Chapter 8

Multiple Choice Questions

1. In the deadlock example shown in Figure 8.1,
A) a deadlock is guaranteed to occur.
B) a deadlock may occur sometime, but not necessarily every time.
C) a deadlock may occur sometime only if there are more than one instances of thread_one and thread_two are running.
D) a deadlock is guaranteed to occur as long as there are more than one instances of thread_one and thread_two are running.

Ans: B
Feedback: 8.2
Difficulty: Easy

2. In the livelock example shown in Figure 8.2,
A) a livelock is guaranteed to occur.
B) a livelock may occur sometime.
C) a livelock may occur sometime only if there are more than one instances of thread_one and thread_two are running.
D) a livelock is guaranteed to occur as long as there are more than one instances of thread_one and thread_two are running.

Ans: B
Feedback: 8.2.1
Difficulty: Medium

3. One necessary condition for deadlock is ____, which states that at least one resource must be held in a nonsharable mode.
A) hold and wait
B) mutual exclusion
C) circular wait
D) no preemption

Ans: B
Feedback: 8.3.1
Difficulty: Easy

4. One necessary condition for deadlock is _____, which states that a resource can be released only voluntarily by the process holding the resource.
A) hold and wait
B) mutual exclusion
C) circular wait
D) no preemption

Ans: D
Feedback: 8.3.1
Difficulty: Easy

5. One necessary condition for deadlock is _____, which states that there is a chain of waiting processes whereby P_0 is waiting for a resource held by P_1, P_1 is waiting for a resource held by P_2, and P_n is waiting for a resource held by P_0.
A) hold and wait
B) mutual exclusion
C) circular wait
D) no preemption

Ans: C
Feedback: 8.3.1
Difficulty: Easy

6. In a system resource-allocation graph, ____.
A) a directed edge from a process to a resource is called an assignment edge
B) a directed edge from a resource to a process is called a request edge
C) a directed edge from a process to a resource is called a request edge
D) None of the above

Ans: C
Feedback: 8.3.2
Difficulty: Easy

7. In a system resource-allocation graph, ____.
A) a directed edge from a resource to a process is called an assignment edge
B) a directed edge from a resource to a process is called a request edge
C) a directed edge from a process to a resource is called an assignment edge
D) None of the above

Ans: A
Feedback: 8.3.2
Difficulty: Easy

8. In a resource allocation graph, when a process releases a resource,
A) a request edge is inserted.
B) an assignment edge is inserted.
C) a request edge is removed.
D) an assignment edge is removed.

Ans: D
Feedback: 8.3.2
Difficulty: Medium

9. A system will never enter a deadlocked state if
A) the system chooses to ignore the problem altogether.
B) the system uses the detection and recovery technique.
C) the system uses the deadlock avoidance technique.
D) None of the above.

Ans: C
Feedback: 8.4
Difficulty: Medium

10. Deadlocks can be prevented only if
A) all four necessary conditions cannot hold.
B) at least one of the four necessary conditions cannot hold.
C) mutual exclusion condition cannot hold.
D) circular wait condition cannot hold.

Ans: B
Feedback: 8.5
Difficulty: Medium

11. A _____ could be preempted from a process.
A) mutex lock
B) CPU registers
C) semaphore
D) file lock

Ans: B
Feedback: 8.5.3
Difficulty: Medium

12. Deadlock prevention using preempting allocated resources cannot be used for
A) mutexes.
B) CPU registers.
C) database transactions.
D) memory.

Ans: A
Feedback: 8.5.3
Difficulty: Medium

13. The Linux Lockdep tool is
A) used to verify locking order in the kernel.
B) a modeler to develop resource allocation graphs
C) a driver that can be used to prevent mutual exclusion for nonsharable resources
D) an implementation of the banker's algorithm available for most operating systems

Ans: A
Feedback: 8.6
Difficulty: Medium

14. Which of the following statement is true?
A) An unsafe state is a deadlocked state.
B) A deadlocked state is a safe state.
C) An unsafe state will lead to a deadlocked state.
D) An unsafe state may lead to a deadlocked state.

Ans: D
Feedback: 8.6.1
Difficulty: Medium

15. Suppose that there are ten resources available to three processes. At time 0, the following data is collected. The table indicates the process, the maximum number of resources needed by the process, and the number of resources currently owned by each process.

Process	Maximum Needs	Currently Owned
P_0	10	4
P_1	3	1
P_2	5	4

At time 1, P1 requests a resource. Which of the following is correct?

A) The request will be granted, since the current state is safe.
B) The request will not be granted, since the current state is not safe.
C) The request will be granted, since the state after granting the request will be safe.
D) The request will not be granted, since the state after granting the request will be unsafe.

16. A claim edge in a resource-allocation graph indicates that
A) a process has been allocated a resource.
B) a process may request a resource.
C) a process may release a resource that it has already been assigned.
D) a process has requested a resource.

17. Which of the following data structures in the banker's algorithm is a vector of length m, where m is the number of resource types?
A) Need
B) Allocation
C) Max
D) Available

18. Suppose that there are two resource types (R1 and R2) with five resources each available to four processes. At time 0, the following data is collected. The table indicates the process, the number of resources of each type currently allocated to the processes, and the current request of each resource type by each process.

	Process Allocation		Request	
	R1	R2	R1	R2
P_0	2	0	3	2
P_1	1	1	1	0
P_2	0	1	1	1
P_3	1	1	3	2

Which of the following sentences is correct?

A) All four processes are currently deadlocked.
B) The system is not deadlocked.
C) Processes P_0 and P_3 are deadlocked.
D) Processes P_0, P_1 and P_3 are deadlocked.

Ans: C
Feedback: 8.6.3
Difficulty: Difficult

19. In a system that uses deadlock detection algorithm,
A) a deadlock is detected as soon as it occurs.
B) a deadlock is detected just before it occurs.
C) a deadlock is detected sometime after it has occurred but not necessarily immediately.
D) a deadlock is detected sometime before it occurs, but not necessarily just before.

Ans: C
Feedback: 8.7.3
Difficulty: Medium

20. Which of the following is not a factor in choosing which process to terminate during recovery from a deadlock?
A) Who the parent of the process is.
B) How many more resources the process needs in order to complete.
C) How many processes will need to be terminated.
D) What the priority of the process is

Ans: A
Feedback: 8.8.1
Difficulty: Easy

Essay Questions

1. Explain what has to happen for a set of processes to achieve a deadlocked state.

Ans: For a set of processes to exist in a deadlocked state, every process in the set must be waiting for an event that can be caused only be another process in the set. Thus, the processes cannot ever exit this state without manual intervention.
Feedback: 8.1
Difficulty: Medium

2. Describe the four conditions that must hold simultaneously in a system if a deadlock is to occur.

Ans: For a set of processes to be deadlocked: at least one resource must remain in a nonsharable mode, a process must hold at least one resource and be waiting to acquire additional resources held by other processes, resources in the system cannot be preempted, and a circular wait has to exist between processes.
Feedback: 8.3.1
Difficulty: Medium

3. What are the three general ways that a deadlock can be handled?

Ans: A deadlock can be prevented by using protocols to ensure that a deadlock will never occur. A system may allow a deadlock to occur, detect it, and recover from it. Lastly, an operating system may just ignore the problem and pretend that deadlocks can never occur.
Feedback: 8.4
Difficulty: Medium

4. What is the difference between deadlock prevention and deadlock avoidance?

Ans: Deadlock prevention is a set of methods for ensuring that at least one of the necessary conditions for deadlock cannot hold. Deadlock avoidance requires that the operating system be given, in advance, additional information concerning which resources a process will request and use during its lifetime.
Feedback: 8.4
Difficulty: Medium

5. Describe two protocols to ensure that the hold-and-wait condition never occurs in a system.

Ans: One protocol requires each process to request and be allocated all its resources before it begins execution. We can implement this provision by requiring that system calls requesting resources for a process precede all other system calls. An alternative protocol allows a process to request resources only when it has none. A process may request some resources and use them. Before it can request any additional resources, however, it must release all the resources that it is currently allocated.
Feedback: 8.5.2
Difficulty: Medium

6. What is one way to ensure that a circular-wait condition does not occur?

Ans: One way to ensure that this condition never holds is to impose a total ordering of all resource types, and to require that each process requests resources in an increasing order of enumeration. This can be accomplished by assigning each resource type a unique integer number to determine whether one precedes another in the ordering.
Feedback: 8.5.4
Difficulty: Medium

7. Describe how a safe state ensures deadlock will be avoided.

Ans: A safe state ensures that there is a sequence of processes to finish their program execution. Deadlock is not possible while the system is in a safe state. However, if a system goes from a safe state to an unsafe state, deadlock is possible. One technique for avoiding deadlock is to ensure that the system always stays in a safe state. This can be done by only assigning a resource as long as it maintains the system in a safe state.
Feedback: 8.6.1
Difficulty: Medium

8. What does a claim edge signify in a resource-allocation graph?

Ans: A claim edge indicates that a process may request a resource at some time in the future. This edge resembles a request edge in direction, but is represented in the graph by a dashed line.
Feedback: 8.6.2
Difficulty: Easy

9. Describe a *wait-for* graph and how it detects deadlock.

Ans: If all resources have only a single instance, then we can define a deadlock-detection algorithm that uses a variant of the resource-allocation graph, called a wait-for graph. We obtain this graph from the resource-allocation graph by removing the resource nodes and collapsing the appropriate edges. To detect deadlocks, the system needs to maintain the wait-for graph and periodically invoke an algorithm that searches for a cycle in the graph.
Feedback: 8.7.1
Difficulty: Difficult

10. What factors influence the decision of when to invoke a detection algorithm?

Ans: The first factor is how often a deadlock is likely to occur; if deadlocks occur frequently, the detection algorithm should be invoked frequently. The second factor is how many processes will be affected by deadlock when it happens.
Feedback: 8.7.3
Difficulty: Medium

11. Describe two methods for eliminating deadlock by aborting a process.

Ans: The first method is to abort all deadlocked processes. Aborting all deadlocked processes will clearly break the deadlock cycle; however, the deadlocked processes may have to be computed for a long time, and results of these partial computations must be discarded and will probably have to be recomputed later. The second method is to abort one process at a time until the deadlock cycle is eliminated. Aborting one process at a time incurs considerable overhead, since, after each process is aborted, a deadlock-detection algorithm must be invoked to determine whether any processes are still deadlocked.
Feedback: 8.8.1
Difficulty: Easy

12. Name three issues that need to be addressed if a preemption is required to deal with deadlocks.

Ans: First, the order of resources and processes that need to be preempted must be determined to minimize cost. Second, if a resource is preempted from a process, the process must be rolled back to

some safe state and restarted from that state. The simplest solution is a total rollback. Finally, we must ensure that starvation does not occur from always preempting resources from the same process.
Feedback: 8.8.2
Difficulty: Medium

True/False Questions

1. The circular-wait condition for a deadlock implies the hold-and-wait condition.

Ans: True
Feedback: 8.3.1
Difficulty: Easy

2. Deadlock prevention and deadlock avoidance are essentially the same approaches for handling deadlock.

Ans: False
Feedback: 8.4
Difficulty: Medium

3. Protocols to prevent hold-and-wait conditions typically also prevent starvation.

Ans: False
Feedback: 8.5.2
Difficulty: Medium

4. Ordering resources and requiring the resources to be acquired in order prevents the circular wait from occurring and therefore prevents deadlock from occurring.

Ans: False
Feedback: 8.5.4
Difficulty: Medium

5. A system in an unsafe state will ultimately deadlock.

Ans: False
Feedback: 8.6.1
Difficulty: Medium

6. If the sum of all needs for a resource type exceeds the available number of instances of that resource type, the system state is unsafe.

Ans: False
Feedback: 8.6.1
Difficulty: Hard

7. The banker's algorithm is useful in a system with multiple instances of each resource type.

Ans: True
Feedback: 8.6.3
Difficulty: Easy

8. Recovery from a deadlock works best when a single process can be identified whose termination will break the circular wait.

Ans: False
Feedback: 8.8.1
Difficulty: Medium