7.1) $\tau_1 = (4, 11)$ $\tau_2 = (5, 16)$ $\tau_3 = (3, 31)$

$U = 4/11 + 5/16 + 3/31 = 0.77 < U(3) = 0.78$.

By UB test, they are schedulable $P_{m_1} = 1$

$\tau_1 : (P(m_1), V(m_1))$

$\tau_3 : (P(m_1), V(m_1))$

$\tau_1 \Rightarrow EEMM$

$\tau_2 \Rightarrow EEEEE$

$\tau_3 \Rightarrow MMM$

No missed deadlines

{See image attached}

2) $\tau_1 = (3, 7)$ $\tau_2 = (1, 9)$ $\tau_3 = (6, 26)$

$U = 3/7 + 1/9 + 6/26 = 0.77 < U(3) = 0.78$

By UB test, they are schedulable

$\tau_1 : (P(m_1), V(m_1))$

$\tau_2 : (P(m_2), V(m_2))$

$\tau_3 : (P(m_1), V(m_1), P(m_2), V(m_2))$

$P_{m_1} = 1$

$P_{m_2} = 2$

$\tau_1 \Rightarrow EM, M_1$

$\tau_2 \Rightarrow M_2$

$\tau_3 \Rightarrow M_1 M_1 M_{12} M_{12} M_2$
M_2

τ_2 misses deadline

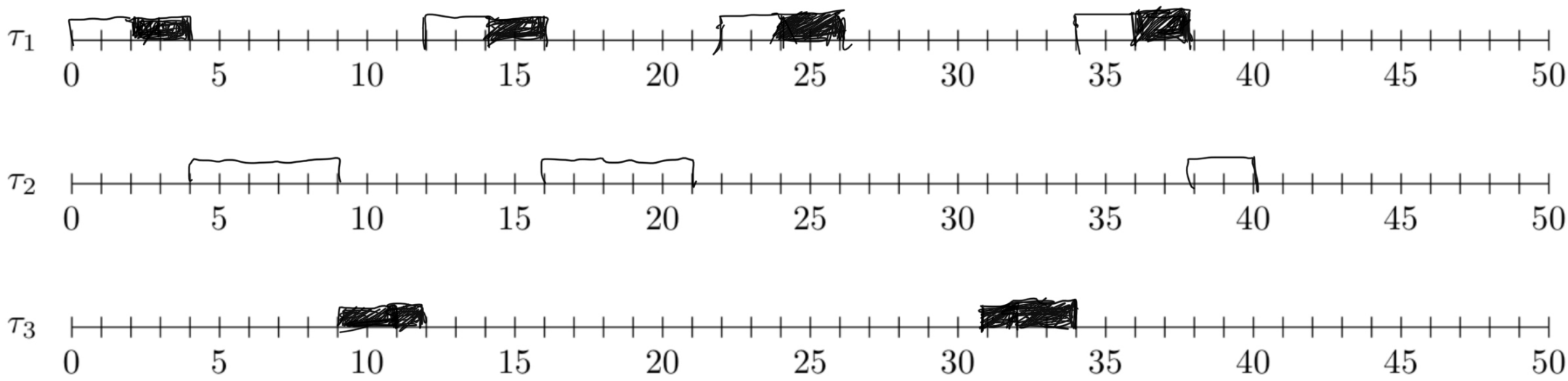
{see image attached

7:31 PM Thu Apr 15 17%
github.com

Unfortunately, we can't always run our tasks in isolation. We need to share resources between each o protect this data from corruption with the following scheme:

- τ_1 uses mutex m_1 for the **last two ticks** of its computation time.
- τ_3 uses mutex m_1 for all three ticks of its computation time.

Please draw the expected scheduling using IPCP of this task set with the mutexes on the timelines $t = 40$. Assume that locks and unlocks can be done instantaneously (0 ticks). Indicate in which ticks holding a mutex by shading that tick in. Does any task miss its deadline?



7:31 PM Thu Apr 15
github.com
17%

- τ_1 uses mutex m_1 for the **last two ticks** of its computation time.
- τ_2 uses mutex m_2 for all of its computation time (lock at the beginning of the tick, unlock at the tick).
- τ_3 uses mutex m_1 for the **first four ticks** of its computation time, and mutex m_2 for the **last fo** its computation time. (There will be two ticks when both mutexes are locked.)

Please draw the expected scheduling using IPCP of this task set with the mutexes on the timelines below t Assume that locks and unlocks can be done instantaneously (0 ticks). Indicate in which ticks each task mutex by shading that tick in. Does any task miss its deadline?