Started on	Tuesday, October 15, 2024, 7:19 PM
State	Finished
Completed on	Tuesday, October 15, 2024, 7:24 PM
Time taken	4 mins 39 secs
Points	8.00/9.00
Grade	**88.89** out of 100.00

Question 1

Correct

1.00 points out of 1.00

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?

Select one:

- 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.

Your answer is correct.

The correct answer is: Several processes could be active in their critical sections at the same time.

Question 2

Correct

1.00 points out of 1.00

An instruction that executes atomically ___.

Select one:

- 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

Your answer is correct.

The correct answer is: executes as a single, uninterruptible unit

Question 3

Correct

1.00 points out of 1.00

A solution to the critical section problem does not have to satisfy which of the following requirements?

Select one:

- a. progress
- b. atomicity ✔
- c. mutual exclusion
- d. bounded waiting

Your answer is correct.

The correct answer is: atomicity

Question 4

Correct

1.00 points out of 1.00

In Peterson's solution, the ___ variable indicates if a process is ready to enter its critical section.

Select one:

- a. turn[i]
- b. turn
- c. flag[i] ✔
- d. lock

Your answer is correct.

The correct answer is: flag[i]

Question 5

Correct

1.00 points out of 1.00

A ___ type presents a set of programmer-defined operations that are provided mutual exclusion within it.

Select one:

- a. monitor ✔
- b. transaction
- c. binary
- d. signal

Your answer is correct.

The correct answer is: monitor

Question 6

Correct

1.00 points out of 1.00

Assume an adaptive mutex is used for accessing shared data on a Solaris system with multiprocessing capabilities. Which of the following statements is not true?

Select one:

- a. A waiting thread may spin while waiting for the lock to become available.
- b. Condition variables and semaphores are never used in place of an adaptive mutex. ✔
- c. A waiting thread may sleep while waiting for the lock to become available.
- d. The adaptive mutex is only used to protect short segments of code.

Your answer is correct.

The correct answer is: Condition variables and semaphores are never used in place of an adaptive mutex.

Question 7

Correct

1.00 points out of 1.00

The first readers-writers problem ___.

Select one:

- a. requires that no reader will be kept waiting unless a reader has already obtained permission to use the shared database
- b. requires that no reader will be kept waiting unless a writer has already obtained permission to use the shared database ✔
- c. requires that, once a writer is ready, that writer performs its write as soon as possible
- d. is not used to test synchronization primitives

Your answer is correct.

The correct answer is: requires that no reader will be kept waiting unless a writer has already obtained permission to use the shared database

Question 8

Incorrect

0.00 points out of 1.00

Which of the following statements is true?

Select one:

- a. Counting semaphores can be used to control access to a resource with a finite number of instances. ✖
- b. Spinlocks can be used to prevent busy waiting in the implementation of semaphore.
- c. A counting semaphore can never be used as a binary semaphore.
- d. A binary semaphore can be used as a counting semaphore.

Your answer is incorrect.

The correct answer is: Spinlocks can be used to prevent busy waiting in the implementation of semaphore.

Question 9

Correct

1.00 points out of 1.00

____ can be used to prevent busy waiting when implementing a semaphore.

Select one:

- a. Waiting queues ✔
- b. Allowing the wait() operation to succeed
- c. Spinlocks
- d. Mutex lock

Your answer is correct.

The correct answer is: Waiting queues