1、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

2、A 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

3、In Peterson's solution, the (C) variable indicates if a process is ready to enter its critical section.

A) turn B) lock C) flag[i] D) turn[i]

4、A (D) type presents a set of programmer-defined operations that are provided mutual exclusion within it.

A) transaction B) signal C) binary D) monitor

5、A solution to the critical section problem does not have to satisfy which of the following requirements? (C)

A) mutual exclusion B) progress C) atomicity D) bounded waiting

6、A(n) (A) refers to where a process is accessing/updating shared data.

A) critical section B) entry section C) mutex D) test-and-set

7、(B) can be used to prevent busy waiting when implementing a semaphore.

A) Spinlocks B) Waiting queues

C) Mutex lock D) Allowing the wait() operation to succeed

8、What is the purpose of the mutex semaphore in the implementation of the bounded-buffer problem using semaphores? (D)

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、How many philosophers may eat simultaneously in the Dining Philosophers problem with 5 philosophers? (B)

A) 1 B) 2 C) 3 D) 5

10、Which of the following statements is true? (D)

A) A counting semaphore can never be used as a binary semaphore.

B) A binary semaphore can never be used as a counting semaphore.

C) Spinlocks can be used to prevent busy waiting in the implementation of semaphore.

D) Counting semaphores can be used to control access to a resource with a finite number of instances.

11、(T) Race conditions are prevented by requiring that critical regions be protected by locks.

12、(F) The value of a counting semaphore can range only between 0 and 1.

13、(F) A deadlock-free solution eliminates（消除） the possibility of starvation.

14、(T) The local variables of a monitor can be accessed by only the local procedures.

15、(F) Monitors are a theoretical concept and are not practiced in modern programming languages.

16、(F) Mutex locks and counting semaphores are essentially the same thing.

17、(T) Mutex locks and binary semaphores are essentially the same thing.

18、What three conditions must be satisfied in order to solve the critical section problem?

19、Explain the deadlock and give an example.

20、Explain starvation and give examples.