

CS 6233 21S

Sp21_midterm_Part
1_MCQ_problem

Section 1

1.

A ____ can be used to prevent a user program from never returning control to the operating system.

- ☐ program counter
- ☐ portal
- ☐ firewall
- ☐ timer interrupt

2.

In a personal computer, the bootstrap program that executes immediately after powerup can be stored in random-access memory (RAM).

- ☐ FALSE
- ☐ TRUE

3.

The operating system kernel contains ____

- ☐ a scheduler
- ☐ system and application programs
- ☐ user programs

4.

A program's entry point is _____

- ☐ the first instruction to execute in a program
- ☐ located at the beginning of the program
- ☐ the instruction that enters the shared library information

5.

The trap machine instruction is used in _____

- ☐ cache coherency
- ☐ trapping misbehaving user programs
- ☐ system calls

6.

Interrupt vectors _____

- ☐ are usually located inside a user program and are used to handle various signals
- ☐ contain the interrupt service routines for various hardware interrupts
- ☐ contain a set of branch instructions that jump to various interrupt service routines
- ☐ contain the interrupt service routines for various software interrupts

7.

The two main CPU modes of operation are _____

- ☐ physical mode and logical mode
- ☐ supervisor mode and system mode
- ☐ kernel mode and privileged mode
- ☐ user mode and kernel mode

Section 2

8.

The major difficulty in designing a layered operating system approach is ____.

- ☐ making sure each layer hides certain data structures, hardware, and operations from higher-level layers
- ☐ debugging a particular layer
- ☐ appropriately defining the various layers
- ☐ making sure each layer is easily converted to a module

9.

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

- ☐ FALSE
- ☐ TRUE

10.

_____ allow operating system components to be loaded dynamically.

- ☐ Virtual machines
- ☐ Kernel modules
- ☐ File systems
- ☐ Graphical user interfaces

11.

A microkernel is a kernel ____.

- ☐ that is compressed before loading in order to reduce its resident memory size
- ☐ containing a large number of components that are optimized to reduce resident memory size
- ☐ that is stripped of all nonessential components
- ☐ that is compiled to produce the smallest size possible when stored to disk

12.

Which of the following statements is incorrect?

- ☐ An operating system manages system resources.
- ☐ An operating system provides an environment for the execution of programs.
- ☐ Operating systems must provide both command line as well as graphical user interfaces

13.

A timesharing multitasking operating system ____

- ☐ is non-preemptive
- ☐ may be preemptive or non-preemptive
- ☐ uses a timer interrupt to invoke the scheduler
- ☐ preempts processes only when they request an I/O operation

14.

In a Unix-like operating system, a process _____

- ☐ has a text section, a data section, a stack and a heap
- ☐ has a text section, a data section, a stack, a heap and interrupt vectors
- ☐ is administered by a virtual machine manager

15.

If a user program issues a system call, it may pass an argument by _____

- ☐ passing its value through an interrupt service routine
- ☐ adding it to the interrupt vectors
- ☐ writing its value to the program counter
- ☐ writing its value to a CPU register other than the program counter

Section 3

16.

When communicating with sockets, a client process may initiate a request for a client socket and is assigned a port by its host computer. Which of the following would be a valid port assignment?

- ☐ 21
- ☐ 550
- ☐ 80
- ☐ 1625

17.

A process control block ____.

- ☐ stores the address of the next instruction to be processed by a different process
- ☐ is an example of a process queue
- ☐ determines which process is to be executed next
- ☐ includes information on the process's state

18.

A child process inherits UNIX ordinary pipes from its parent because:

- ☐ A common stack facilitates this sharing.
- ☐ The pipe is part of the code and children inherit code from their parents.
- ☐ A pipe is treated as a file descriptor and child processes inherit open file descriptors from their parents.

19.

In a single-processor system, there will never be more than one process in the Running state.

- ☐ FALSE
- ☐ TRUE

20.

A child process inherits UNIX ordinary pipes from its parent because:

- ☐ A common stack facilitates this sharing.
- ☐ The pipe is part of the code and children inherit code from their parents.
- ☐ A pipe is treated as a file descriptor and child processes inherit open file descriptors from their parents.

21.

In a single-processor system, there will never be more than one process in the Running state.

- ☐ FALSE
- ☐ TRUE

22.

In UNIX, a process that has terminated, but whose parent has not yet called wait(), is known as _____ process.

- ☐ terminated
- ☐ zombie
- ☐ init
- ☐ orphan

23.

The list of processes waiting for a particular I/O device is called a(n) ____.

- ☐ device queue
- ☐ interrupt queue
- ☐ ready queue

24.

In UNIX, the `exec()` system call creates a new process.

- ☐ FALSE
- ☐ TRUE

Section 4

25.

A critical race condition _____

- ☐ may results when several threads try to read the same data concurrently
- ☐ will result only if the outcome of execution does not depend on the order in which instructions are executed
- ☐ may results when several threads try to increment the same data concurrently

26.

In a multi-threaded process, each thread is characterized by having its own program counter, register set, thread ID, and heap.

- ☐ TRUE
- ☐ FALSE

27.

A mutex lock _____

- ☐ holds a boolean variable
- ☐ is exactly identical to a counting semaphore
- ☐ cannot be used to control access to a thread's critical section
- ☐ holds a variable indicating the current CPU mode

28.

The ____ multithreading model multiplexes many user-level threads to a smaller or equal number of kernel threads.

- ☐ one-to-one model
- ☐ many-to-some model
- ☐ many-to-many model
- ☐ many-to-one model

29.

A solution to the critical section problem DOES NOT have to satisfy which of the following requirements?

- ☐ progress
- ☐ mutual exclusion
- ☐ bounded waiting
- ☐ priority scheduling

30.

____ can be used to prevent busy waiting when implementing a semaphore.

- ☐ Allowing the wait() operation to succeed
- ☐ Wait queues
- ☐ Mutex lock
- ☐ Spinlocks

31.

A thread's cancellation points are associated with ____ cancellation.

- ☐ synchronous
- ☐ asynchronous
- ☐ non-deferred
- ☐ deferred

32.

When using semaphores, a process invokes the `wait()` operation before accessing its critical section, followed by the `signal()` operation upon completion of its critical section. Consider reversing the order of these two operations—first calling `signal()`, then calling `wait()`. What would be a possible outcome of this?

- ☐ Deadlock is possible.
- ☐ Starvation is possible.
- ☐ Mutual exclusion is still assured.
- ☐ Several processes could be active in their critical sections at the same time.

33.

According to Amdahl's Law, what is the speedup gain for an application that has 20% sequential code and we run it on a machine with 16 processing cores?

Your answer