Yingnan Zhao
260563769

1. Conceptually, mutex and semaphores are different. A mutex is used to lock critical sections, to restrict the access of shared variable. Semaphore is like a signaling mechanism, for example to keep track of item produced and consumed. If we were to only use semaphore, the shared variable for example an array list is not protected, even the number of item produced and consumed has been keeping track of. Both the consumer and the producer might make change to the shared list at the same time thus causing race condition. For example, the list is empty, and the producer is putting an item into the list, thus their empty semaphore is larger than 0 the consumer can proceed, the consumer is trying to get the item before the producer finishing putting it.
2. It is not possible, assume producer always have higher priority than the consumer. Even if there is a steady stream of high priority producer producing preventing the consumer from consuming. If there is an upper limit of the item that can produce that is being keeping tracked by empty and full semaphore, the producer has to wait on the empty semaphore at some point. If there is no empty slot it will not keep producing, thus the consumer can run.
3. Mutex has ownership. A mutex can only be released by the thread that has ownership to it (the thread that called wait on that mutex). However, semaphores can be signaled by any thread, which make it possible that a binary semaphore being locked by one thread and being unlocked by another.
4. Because, if we were to have a finite buffer to store the item the producer produces, full semaphore is used to block consumer if the buffer is empty, and empty semaphore is used to block producer if the buffer is full. If we only have the full semaphore, there is nothing stopping the producer from producing infinitely. If we try to fix this by calculating the empty value using only the Full value, this calculation needs to be put into the critical section that is protected by the mutex, thus the critical section gets longer.