MLQ - Priority donation (simple example)

- I. TI (low priority) starts, runs and acquires the lock 1
- 2. T2 (medium priority) starts, preempts the CPU and runs
- 3. T3 (high priority) starts, preempts the CPU, runs but gets blocked while trying to acquire the lock 1
- 4. T3 gives its high priority to TI
- 5. TI (now high priority) runs, releases the lock and returns to low priority immediately after
- 6. T3 (now unblocked) preempts the CPU and runs

Solutions to other MLQ problems

- → To prevent starvation of low priority thread change the priority over time by either
 - · increase priority as a function of waiting time
 - or decrease priority as a function of CPU consumption
- → To decide on the priority
 by observing and keeping track of the thread CPU usage