**Chapter 6 – Concurrency: Deadlock and Starvation**

**TRUE/FALSE QUESTIONS:**

T F 1)  Deadlock is permanent because none of the events are ever triggered.

T F 2)  All deadlocks involve conflicting needs for resources by two or more processes.

T F 3)  Interrupts, signals, messages, and information in I/O buffers are all examples

of reusable resources.

T F 4)  A useful tool in characterizing the allocation of resources to processes is the

resource allocation graph.

T F 5)  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.

T F 6)  An indirect method of deadlock prevention is to prevent the occurrence of a

circular wait.

T F 7)  If access to a resource requires mutual exclusion then mutual exclusion must be

supported by the OS

T F 8)  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.

T F 9)  Deadlock avoidance requires knowledge of future process resource requests.

T F 10)  An unsafe state is one in which there is at least one sequence of resource allocations

to processes that does not result in a deadlock.

T F 11)  An atomic operation executes without interruption and without interference.

T F 12)  Deadlock avoidance is more restrictive than deadlock prevention.

T F 13)  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.

T F 14)  A signal is similar to a hardware interrupt but does not employ priorities.

T F 15)  A mutex is used to ensure that only one thread at a time can access the resource

protected by the mutex.

**MULTIPLE CHOICE QUESTIONS:**

1)  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)  spinlocked   B)  stagnant

C)  preempted   D)  deadlocked

2)  Examples of \_\_\_\_\_\_\_\_\_\_ include processors, I/O channels, main and secondary memory, devices,

and data structures such as files, databases, and semaphores.

A)  regional resources   B)  joint resources

C)  reusable resources   D)  consumable resources

3)  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)  mutual exclusion

C)  no preemption   D)  circular wait

4)  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)  mutual exclusion

C)  circular wait   D)  hold and wait

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

A)  fatal region   B)  regional resources

C)  spinlock   D)  hold and wait

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

deadlock is excluded.

A)  prevention   B)  detection

C)  diversion   D)  avoidance

7)  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)  mutual exclusion   B)  hold and wait

C)  circular wait   D)  no preemption

8)  The fastest form of interprocess communication provided in UNIX is \_\_\_\_\_\_\_\_\_\_ .

A)  shared memory   B)  message

C)  pipe   D)  semaphore

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

A)  hold and wait   B)  no preemption

C)  mutual exclusion   D)  circular wait

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

A)  preemption   B)  deadlock detection

C)  mutual exclusion   D)  deadlock avoidance

11)  One of the most significant contributions of UNIX to the development of operating

systems is the \_\_\_\_\_\_\_\_\_ .

A)  semaphore   B)  shared memory

C)  message   D)  pipe

12)  A \_\_\_\_\_\_\_\_\_\_ is a software mechanism that informs a process of the occurrence of

asynchronous events.

A)  signal   B)  message

C)  mailbox   D)  kernel

13)  The most common technique used for protecting a critical section in Linux is the \_\_\_\_\_\_\_ .

A)  signal   B)  atomic bitmap operation

C)  atomic integer operation   D)  spinlock

14)  The \_\_\_\_\_\_\_\_\_ allows multiple threads to have simultaneous read-only access to an

object protected by the lock.

A)  barrier   B)  condition variable

C)  readers/writer lock   D)  mutex

15)  The \_\_\_\_\_\_\_\_\_ is useful in sending a signal to a thread indicating that a particular event

has occurred.

A)  mutex object   B)  semaphore object

C)  event object   D)  waitable timer object

**SHORT ANSWER QUESTIONS:**

1)  \_\_\_\_\_\_\_\_\_ can be defined as the *permanent* blocking of a set of processes that either compete

for system resources or communicate with each other.

2)  A \_\_\_\_\_\_\_\_\_ resource is one that can be safely used by only one process at a time and is not

depleted by that use.

3)  A \_\_\_\_\_\_\_\_\_ resource is one that can be created and destroyed.

4)  The \_\_\_\_\_\_\_\_\_ is a directed graph that depicts a state of the system of resources and processes,

with each process and each resource represented by a node.

5)  Three conditions of policy must be present for a deadlock to be possible: mutual exclusion, no

preemption, and \_\_\_\_\_\_\_\_\_ .

6)  Three general approaches exist for dealing with deadlock: prevent, avoid, and \_\_\_\_\_\_\_\_\_\_ .

7)  \_\_\_\_\_\_\_\_\_\_ allows the three necessary conditions but makes judicious choices to assure that the

deadlock point is never reached.

8)  The strategy of resource allocation denial is referred to as the \_\_\_\_\_\_\_\_\_\_\_ .

9)  The \_\_\_\_\_\_\_\_\_ of the system reflects the current allocation of resources to processes.

10)  \_\_\_\_\_\_\_\_\_ strategies are very conservative and solve the problem of deadlock by limiting access

to resources and by imposing restrictions on processes.

11)  Inspired by the concept of co-routines, a \_\_\_\_\_\_\_\_\_ is a circular buffer allowing two processes

to communicate on the producer-consumer model.

12)  Two types of atomic operations are defined in Linux: integer operations and \_\_\_\_\_\_\_\_\_\_ .

13)  Linux provides three types of semaphore facilities in the kernel: binary semaphores, counting

semaphores, and \_\_\_\_\_\_\_\_\_\_ .

14)  An executable entity within a process is a \_\_\_\_\_\_\_\_\_\_ object.

15)  A program invocation, including the address space and resources required to run the program

is a \_\_\_\_\_\_\_\_\_\_ object.