Chapter 2

Multiple Choice Questions

1. Which of the following statements is incorrect?
A) An operating system provides an environment for the execution of programs.
B) An operating system manages system resources.
C) Operating systems provide both command line as well as graphical user interfaces.
D) Operating systems must provide both protection and security.

Ans: C
Feedback: 2.1
Difficulty: Easy

2. Which of the following is an inter-process communication that requires two processes share memory?
A) message passing
B) shared memory
C) socket communication
D) all of the above

Ans: B
Feedback: 2.1
Difficulty: Easy

3. Which of the following is not an operating system service?
A) program execution
B) I/O operation
C) protection and security
D) graphical user interface

Ans: D
Feedback: 2.2
Difficulty: Easy

4. Which of the following is a user interface where human fingers slide across the screen or press buttons to select choices?
A) touch screen
B) graphical user interface
C) command line
D) voice control

Ans: A
Feedback: 2.2.1
Difficulty: Easy

5. Which of the following is not a type of command interpreter?
A) Bourne shell
B) C shell
C) Korn shell
D) KDE or GNOME

Ans: D
Feedback: 2.2.1
Difficulty: Easy

6. _____ provide(s) an interface to the services provided by an operating system.
A) Shared memory
B) System calls
C) Simulators
D) Communication

Ans: B
Feedback: 2.3
Difficulty: Medium

7. Which of the following is the only gateway between user space and kernel space?
A) user interface
B) system call
C) operating system
D) hardware drivers

Ans: B
Feedback: 2.3.2
Difficulty: Medium

8. What is the relationship between library call *open()* and *open()* system call?
A) open() system call is part of open()
B) open() system call in an alternate version of open()
C) open() is a predecessor version of open() system call
D) open() invokes open() system call to get service from operating system

Ans: D
Feedback: 2.3.2
Difficulty: Medium

9. What category of system call includes the ones for process creation of and termination?
A) process control
B) file management
C) device management
D) protection

Ans: A
Feedback: 2.3.3.1
Difficulty: Easy

10. Which of the following defines the view of the operating system seen by most users?
A) application and system programs
B) system calls
C) device drivers
D) library calls

Ans: A
Feedback: 2.4
Difficulty: Easy

11. The Windows CreateProcess() system call creates a new process. What is the equivalent system call in UNIX?
A) NTCreateProcess()
B) process()
C) fork()
D) getpid()

Ans: C
Feedback: 2.3.3.1
Difficulty: Easy

12. The close() system call in UNIX is used to close a file. What is the equivalent system call in Windows?
A) CloseHandle()
B) close()
C) CloseFile()
D) Exit()

Ans: A
Feedback: 2.3.3.1
Difficulty: Easy

13. The Windows CreateFile() system call is used to create a file. What is the equivalent system call in UNIX?
A) ioctl()
B) open()
C) fork()
D) createfile()

Ans: B
Feedback: 2.3.3.1
Difficulty: Easy

14. The _____ provides a portion of the system call interface for UNIX and Linux.
A) POSIX
B) Java
C) Standard C library
D) Standard API

Ans: C
Feedback: 2.3.3.1
Difficulty: Medium

15.   A message-passing model is ____.
A)   easier to implement than a shared memory model for intercomputer communication
B)   faster than the shared memory model
C)   a network protocol, and does not apply to operating systems
D)   only useful for small simple operating systems

Ans: A
Feedback: 2.3.3.5
Difficulty: Medium

16. Microkernels use _____ for communication.
A) message passing
B) shared memory
C) system calls
D) virtualization

Ans: A
Feedback: 2.8.3
Difficulty: Easy

17. Android runs Java programs _____
A) in the Android RunTime (ART) virtual machine.
B) natively.
C) in the Java virtual machine.
D) Android does not run Java programs.

Ans: A
Feedback: 2.8.5.2
Difficulty: Medium

18. _____ is a mobile operating system designed for the iPhone and iPad.
A) Mac OS X
B) Android
C) UNIX
D) iOS

Ans: D
Feedback: 2.8.5.1
Difficulty: Medium

19. Which of the following operating system structure is the one for MS-DOS?
A) monolithic structure
B) layered structure
C) microkernel
D) modular approach

Ans: A
Feedback: 2.8.1
Difficulty: Easy

20. Which of the following inter-process communication is used in microkernel?
A) message passing
B) shared memory
C) socket communication
D) all of the above

Ans: A
Feedback: 2.8.3
Difficulty: Medium

21. Which of the following operating system structure is the one for Mach OS?
A) monolithic structure
B) layered structure
C) microkernel
D) modular approach

Ans: C
Feedback: 2.8.3
Difficulty: Easy

22. Which of the following operating system structure involves using loadable kernel module in design?
A) monolithic structure
B) layered structure
C) microkernel
D) modular approach

Ans: D
Feedback: 2.8.4
Difficulty: Easy

23. Which of the following is a much more restricted, or even closed, system to developers than others?
A) Linux
B) Mac OS X
C) iOS
D) Android

Ans: C
Feedback: 2.8.5.1
Difficulty: Easy

24. Which of the following is the correct program type for GRUB in Linux and Unix systems?
A) bootstrap program
B) compiler program
C) binder program
D) system utility

Ans: A
Feedback: 2.9.2
Difficulty: Easy

25. When a process fails, the operating system takes a _____ which can then be probed by a debugger for failure analysis?
A) core dump
B) crash dump
C) capture of network statistic information
D) capture of CPU usage information

Ans: A
Feedback: 2.10.1
Difficulty: Easy

Essay Questions

1.   There are two different ways that commands can be processed by a command interpreter. One way is to allow the command interpreter to contain the code needed to execute the command. The other way is to implement the commands through system programs. Compare and contrast the two approaches.

Ans: In the first approach, upon the user issuing a command, the interpreter jumps to the appropriate section of code, executes the command, and returns control back to the user. In the second approach, the interpreter loads the appropriate program into memory along with the appropriate arguments. The advantage of the first method is speed and overall simplicity. The disadvantage to this technique is that new commands require rewriting the interpreter program which, after a number of modifications, may get complicated, messy, or too large. The advantage to the second method is that new commands can be added without altering the command interpreter. The disadvantage is reduced speed and the clumsiness of passing parameters from the interpreter to the system program.
Feedback: 2.2
Difficulty: Hard

2.   Describe the relationship between an API, the system-call interface, and the operating system.

Ans: The system-call interface of a programming language serves as a link to system calls made available by the operating system. This interface intercepts function calls in the API and invokes the necessary system call within the operating system. Thus, most of the details of the operating-system interface are hidden from the programmer by the API and are managed by the run-time support library.
Feedback: 2.3
Difficulty: Hard

3.   Describe three general methods used to pass parameters to the operating system during system calls.

Ans: The simplest approach is to pass the parameters in registers. In some cases, there may be more parameters than registers. In these cases, the parameters are generally stored in a block, or table, of memory, and the address of the block is passed as a parameter in a register. Parameters can also be placed, or pushed, onto the stack by the program and popped off the stack by the operating system.
Feedback: 2.3
Difficulty: Medium

4.   Describe some requirements, or goals, when designing an operating system.

Ans: Requirements can be divided into user and system goals. Users desire a system that is convenient to use, easy to learn, and to use, reliable, safe, and fast. System goals are defined by those people who must design, create, maintain, and operate the system: The system should be easy to design, implement, and maintain; it should be flexible, reliable, error-free, and efficient.
Feedback: 2.7.1
Difficulty: Medium

5.   What are the advantages of using a higher-level language to implement an operating system?

Ans: The code can be written faster, is more compact, and is easier to understand and debug. In addition, improvements in compiler technology will improve the generated code for the entire operating system by simple recompilation. Finally, an operating system is far easier to port — to move to some other hardware — if it is written in a higher-level language.
Feedback: 2.7.3
Difficulty: Medium

6. What are the advantages and disadvantages of using a microkernel approach?

Ans: One benefit of the microkernel approach is ease of extending the operating system. All new services are added to user space and consequently do not require modification of the kernel. The microkernel also provides more security and reliability, since most services are running as user — rather than kernel — processes. Unfortunately, microkernels can suffer from performance decreases due to increased system function overhead.
Feedback: 2.8.3
Difficulty: Medium

7. Explain why a modular kernel may be the best of the current operating system design techniques.

Ans: The modular approach combines the benefits of both the layered and microkernel design techniques. In a modular design, the kernel needs only to have the capability to perform the required functions and know how to communicate between modules. However, if more functionality is required in the kernel, then the user can dynamically load modules into the kernel. The kernel can have sections with well-defined, protected interfaces, a desirable property found in layered systems. More flexibility can be achieved by allowing the modules to communicate with one another.
Feedback: 2.8.4
Difficulty: Hard

8. Describe how Mac OS X is considered a hybrid system.

Ans: Primarily because the kernel environment is a blend of the Mach microkernel and BSD UNIX (which is closer to a monolithic kernel.)
Feedback: 2.8.5.1
Difficulty: Medium

9. Describe how Android uses a unique virtual machine for running Java programs.

Ans: The Android RunTime (ART) virtual machine is designed specifically for Android and has been optimized for mobile devices with limited memory and CPU processing capabilities.
Feedback: 2.8.5.2
Difficulty: Medium

True/False Questions

1. Application programmers typically use an API rather than directory invoking system calls.

Ans: True
Feedback: 2.3
Difficulty: Easy

2. In general, Windows system calls have longer, more descriptive names and UNIX system calls use shorter, less descriptive names.

Ans: True
Feedback: 2.4
Difficulty: Easy

3.   Many operating system merge I/O devices and files into a combined file because of the similarity of system calls for each.

Ans: True
Feedback: 2.4.3
Difficulty: Medium

4. Applications compiled on one operating system can be directly executable on other operating systems due to common structure.

Ans: False
Feedback: 2.6
Difficulty: Easy

5. One important principle is the separation of policy from mechanism.

Ans: True
Feedback: 2.7.2
Difficulty: Easy

6. Mac OS X is a hybrid system consisting of both the Mach microkernel and BSD UNIX.

Ans: True
Feedback: 2.8.5.1
Difficulty: Medium

7. iOS is open source, Android is closed source.

Ans: False
Feedback: 2.8.5.1
Difficulty: Medium

8. iOS and Android combine different structures, resulting in hybrid systems that address performance, security, and usability issues.

Ans: True
Feedback: 2.8.5
Difficulty: Easy