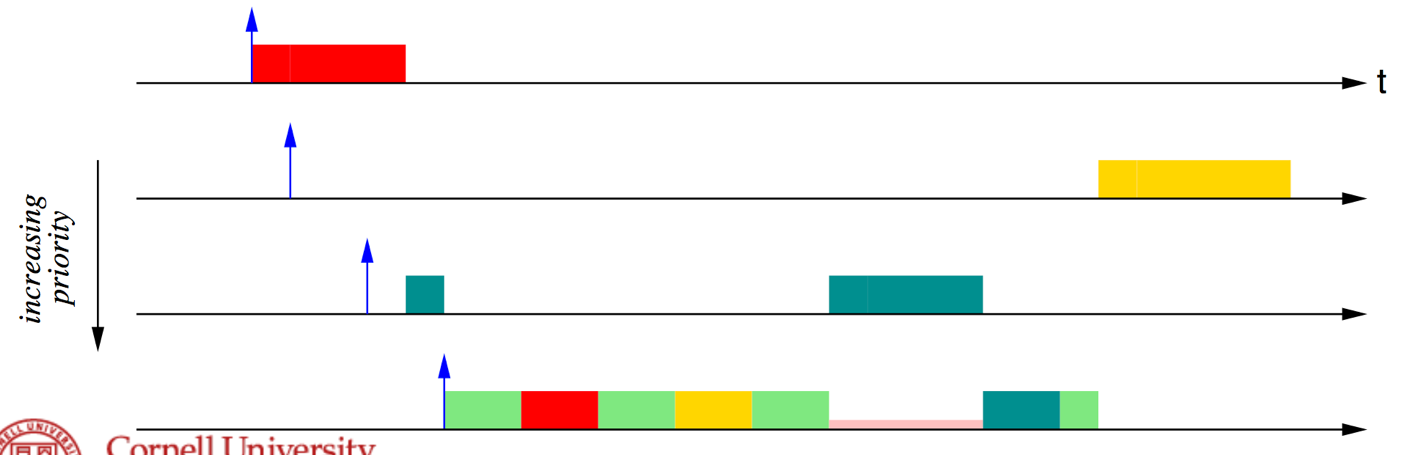
2. 

The error occurs in the circled region. Note, that each lock is assigned the same ceiling, in this example 3, as process 4 uses all three locks and has priority 3. Hence, the priority of the red CS in process 4 is no greater than the priority of the blue CS in process 3, as p4 > max{C(lblue) | lblue is locked by tasks ≠ τ4}. Thus, red cannot pre-empt blue, so in process 4, after it finishes its first green section, P3 will then be allowed to finish its blue CS, then it will switch back to P4 to continue processing.

Overall, P1 starts executing. P2’s and P3’s priority is not higher than the ceilings of the active locks (P1’s lock). P3 runs after P1 finishes its CS because after P1 finishes there are no active CS and P3 executes as it has a higher priority than P2. In P4, after it finishes its first green section, P3 will then be allowed to finish its blue CS, then it will switch back to P4 to continue processing. Once P4 gets to it’s yellow CS