Chapter 4

Multiple Choice

1. Which of the following items are shared across multiple threads belonging to the same process?
A) code, data, files
B) registers
C) stack
D) all of the above

Ans: A
Feedback: 4.1
Difficulty: Easy

2. In a multithreaded server architecture, which of the following is used to service a new user request?
A) a new created thread
B) a new created process
C) the same process for prior users
D) none of the above

Ans: A
Feedback: 4.1.1
Difficulty: Medium

3. Which of the following benefits go to multithreaded programming?
A) responsiveness
B) resource sharing
C) economy
D) scalability
E) all of the above

Ans: E
Feedback: 4.1.2
Difficulty: Easy

4. Which of the following refers to the capability to allow multiple tasks make progress on a single processor system?
A) concurrency
B) parallelism
C) data parallelism
D) task parallelism

Ans: A
Feedback: 4.2
Difficulty: Medium

5. ___________ is a formula that identifies potential performance gains from adding additional computing cores to an application that has a parallel and serial component.
A) Task parallelism
B) Data parallelism
C) Data splitting
D) Amdahl's Law

Ans: D
Feedback: 4.2.1
Difficulty: Medium

6. _____ is not considered a challenge when designing applications for multicore systems.
A) Deciding which activities can be run in parallel
B) Ensuring there is a sufficient number of cores
C) Determining if data can be separated so that it is accessed on separate cores
D) Identifying data dependencies between tasks.

Ans: B
Feedback: 4.2.1
Difficulty: Medium

7. Which of the following models are possible for the relationship between the user threads and kernel threads?
A) many-to-one model
B) one-to-one model
C) many-to-many model
D) two-level model
E) all of the above

Ans: E
Feedback: 4.3
Difficulty: Easy

8. ____ is a thread library for Solaris that maps many user-level threads to one kernel thread.
A) Pthreads
B) Green threads
C) Sthreads
D) Java threads

Ans: B
Feedback: 4.3.1
Difficulty: Medium

9. The _____ model maps each user-level thread to one kernel thread.
A) many-to-many
B) two-level
C) one-to-one
D) many-to-one

Ans: C
Feedback: 4.3.2
Difficulty: Easy

10. The _____ model multiplexes many user-level threads to a smaller or equal number of kernel threads.
A) many-to-many
B) two-level
C) one-to-one
D) many-to-one

Ans: A
Feedback: 4.3.3
Difficulty: Easy

11. Which of the following is a function that can be provided by Pthreads API for constructing a multithreaded program?
A) pthread attr init
B) pthread_create
C) pthread_join
D) all of the above

Ans: D
Feedback: 4.4.1
Difficulty: Easy

12. In Pthreads, a parent uses the pthread_join() function to wait for its child thread to complete. What is the equivalent function in Win32?
A) win32_join()
B) wait()
C) WaitForSingleObject()
D) join()

Ans: C
Section 4.4.2
Difficulty: Medium

13. Which of the following statements regarding threads is false?
A) Sharing is automatically provided in Java threads.
B) Both Pthreads and Win32 threads share global data.
C) The start() method actually creates a thread in the Java virtual machine.
D) The Java method join() provides similar functionality as the WaitForSingleObject in Win32.

Ans: A
Feedback: 4.4.3
Difficulty: Medium

14. The most common technique for writing multithreaded Java programs is _____.
A) extending the Thread class and overriding the run() method
B) implementing the Runnable interface and defining its run() method
C) designing your own Thread class
D) using the CreateThread() function

Ans: B
Feedback: 4.4.3
Difficulty: Easy

15. Which of the following is a method for implicit threading?
A) thread pools
B) OpenMP
C) grand central dispatch
D) all of the above

Ans: D
Feedback: 4.5
Difficulty: Easy

16. Which of the following implicit threading methodology involves dispatch queues?
A) thread pools
B) OpenMP
C) grand central dispatch
D) all of the above

Ans: C
Feedback: 4.5.4
Difficulty: Easy

17. Which of the following is an asynchronous signal?
A) illegal memory access
B) division by zero
C) terminating a process with specific keystrokes
D) none of the above
Ans: C
Feedback: 4.6.2
Difficulty: Medium

18. Which of the following is a synchronous signal?
A) illegal memory access
B) terminating a process with specific keystrokes
C) having a timer expire
D) none of the above

Ans: A
Feedback: 4.6.2
Difficulty: Medium

19. Which of the following options exist to deliver signals in multithreaded program?
A) deliver the signal to the thread to which the signal applies
B) deliver the signal to every thread in the process
C) deliver the signal to certain threads in the process
D) assign a specific thread to receive all signals for the process
E) all of the above

Ans: E
Feedback: 4.6.2
Difficulty: Easy

20. Which of the following options to deliver signals in multithreaded program should be applied to an asynchronous signal that terminates a process (<control><C>, for example)?
A) deliver the signal to the thread to which the signal applies
B) deliver the signal to every thread in the process
C) deliver the signal to certain threads in the process
D) assign a specific thread to receive all signals for the process
E) all of the above

Ans: B
Feedback: 4.6.2
Difficulty: Medium

21. Which of the following options to deliver signals in multithreaded program should be applied to a synchronous signal?
A) deliver the signal to the thread to which the signal applies
B) deliver the signal to every thread in the process
C) deliver the signal to certain threads in the process
D) assign a specific thread to receive all signals for the process
E) all of the above

Ans: A
Feedback: 4.6.2
Difficulty: Medium

22. Signals can be emulated in windows through ____.
A) asynchronous procedure calls
B) local procedure calls
C) remote procedure calls
D) none of the above

Ans: A
Feedback: 4.6.2
Difficulty: Medium

23. Which of the following cancellation modes are supported by Pthreads?
A) off mode
B) deferred mode
C) asynchronous mode
D) all of the above

Ans: D
Feedback: 4.6.3
Difficulty: Easy

24. Which of the following cancellation modes is the default cancellation type?
A) off mode
B) deferred mode
C) asynchronous mode
D) all of the above

Ans: B
Feedback: 4.6.3
Difficulty: Easy

25. Which of the following Pthread function is used to respond to a cancellation request using deferred cancellation?
A) pthread attr init
B) pthread_create
C) pthread_join
D) pthread_testcancel

Ans: D
Feedback: 4.6.3
Difficulty: Medium

26. To associate each thread created using an implicit technique such as a thread pool, with its unique transaction identifier, we could use ____?
A) global variable
B) local variable
C) static data
D) thread-local storage

Ans: D
Feedback: 4.6.4
Difficulty: Easy

27. LWP is ____.
A) short for lightweight processor
B) placed between system and kernel threads
C) placed between user and kernel threads
D) common in systems implementing one-to-one multithreading models

Ans: C
Feedback: 4.6.5
Difficulty: Easy

28. In multithreaded programs, the kernel informs an application about certain events using a procedure known as a(n) ____.
A) signal
B) upcall
C) event handler
D) pool

Ans: B
Feedback: 4.6.5
Difficulty: Medium

29. Which are included in the context of a thread?
A) register set
B) stacks
C) private storage area
D) all of the above

Ans: D
Feedback: 4.7.1
Difficulty: Easy

30. Which of the following information is shared when the flag CLONE_VM is set up?
A) file-system information
B) memory space
C) signal handlers
D) set of open files

Ans: B
Feedback: 4.7.2
Difficulty: Easy

Essay Questions

1. Why should a web server not run as a single-threaded process?

Ans: For a web server that runs as a single-threaded process, only one client can be serviced at a time. This could result in potentially enormous wait times for a busy server.
Feedback: 4.1.1
Difficulty: Medium

2. List the four major categories of the benefits of multithreaded programming. Briefly explain each.

Ans: The benefits of multithreaded programming fall into the categories: responsiveness, resource sharing, economy, and utilization of multiprocessor architectures. Responsiveness means that a multithreaded program can allow a program to run even if part of it is blocked. Resource sharing occurs when an application has several different threads of activity within the same address space. Threads share the resources of the process to which they belong. As a result, it is more economical to create new threads than new processes. Finally, a single-threaded process can only execute on one processor regardless of the number of processors actually present. Multiple threads can run on multiple processors, thereby increasing efficiency.
Feedback: 4.1.2
Difficulty: Hard

3. Distinguish between parallelism and concurrency.

Ans: A parallel system can perform more than one task simultaneously. A concurrent system supports more than one task by allowing multiple tasks to make progress.
Feedback: 4.2
Difficulty: Medium

4. Multicore systems present certain challenges for multithreaded programming. Briefly describe these challenges.

Ans: Multicore systems have placed more pressure on system programmers as well as application developers to make efficient use of the multiple computing cores. These challenges include determining how to divide applications into separate tasks that can run in parallel on the different cores. These tasks must be balanced such that each task is doing an equal amount of work. Just as tasks must be separated, data must also be divided so that it can be accessed by the tasks running on separate cores. So that data can safely be accessed, data dependencies must be identified and where such dependencies exist, data accesses must be synchronized to ensure the safety of the data. Once all such challenges have been met, there remains considerable challenges testing and debugging such applications.
Feedback: 4.2.1
Difficulty: Difficult

5. Distinguish between data and task parallelism.

And: Data parallelism involves distributing subsets of the same data across multiple computing cores and performing the same operation on each core. Task parallelism involves distributing tasks across the different computing cores where each task is performing a unique operation.

Feedback: 4.2.2
Difficulty: Hard

6. What are the two different ways in which a thread library could be implemented?

Ans: The first technique of implementing the library involves ensuring that all code and data structures for the library reside in user space with no kernel support. The other approach is to implement a kernel-level library supported directly by the operating system so that the code and data structures exist in kernel space.
Feedback: 4.4
Difficulty: Medium

7. What are the general components of a thread in Windows?

Ans: The thread consists of a unique ID, a register set that represents the status of the processor, a user stack for user mode, a kernel stack for kernel mode, and a private storage area used by run-time libraries and dynamic link libraries.
Feedback: 4.4.2
Difficulty: Medium

8. Describe two techniques for creating Thread objects in Java.

Ans: One approach is to create a new class that is derived from the Thread class and to override its run() method. An alternative — and more commonly used — technique is to define a class that implements the Runnable interface. When a class implements Runnable, it must define a run() method. The code implementing the run() method is what runs as a separate thread.
Feedback: 4.4.3
Difficulty: Medium

9. In Java, what two things does calling the start() method for a new Thread object accomplish?

Ans: Calling the start() method for a new Thread object first allocates memory and initializes a new thread in the JVM. Next, it calls the run() method, making the thread eligible to be run by the JVM. Note that the run() method is never called directly. Rather, the start() method is called, which then calls the run() method.
Feedback: 4.4.3
Difficulty: Medium

10. What is a thread pool and why is it used?

Ans: A thread pool is a collection of threads, created at process startup, that sit and wait for work to be allocated to them. This allows one to place a bound on the number of concurrent threads associated with a process and reduce the overhead of creating new threads and destroying them at termination.
Feedback: 4.5.1
Difficulty: Medium

11. Describe how OpenMP is a form of implicit threading.

Ans: OpenMP provides a set of compiler directives that allows parallel programming on systems that support shared memory. Programmers identify regions of code that can run in parallel by placing them in a block of code that begins with the directive #pragma omp parallel. When the compiler encounters this parallel directive, it creates as many threads as there are processing cores in the system.
Feedback: 4.5.2
Difficulty: Hard

12. Describe how Grand Central Dispatch is a form of implicit threading.

Ans: Grand Central Dispatch (GCD) is a technology for Mac OS X and iOS systems that is a combination of extensions to the C language, an API, and a runtime library that allows developers to construct "blocks" - regions of code that can run in parallel. GCD then manages the parallel execution of blocks in several dispatch queues.
Feedback: 4.5.3
Difficulty: Hard

13. Some UNIX systems have two versions of fork(). Describe the function of each version, as well as how to decide which version to use.

Ans: One version of fork() duplicates all threads and the other duplicates only the thread that invoked the fork() system call. Which of the two versions of fork() to use depends on the application. If exec() is called immediately after forking, then duplicating all threads is unnecessary, as the program specified in the parameters to exec() will replace the process. If, however, the separate process does not call exec() after forking, the separate process should duplicate all threads.
Feedback: 4.6.1
Difficulty: Hard

14. How can deferred cancellation ensure that thread termination occurs in an orderly manner as compared to asynchronous cancellation?

Ans: In asynchronous cancellation, the thread is immediately cancelled in response to a cancellation request. There is no insurance that it did not quit in the middle of a data update or other potentially dangerous situation. In deferred cancellation, the thread polls whether or not it should terminate. This way, the thread can be made to cancel at a convenient time.
Feedback: 4.6.3
Difficulty: Medium

15. Describe the difference between the fork() and clone() Linux system calls.

Ans: The fork() system call is used to duplicate a process. The clone() system call behaves similarly except that, instead of creating a copy of the process, it creates a separate process that shares the address space of the calling process.
Feedback: 4.7.2
Difficulty: Medium

True/False Questions

1. Virtually all contemporary operating systems support kernel threads.

Ans: True
Feedback: 4.1.1
Difficulty: Easy

2. Amdahl's Law describes performance gains for applications with both a serial and parallel component.

Ans: True
Feedback: 4.2
Difficulty: Medium

3. Concurrency is inherently equivalent to parallelism.

Ans: False
Feedback: 4.2
Difficulty: Medium

4. Parallelism can be achieved on single-processor systems.

Ans: False
Feedback: 4.2
Difficulty: Medium

5. Task parallelism distributes threads and data across multiple computing cores.

Ans: False
Feedback: 4.2.2
Difficulty: Difficult

6. One-to-one model provides more concurrency than the many-to-one model by allowing another thread to run when a thread makes a blocking system call.

Ans: True
Feedback: 4.3.2
Difficulty: Easy

7. In Java, data shared between threads is simply declared globally.

Ans: False
Feedback: 4.4.3
Difficulty: Medium

8. The trend in developing parallel applications is to use implicit threading.

Ans: True
Feedback: 4.5
Difficulty: Medium

9. The single benefit of a thread pool is to control the number of threads.

Ans: False
Feedback: 4.5.1
Difficulty: Easy

10. OpenMP only works for C, C++, and Fortran programs.

Ans: True
Feedback 4.5.2:
Difficulty: Medium

11. Grand Central Dispatch requires multiple threads.

Ans: False
Feedback: 4.5.3
Difficulty: Medium

12. The default signal handler can be overridden by a user-defined signal handler.

Ans: True
Feedback: 4.6.2
Difficulty: Easy

13. Deferred cancellation is preferred over asynchronous cancellation.

Ans: True
Feedback: 4.6.3
Difficulty: Easy

14. Thread-local storage is inherently equivalent to local variables.

Ans: False
Feedback: 4.6.4
Difficulty: Medium

15. When fork() is invoked, it is passed a set of flags that determine how much sharing is to take place between the parent and child tasks.

Ans: False
Feedback: 4.7.2
Difficulty: Medium