OS 隨堂測驗:

1. (O)Race conditions are prevented by requiring that critical regions be protected by locks.
2. (X)The value of a counting semaphore can range only between 0 and 1.
3. (O)Mutex locks and binary semaphores are essentially the same thing.
4. (O)A nonpreemptive kernel is safe from race conditions on kernel data structures.
5. A race condition __**B**__.
   A) results when several threads try to access the same data concurrently
   B) results when several threads try to access and modify the same data concurrently
   C) will result only if the outcome of execution does not depend on the order in which instructions are executed
   D) None of the above
6. An instruction that executes atomically __**B**__.
   A) must consist of only one machine instruction
   B) executes as a single, uninterruptible unit
   C) cannot be used to solve the critical section problem
   D) All of the above
7. A counting semaphore __**A**__.
   A) is essentially an integer variable
   B) is accessed through only one standard operation
   C) can be modified simultaneously by multiple threads
   D) cannot be used to control access to a thread's critical sections
8. A mutex lock __**B**__.
   A) is exactly like a counting semaphore
   B) is essentially a boolean variable
   C) is not guaranteed to be atomic
   D) can be used to eliminate busy waiting
9. (**A**) What is the correct order of operations for protecting a critical section using mutex locks?
   A) release() followed by acquire()
   B) acquire() followed by release()
   C) wait() followed by signal()
   D) signal() followed by wait()
10. (**C**) A solution to the critical section problem does not have to satisfy which of the following requirements?
    A) mutual exclusion
    B) progress
    C) atomicity
    D) bounded waiting