synchronization primitives are to be used in user-level programs.

If a user-level program can disable interrupts, it can disable the timer interrupt and prevent context switching from taking place, allowing itself to use the processor without letting other processes to execute.

6.9 Show that, if the wait() and signal() semaphore operations are not executed atomically, then mutual exclusion may be violated.

If the two operations are not performed atomically, then it is possible that both operations might proceed to decrement the semaphore value, then it will violate the mutual exclusion.