

Styresystemer og Multiprogrammering

G-opgave #3

Jenny-Margrethe Vej
(rwj935@alumni.ku.dk)
Klaes Bo Rasmussen
(twb822@alumni.ku.dk)

Datalogisk Institut, Københavns Universitet
Blok 3 - 2013

1 Locks and condition variables for kernel threads in Buenos

1. This was our understanding of the first task, it may be wrong, but we would really like some feedback as to how it's wrong, perhaps just verbal feedback.

`lock_reset`

This resets the spinlock and the locked variable in the given lock.

`lock_acquire`

Saves the interrupt status, setting it to disabled. Then acquires spinlock for the given lock. It runs and keeps checking if the lock is still locked. If so, it adds the lock to the sleep queue, releases the spinlock and switches thread. When it is no longer locked, it locks it again so its locked to itself and the code can run without disruptions.

`lock_release`

Disables interrupts, saving the status. Then acquires the spinlock. If lock is no longer locked, wake it up and continue to release spinlock and set the interrupt status back to its former glory.

2. This is how we understood the conditional statements, but reading at people's reaction to giving lock as an argument to condition signal makes it seem this could be wrong, we hope to get a better explanation for this part, the code is pretty self explanatory, as short as it is.

2 One-Lane Synchronisation

1. When synchronising a stretch of road like this, it would be preferable to account for different times of day when traffic amount changes. But if we consider a simple solution:

Let the first car that gets to the crossing be put into the queue first, let it pass. If no other cars get to the crossing while the first is passing it, reset the queue when the car exits.

If more cars come, open up for new cars going in the same direction. When a car approaches from the other direction, set a timer of 1 minute before cars need to stop from the first direction, and open up for the other direction, given the same rules. Again, if there are no cars as the last crossing car leaves the narrow path, reset the queue.

2. We did not manage to complete this part of the task yet.

3. And not this either.