CSCI 4500 / 8506 – Fall 2014
Sample Questions for Quiz 1
Covers Modules 1 and 2

1. The first microprocessors did not have separate modes for the kernel or supervisor and user. Of what significance is this to the operating system for such a microprocessor?

   a. As long as the user applications are developed using interpretive languages like Perl, PHP, and Python, there really shouldn't be any problems for the operating system, since the interpreter can detect user actions that would cause problems before they occur.
   b. Plenty! In these systems there is no way to prevent user programs and applications from intentionally or unintentionally corrupting code and data structures belonging to the operating system, or from attempting to use a shared I/O device.
   c. Memory protection is the major issue, since it's not possible to prevent user applications from looking at or changing memory that belongs to the kernel. As long as the kernel periodically checks the validity of its data, though, the lack of separate user and kernel modes really isn't a problem.

2. It has been said that the operating system is that portion of a system's software that runs in kernel mode or supervisor mode. Why is this definition inadequate?

   a. Some systems (like MINIX 3) run traditional operating system components in user mode. Still others may run multiple operating systems in user mode, with special virtualization software used to multiplex the real hardware resources between the operating systems.
   b. It fails to include important operating system components like the shell (command interpreter), editors, and compilers.
   c. Operating systems written in high-level languages, like Java, do not ever run in kernel mode.

3. At the machine language level, how are devices controlled?

   a. Special registers called device registers can be loaded to direct a device to perform an operation.
   b. Each device has its own set of machine language instructions that can be executed to cause the device to perform operations.
   c. Devices are not controlled by anything at the machine language level. All device control is done at the microarchitecture level.

4. What are the major functions of an operating system?

   a. Accept commands from the user, either through a textual interface like a shell or a graphical user interface, and arrange for their execution.
   b. Perform input/output on disks (and other similar devices) and terminals (keyboards, mice, and displays), and allocate and deallocate primary memory for user processes.
   c. Provide an extended machine which is easier to use than the bare hardware, and manage the use of complex system resources.

5. A computer system's resources can be shared in two ways. What are these?

   a. Sharing in time: user 1, user 2, user 1, ...; and sharing in space: user 1 and user 2 each get part of the resource.
   b. Sharing in place: each user can only use the resource where it is located; and sharing by relocation: users may move the resource to meet their needs.
   c. Sharing by lottery: a user needing the resource is randomly selected from all users needing the resource; and sharing by payment: a user willing to pay the most for the resource is allowed to use it first.

6. Who designed the first true digital computer?

   a. Charles Babbage
   b. Fred Brooks
   c. J. Presper Eckert

7. Which of the following devices was used in the first generation of real digital computers (that is, those that were actually built and used)?

   a. LSI technology
   b. vacuum tubes
   c. bioelectronic circuits

8. Who was the father of the person after whom the Ada (reg tm) programming language was named?

   a. General Montgomery
   b. Andrew Tanenbaum
   c. Lord Byron

9. What device(s) was(were) used for the input of jobs to, and output of results from, the main computer in a batch computing system?

   a. CD-ROMs
   b. magnetic tape
   c. punched cards and line printers

10. For what purpose was the IBM 1401 computer frequently used?

    a. It was often used as a web server.
    b. It was often used to copy jobs from punched cards to magnetic tape, and to copy results from magnetic tape to line printers.
    c. It was used as a telecommunications front-end processor.

11. Actions taken by a computer in processing a batch job were primarily controlled by what?

    a. Control cards such as $JOB, $LOAD, and $RUN
    b. A script (of sorts) selected by the setting of a group of switches on the front panel of the computer
    c. Commands from a control operator at a message forwarding system

12. Just before the introduction of IBM 360 computer systems, what were the main types of computers available?

    a. Number-crunching scientific systems and character-oriented systems for sorting and printing
    b. Systems that used vacuum tubes and systems that used transistors
    c. Systems that used wired plugboards for input, those that used punched cards, and those that used magnetic tapes

13. The first major computer to use integrated circuits was

    a. the Bendix G-15.
    b. the IBM 360.
    c. the IBM 7094.

14. Multiprogramming attempts to eliminate significant waste of what computer system resource(s)?

    a. CPU
    b. disks
    c. tape drives

15. Which of the following special features is needed to support multiprogramming?

    a. tagged memory architecture
    b. memory protection hardware
    c. fast carry look-ahead generators

16. What unrealistic goal was associated with the OS/360 operating system?

    a. It had to run well on small and large systems, on commercial and scientific systems, on systems with few and with many peripheral devices, and it had to be efficient in all cases.
    b. It had to provide emulation for programs that previously ran on systems with 32, 36, and 60 bits per word.
    c. It had to be sufficiently efficient so it never used more than five (5) percent of the CPU resource, even under heavy load on small low-end 360 systems.

17. To what does the acronym spooling refer?

    a. Simultaneous Peripheral Operations On Line
    b. Shared Partial Object-Oriented Linking
    c. Sequential Polling Of Operation Licensing

18. When a system uses spooling,

    a. programs are validated for execution by the presence of digital signatures called licenses.
    b. several sequential system calls from a program can be submitted as a group to save overhead during kernel entry.
    c. off-line systems to copy cards to tape and tape to print are not necessary.

19. How are timesharing and multiprogramming related?

    a. With timesharing, every user is given complete control of the system for a short time; multiprogramming only shares memory among the user programs.
    b. Timesharing is a variant of multiprogramming where each user has an online terminal.
    c. Timesharing and multiprogramming are different approaches to optimizing the use of primary memory which is typically underutilized if only a single program at a time is executed.

20. CTSS is the acronym for

    a. Compatible Time Sharing System
    b. Common Terminal Sharing Software
    c. CPU-Terminal Synchronization System

21. The MULTICS concept is most similar to which of the following?

    a. a world-wide shipping service
    b. a trash collection service
    c. a power distribution system

22. The MULTICS system was designed to run on which of the following processors?

    a. GE 645
    b. IBM 7094
    c. Cray I

23. In what programming language was MULTICS written?

    a. PL/I
    b. COBOL
    c. assembler language

24. One possible advantage of connecting a group of PCs or workstations via a LAN to a fileserver is that

    a. the PC or workstation processors do not need to be as powerful as the processor in the fileserver.
    b. the PCs can be used as a parallel system.
    c. the administrator has to install and protect only one set of programs and data.

25. The term middleware is used to describe

    a. operating systems used on systems that normally run a different operating system (for example, software that allows Windows executables to run on a Macintosh.)
    b. software used in heterogeneous environments to bridge the gap between local users and remote servers.
    c. mid-range servers and the software that runs on them.

26. In which generation of computers was the first minicomputer developed?

    a. the second generation
    b. the third generation
    c. the fourth generation

27. What company developed the first minicomputer?

    a. IBM
    b. DEC
    c. UNIVAC

28. What was the PDP-1?

    a. It was the first mainframe computer manufactured by Digital Equipment Corporation.
    b. It was the first minicomputer.
    c. It was the first system manufactured by Paradyne Data Products.

29. What was the last computer in the PDP series?

    a. the PDP-11
    b. the PDP-7
    c. the PDP-9

30. On which computer system was UNIX development first started?

    a. IBM 360
    b. IBM 7094
    c. PDP-7

31. What is POSIX?

    a. It is a standard for UNIX that defines a minimal system call interface.
    b. It is the hardware on which development of the UNIX operating system was developed.
    c. It is the first Apple operating system that ran on Intel processors.

32. In what year was the Intel 8080 introduced?

    a. 1968
    b. 1984
    c. 1974

33. The Intel 8080 is what class of computer?

    a. distributed computer
    b. microcomputer
    c. mainframe

34. What was the name of the most popular operating system used on Intel 8080 computer systems?

    a. AOS
    b. CP/M
    c. OS-X

35. What was the 6800?

    a. the first microprocessor to use LSI circuits
    b. the first microprocessor manufactured by Motorola
    c. the first 16-bit microprocessor

36. What was the 6502?

    a. the first 16-bit microprocessor
    b. the first microprocessor to use TTL circuits
    c. the microprocessor used in Apple II computer systems

37. Which microprocessor was used in the first IBM PC?

    a. the 8088
    b. the 6502
    c. the 8086

38. Which operating system dominated the early IBM PC market?

    a. MS-DOS
    b. CP/M
    c. OS-360

39. The acronym GUI stands for what?

    a. Gross Utility Index
    b. Graphical User Interface
    c. General User Interface

40. What processor was first used in the Apple Macintosh?

    a. the Sun SPARC
    b. an IBM RISC chip
    c. the Motorola 68000

41. The acronym ROM stands for what?

    a. RAID Only Multiplexor
    b. Relative Offset Mask
    c. Read Only Memory

42. What was the major distinguishing feature of the operating system for the Apple Macintosh?

    a. It provided a graphical user interface.
    b. It was the first operating system for a RISC processor.
    c. It was built on top of the Apple II's operating system.

43. The Microsoft Windows operating system

    a. was the first to run on a 32-bit processor.
    b. was originally a derivative of the NT operating system.
    c. was originally a GUI that ran on top of MS-DOS.

44. Which of the following operating systems is most frequently used on high-end computer workstations and network servers?

    a. UNIX (or a derivative)
    b. IBM's OS-390
    c. Microsoft Windows XP

45. The X Window system is

    a. the current Apple operating system for the Macintosh.
    b. the windowing system most commonly found on UNIX systems.
    c. the predecessor to the Y Window system.

46. What is a major difference between a network operating system and a distributed operating system?

    a. With a network OS, users are aware of the existence of multiple computers; a distributed OS appears to users as a traditional uniprocessor OS.
    b. Nothing; they are interchangeable terms for the same concept.
    c. A distributed OS always requires significantly more primary memory for each processor than a network OS.

47. The developer of Linux was significantly influenced by which of the following operating systems?

    a. MINIX
    b. VAX/VMS
    c. OS-390

48. The acronym LAMP is sometimes used to refer to

    a. Low-Allocation Memory Programs
    b. Linux, Apache, MySQL, and Perl or PHP.
    c. Linux API for MultiProcessors

49. *System calls* are

    a. calls from the operating system to device drivers.
    b. reports or requests from devices to the operating system communicating the results of an operation or a request for service.
    c. the extended instructions provided by an operating system.

50. The MULTICS operating system

    a. supported the concept of virtual machines with multiple virtual clock ticks occurring during every real clock tick.
    b. utilized hardware support for a series of concentric rings, with more privileged code executing running in smaller-numbered rings.
    c. None of the above.

51. One modern processor that provides hardware support for the ring mechanism is

    a. the IBM PowerPC.
    b. the Intel Pentium.
    c. all of the above.

52. The CMS system

    a. ran on the GE-645 processor which provided hardware support for the ring mechanism.
    b. is a single-user interactive system commonly used with VM/370.
    c. none of the above.

53. How many bytes of data were contained on a punched card?

    a. 256
    b. 80
    c. 100

54. A *process* is

    a. the address space associated with a running program.
    b. a program in execution.
    c. the microcode associated with a device controller.

55. A *process table* is

    a. an array in which a copy of the contents of the memory for a process is saved when that process is not running.
    b. a table that maps from user IDs to process IDs.
    c. an array or linked list of structures with one entry for each process in existence.

56. A *shell* is

    a. a process that reads commands and arranges for their execution.
    b. a process that is just a placeholder for another process that is being created.
    c. a representation of the memory region that will be used by a process that is being created.

57. Suppose a process successfully executes the system call to create a new process. That new process is usually referred to as a

    a. slave process.
    b. child process.
    c. ancillary process.

58. Which of the following will result in a signal being sent to a process?

    a. A process attempts to use an invalid address.
    b. A process executes an instruction that attempts to divide by zero.
    c. Each of the above will result in a signal being sent.

59. A *signal* is the software analog of

    a. a network packet arrival.
    b. a hardware interrupt.
    c. a system call.

60. Every user of a MINIX 3 (or UNIX) system is identified by

    a. a GID.
    b. a UID and a GID.
    c. a smart card with an embedded IC and hologram.

61. In UNIX, which user is endowed with special powers, and may violate many of the protection rules?

    a. the superuser
    b. the principal user
    c. the operator

62. A *directory* is

    a. a list of all users on a system, ordered by group.
    b. a list of all authorized users for a system.
    c. a way of grouping files together.

63. Entries in directories may be files or

    a. user IDs.
    b. process IDs.
    c. directories.

64. A hierarchy of directories gives rise to the concept of

    a. a volume label.
    b. a b-tree.
    c. a file system.

65. Each file in a directory hierarchy can be identified by giving its

    a. hash code.
    b. file name.
    c. path name.

66. The directory at the top of the directory hierarchy is called

    a. the root directory.
    b. the top directory.
    c. the primary allocation.

67. An absolute path starts with

    a. the root directory.
    b. the primary allocation.
    c. the first entry in the file system hash table.

68. If a path is not absolute, it begins with

    a. a positive or negative directory.
    b. the current working directory.
    c. the primary allocation.

69. A process can change its current working directory

    a. by closing the current working directory and opening the new working directory.
    b. by writing the new working directory name into the root directory.
    c. by issuing the appropriate system call.

70. The binary protection code in MINIX 3 has

    a. 10 bits.
    b. 8 bits.
    c. 11 bits.

71. Suppose the owner's *rwx* bits for a particular file are set to --*x*. This means the owner may

    a. execute the file.
    b. move the file to a different location in the directory hierarchy.
    c. read and write the file, but not execute it.

72. If the *x* permission bit is set for a directory, then

    a. the directory may be searched.
    b. the directory is encrypted.
    c. the directory is being exclusively used by a user.

73. A file descriptor is

    a. a small integer returned from a file open operation.
    b. the depth of a file in the directory hierarchy.
    c. a checksum of a file's contents used to detect tampering.

74. A *mounted* file system is

    a. one that replaces the previous root file system.
    b. copied into a predefined region of memory.
    c. attached to a directory using the *mount* system call.

75. A *special file*

    a. a file that contains a script that can be interpreted.
    b. makes an I/O device look like a file.
    c. one to which no user has read or write access.

76. A block special file

    a. is one that causes a process to block when it is read.
    b. can be used to model a disk.
    c. can only appear in a mounted file system.

77. A *pipe* is

    a. a stream that connects a process to a character device like a keyboard or a printer.
    b. a secure network communication channel.
    c. used to allow two processes to communicate by reading and writing the pipe.

78. A *shell* is

    a. an archaic name for a directory.
    b. an application program used for sorting.
    c. an application program that is not part of the operating system.

79. When a shell is given a user command to execute, it

    a. creates a child process to execute the command.
    b. checks to see if the user is logged in.
    c. checks to see if the owner of the file containing the program to be executed has explicitly allowed the current user to execute the command.

80. If a shell is given the command `who | wc -l`, it will

    a. connect the output of the `who` command to the input of the `wc -l` command and run both commands at the same time.
    b. run the `who` and `wc -l` commands at the same time, and terminate the longer-running command if the shorter-running command terminates successfully.
    c. first run the `who` command, then run the `wc -l` command.

81. Placing an ampersand after a command

    a. causes the output of the command to be discarded.
    b. causes the output of the command to be sent to the default printer.
    c. directs the shell to not wait for the command to complete before prompting for another command.

82. POSIX procedure calls generally return ____ to indicate an error.

a. NULL
b. 1
c. -1

83. If the *fork* system call returns successfully,

a. the process that called *fork* is potentially running a different executable program.
b. a child process will be created when the process that called *fork* terminates.
c. a child process has been created.

84. In most cases, after calling *fork* a new child process will

a. send a signal to its parent process notifying it that the process creation was successful.
b. wait for a signal from its parent to indicate it should begin executing.
c. call one of the *exec** system calls to load and execute a different program.

85. A shell, after creating a process to execute a command, will typically

a. wait for the child process to terminate.
b. delay a few seconds to allow the new process to get started.
c. wait for a signal from the child process.

86. The first argument to the function with which the execution of a process begins

a. a pointer to an integer that indicates the number of command line arguments.
b. is an integer that indicates the number of string arguments pointed to by entries in the second argument.
c. a pointer to the process table entry for the process.

87. The regions of memory into which a MINIX 3 process are divided are

a. allocated using memory that originally belonged to the parent process.
b. text (containing code), data (containing variables), const (containing constants), heap (containing dynamically-allocated storage), and stack.
c. text (containing code), data (containing variables), and stack.

88. The *sbrk* system call

a. is used to implement the *break* statement in C, C++, and Java.
b. is used to change the size of the data segment for a process.
c. is used to terminate the execution of a subroutine.

89. Which system call returns the process ID of the caller?

a. *rpid*
b. *getpid*
c. *pidget*

90. Which system call may be used by a process to announce that it is prepared to handle a particular type of signal?

    a. *registersig*
    b. *ssig*
    c. *sigaction*

91. What happens when a signal is sent to a process, and the process has indicated it is willing to handle that particular signal?

    a. The signal handler for the signal is called, if the signal's delivery has not been blocked.
    b. The current statement is completed, after which the function being executed returns to the signal handler.
    c. The signal handler for the signal is called.

92. What is the difference between *ignoring* a signal and *blocking* a signal?

    a. An ignored signal is discarded; a blocked signal remains pending and will be delivered if the signal is later unblocked.
    b. A blocked signal is delivered to a signal handler that always immediately returns; an ignored signal is never delivered to any signal handler. Thus blocked signals take longer to handle than ignored signals.
    c. There is no difference; they are different terms for the same concept.

93. Which of the following system calls may be used to send a signal?

    a. *kill*
    b. *signal*
    c. *transmit*

94. Which of the following system calls may be used to create a file?

    a. *new_file*
    b. *creat*
    c. *create*

95. Which of the following system calls may be used to create a special file?

    a. *mknod*
    b. *newspec*
    c. *crspec*

96. Which of the following system calls is probably the most heavily used?

    a. *open*
    b. *dup*
    c. *read*

97. The *read* and *write* system calls have three parameters. The second parameter is

    a. a pointer to a string containing the path of the file to be read or written.
    b. a pointer to an integer into which will be stored the number of bytes actually read or written when the call completes.
    c. a pointer to a buffer.

98. The *lseek* system call is used to

    a. repeat the last *seek* system call's operation.
    b. locate a point in a file from which a *seek* operation can be issued.
    c. specify the file offset of the next byte in a file to be read or written.

99. The *dup* system call

    a. duplicates the contents of a region of memory by dynamically allocating a region of memory the same size and copying the contents of the region to it.
    b. duplicates the contents of a file and gives it a new file name.
    c. returns a file descriptor that references the same file as another that referenced by another open file descriptor.

100. The *pipe* system call, if successful,

    a. creates a virtual file that can then be opened for reading and writing.
    b. is only used by a shell to connect the standard output of one command to the standard input of another.
    c. results in the creation of two open file descriptors, one for reading, and one for writing.

101. Which of the following system calls may be used to change a terminal's mode to *raw*?

    a. *ioctl*
    b. *setraw*
    c. *tcsetattr*

102. When a terminal is in raw (or non-canonical) mode

    a. every input character is passed directly to a reading program with no special processing.
    b. every input character is passed directly to a reading program except for the end of line character.
    c. no characters are passed to a reading program until an end of line character or a character that generates a signal has been entered.

103. When a terminal is in cooked (or canonical) mode

    a. every input character is passed directly to a reading program after checking for characters that might result in the generation of a signal.
    b. input characters are not passed to a reading program until the end of line character or a character that generates a signal has been entered.
    c. end of line characters are converted to carriage returns.

104. When the control-C character is typed on a terminal's keyboard

    a. the system's response depends on the terminal mode.
    b. the control-C is converted to an empty line (that is, a line containing only an end of line character).
    c. the process associated with the terminal is terminated.

105. Which of the following system calls can be used to control advisory file locking?

    a. *open*
    b. *fcntl*
    c. *fattr*

106. The *rmdir* system call is used to

    a. obtain a remote listing of the contents of a directory.
    b. remove a directory if it is empty.
    c. remove a directory and any files it contains.

107. In traditional UNIX/LINUX/MINIX file systems, a *link*

    a. may be used to allow a file to have multiple names, often in multiple directories.
    b. a pointer (perhaps one of many) identifying the disk addresses of the blocks comprising a file.
    c. an alternate name for an *i-node*.

108. A *link* system call, if successful,

    a. moves a directory entry from one directory to another.
    b. changes the size of a file.
    c. makes a new entry in a directory.

109. In UNIX/LINUX/MINIX, a directory is essentially

    a. a set of (disk locations, ASCII name) pairs.
    b. a set of (i-number, ASCII name) pairs.
    c. a regular file with contents arranged in a well-defined format.

110. In many operating systems a *block cache* is used to

    a. record a list of modifications made to files so they can be repeated if the system or device should later fail.
    b. keep copies of recently used blocks to avoid having to repeatedly read them.
    c. keep copies of blocks that have been read from a device and not modified.

111. In MINIX 3 (and other similar UNIX systems), the *sync* system call

    a. causes the heads of disks, CD/DVD drives, and similar devices with moving media to be returned to a reference point (cylinder zero) to allow the position of the heads to be validated.
    b. tells the system to begin writing all modified blocks back to the devices from which they were originally read.
    c. synchronizes the system time with the time from a well-known stable time source.

112. The *chdir* system call

    a. changes the name of a directory.
    b. changes the permissions associated with a directory.
    c. changes the current working directory for a process.

113. The *chroot* system call changes

    a. the permissions associated with the root directory.
    b. the identity of the superuser.
    c. the interpretation of absolute path names for a process.

114. The *chmod* system call

    a. switches the hierarchical mode of a file system between hierarchical and flat.
    b. is used to change the state of a dynamically linked kernel module.
    c. changes the protection bits associated with a file.

115. When a program marked SETUID is executed

    a. the process acquires an effective UID potentially different from its real UID.
    b. its UID is check to ensure it matches that of the owner of the directory in which the program's executable file is located.
    c. None of the above.

116. The *umask* system call

    a. can be executed by ordinary user processes.
    b. removes any mask associated with the UID or GID of the owner of an executable file.
    c. masks the UID and GID associated with the owner of an executable file.

117. The *access* system call

    a. tests if a particular type of file access will be permitted for the real owner of a process.
    b. vacuously accesses a file to cause the time of last access in the file's metadata to be updated.
    c. None of the above.

118. The *time* system call

    a. returns the current time, in seconds, since midnight on on January 1, 1970.
    b. returns the total CPU time used in user and kernel mode by the current process.
    c. None of the above.

119. The most common operating system structure is called

    a. a layered system.
    b. an exokernel.
    c. a monolithic system.

120. When an monolithic operating system is constructed,

    a. the files containing the code for the system calls used by each program are compiled and linked with the individual programs.
    b. all the files containing the code are compiled and linked together in a single object file.
    c. the individual functions are compiled, but are not linked with the remainder of the operating system kernel until a process requrests use of a service provided by that function.

121. A supervisor or kernel call instruction

    a. switches the machine from user mode to kernel mode and transfers control to the operating system.
    b. can only be executed when it is placed in a memory location that is accessed using a real (not virtual) memory address.
    c. operates exactly like a regular function call instruction, but also verifies that the calling program is executing in user mode.

122. In a monolithic system, it is common for the supervisor or kernel call instruction, or trap instruction,

    a. to result in the state of the executing process, including its memory image, to be saved on disk.
    b. to be located in a library procedure, possibly written in assembly language.
    c. None of the above.

123. The THE operating system was

    a. a simple monolothic system that provided the first timesharing facilities in any operating system.
    b. a simple batch system organized as a set of layers.
    c. organized as a smaller kernel that provided message passing services and a set of client applications that provided the typical operating system services.

124. In a layered system, code running in a particular layer

    a. can only call services in the layers above it.
    b. executes in user mode, and the code in all other layers *at that time* is treated as being executable only in kernel mode.
    c. can only call services in the layers below it.

125. Each *mount* system call, if successful,

    a. allows the files in one file system to be accessed through a directory in another file system.
    b. dynamically creates a new drive letter (C:, D:, etc.) and makes the files on the media in a drive available for reading and writing.
    c. partitions a physical disk drive into multiple separate regions, each of which contains a separate file system.

126. In a monolithic system, system calls provided by the operating system

    a. are requested by sending a network packet.
    b. are linked to every application program.
    c. are requested by executing a kernel call or a supervisor call.