OS 隨堂測驗 504

1. (**C**) The first readers-writers problem ____.
   A) requires that, once a writer is ready, that writer performs its write as soon as possible.
   B) is not used to test synchronization primitives.
   C) requires that no reader will be kept waiting unless a writer has already obtained permission to use the shared database.
   D) requires that no reader will be kept waiting unless a reader has already obtained permission to use the shared database.
2. (**A**) _____ occurs when a higher-priority process needs to access a data structure that is currently being accessed by a lower-priority process.
   A) Priority inversion
   B) Deadlock
   C) A race condition
   D) A critical section
3. (**B**) What is the correct order of operations for protecting a critical section using a mutex lock?
   A) release() followed by acquire()
   B) acquire() followed by release()
   C) wait() followed by signal()
   D) signal() followed by wait()
4. (**C**) What is the correct order of operations for protecting a critical section using a binary semaphore?
   A) release() followed by acquire()
   B) acquire() followed by release()
   C) wait() followed by signal()
   D) signal() followed by wait()
5. (**B**) How many philosophers may eat simultaneously in the Dining Philosophers problem with 5 philosophers?
   A) 1
   B) 2
   C) 3
   D) 5
6. (**B**) When using semaphores, a process invokes the wait() operation before accessing its critical section, followed by the signal() operation upon completion of its critical section. Consider reversing the order of these two operations—first calling signal(), then calling wait(). What would be a possible outcome of this?
   A) Starvation is possible.
   B) Several processes could be active in their critical sections at the same time.
   C) Mutual exclusion is still assured.
   D) Deadlock is possible.

7. (**A**) A(n) _____ refers to where a process is accessing/updating shared data.
   A) critical section
   B) entry section
   C) mutex
   D) test-and-set
8. (**C**) What is the purpose of the mutex semaphore in the implementation of the bounded-buffer problem using semaphores?
   A) It indicates the number of empty slots in the buffer.
   B) It indicates the number of occupied slots in the buffer.
   C) It controls access to the shared buffer.
   D) It ensures mutual exclusion.
9. __**X**__ A deadlock-free solution eliminates the possibility of starvation.
10. __**X**__Monitors are a theoretical concept and are not practiced in modern programming languages