Week 5b

1) The OS may preempt the second process and require it to release its resources if a process requests a resource that is currently held by another process.

**True**

2) Once the processes have progressed into the \_\_\_\_\_\_\_\_\_\_ , those processes will deadlock.

a. spinlock

b. hold and wait

**c. fatal region**

d. regional resources

3) The dining philosophers’ problem can be representative of problems dealing with the coordination of shared resources which may occur when an application includes concurrent threads of execution.

**True**

4) Requested resources are granted to processes whenever possible with \_\_\_\_\_\_\_\_\_ .

a. preemption

b. mutual exclusion

c. deadlock avoidance

**d. deadlock detection**

5) A mutex is used to ensure that only one thread at a time can access the resource protected by the mutex.

**True**

6) An unsafe state is one in which there is at least one sequence of resource allocation to processes that does not result in a deadlock.

**False**

7) Deadlock avoidance is more restrictive than deadlock prevention.

**False**

8) All deadlocks involve conflicting needs for resources by two or more processes.

**True**

9) A set of processes is \_\_\_\_\_\_\_\_\_ when each process in the set is blocked awaiting an event that can only be triggered by another blocked process in the set.

a. preempted

b. stagnant

**c. deadlock**

d. spinlocked

10) The \_\_\_\_\_\_\_\_\_\_ condition can be prevented by defining a linear ordering of resource types.

a. hold and wait

b. no preemption

**c. circular wait**

d. mutual exclusion

11) For deadlock to occur, there must not only be a fatal region, but also a sequence of resource requests that has led into the fatal region.

**True**

12) The \_\_\_\_\_\_\_\_\_ condition can be prevented by requiring that a process request all of its required resources at one time and blocking the process until all requests can be granted simultaneously.

a. no preemption

**b. hold and wait**

c. mutual exclusion

d. circular wait

13) With \_\_\_\_\_\_\_\_\_ only one process may use a resource at a time and no process may access a resource unit that has been allocated to another process.

a. hold and wait

b. circular wait

**c. mutual exclusion**

d. no preemption

14) Interrupts, signals, messages, and information in I/O buffers are all examples of reusable resources.

**False**

15) Deadlock is permanent because none of the events are ever triggered.

**True**

16) Examples of \_\_\_\_\_\_\_\_\_\_ include processors, I/O channels, main and secondary memory, devices, and data structures such as files, databases, and semaphores.

a. consumable resources

b. joint resources

**c. reusable resources**

d. regional resources

17) Deadlock avoidance requires knowledge of future process resource requests.

**True**

18) The strategy of deadlock \_\_\_\_\_\_\_\_\_ is to design a system in such a way that the possibility of deadlock is excluded.

a. avoidance

**b. prevention**

c. diversion

d. detection

19) A closed chain of processes exists, such that each process holds at least one resource needed by the next process in the chain is the condition of \_\_\_\_\_\_\_\_\_\_ .

a. no preemption

b. hold and wait

**c. circular wait**

d. mutual exclusion

20) An indirect method of deadlock prevention is to prevent the occurrence of a circular wait.

**False**