

Another Synchronization Construct

Semaphore

An abstract data type to provide mutual exclusion described by *Dijkstra* in the "*THE multiprogramming system*" in 1968

➔ Semaphores are “integers” that support two operations:

- Semaphore::P() decrement, block until semaphore is open
a.k.a wait(), or sem_wait(), or sema_down()
- Semaphore::V() increment, allow another thread to enter
a.k.a signal(), or sem_post(), or sema_up()

✓ Semaphore safety property
the semaphore value is always greater than or equal to 0

Blocking mechanism

Associated with each semaphore is a queue of waiting threads

➡ When $P()$ is called by a thread:

- If semaphore is open, thread continue
- If semaphore is closed, thread blocks on queue

➡ Then $V()$ opens the semaphore

- If a thread is waiting on the queue, the thread is unblocked
- If no threads are waiting on the queue, the signal is remembered for the next thread