CSCI 331: Operating Systems I

Midterm # 1 -- closed book section

The entire exam is to be turned in at 4:20PM. Work the closed book section first and turn it in before you consult your books and notes to work on the open book section. For the closed book section, write your answers on the exam itself.

There are twenty-five questions. Each is worth two points. Choose the most appropriate phrase for terminating each sentence. Circle your choice.

In most operating system implementations, a system call is initiated by:
A: a recursive subroutine invocation.
B: a special hardware **trap** instruction.
C: ``executing'' an illegal instruction.
D: creating a new process.

A likely value for a quantum is
A: 100 milliseconds.
B: 1 millisecond.
C: 10 seconds.
D: 10 microseconds.

In operating systems, a process is
A: program code.
B: a running program.
C: a compiled program.
D: a sequence for starting a program.

Critical sections of code
A: must be executed quickly.
B: must be executed by at most one process at a time.
C: must be protected from user access.
D: have a half-life of five years.

One of the reasons very early machines were used inefficiently was that
A: the setup time between jobs was great.
B: their operating systems did not support process creation.
C: semaphores often deadlocked.
D: operating system designers were poorly paid.

Multiprogramming is
A: several programmers writing different sections of a large program.
B: several processes sharing a single computer at the same time.
C: programming in more than one computer language.
D: a job consisting of several steps, *e.g.*, a compile followed by a load.

The code of an ordinary program, *e.g.*, a text editor, is executed in:
A: monitor mode.
B: modify mode.
C: user mode.
D: kernel mode.

In a batch system, a job is a
A: a way to earn a leaving.
B: a set (sequence) of user-submitted programs.
C: an operating system task.
D: an on-line command parser.

Programmers using a time-sharing system
A: interact with the computer with on-line terminals.
B: sign up for slots to use the console.
C: submit jobs to operators at remote sites.
D: must be careful not to use other programmer's shares.

The MULTICS operating system was designed at
A: Bell labs.
B: Digital Equipment Corporation.
C: International Business Machines.
D: Massachusetts Institute of Technology.

The Minix program **grep** can be used to
A: create directories.
B: search files for a particular string.
C: recursively list the contents of a directory.
D: compile and link grep scripts.

The Minix command for compiling a C program is
A: **make**.
B: **c**.
C: **cc**.
D: **compile**.

In Minix the shell
A: controls the actions of the kernel.
B: encloses the kernel.
C: is a command interpreter.
D: is started at boot time.

In Minix a file descriptor is
A: a C structure giving important attributes of the file.
B: an integer used to identify the file for system calls.
C: a name for locating the file within the file structure.
D: a pointer to an I/O buffer.

In Minix, a special file
A: is protected against writing by all users.
B: can not be deleted.
C: is owned by the superuser.
D: corresponds to a specific device.

The number of system calls in Minix is approximately
A: 5.
B: 50.
C: 500.
D: 5000.

The text segment of a Minix process contains:
A: ASCII code.
B: compiled code.
C: program documentation.
D: operating system documentation.

In Minix, files can be protected by:
A: logging in as root.
B: sending mail to the superuser.
C: setting the mode bits.
D: setting the hidden attribute.

A virtual machine is:
A: an operating system which allows users to share several processes.
B: an operating system interface that mimics a hardware architecture.
C: an operating system mechanism for isolating user processes.
D: an operating system which runs on many different hardware architectures.

The scheduler of a operating system decides
A: which system call to execute next.
B: which process to execute next.
C: which user may log in next.
D: which print job should be spooled next.

The root password on your Minix disk is
A: Watchwoort
B: Geneva
C: Geheim
D: SuperUser

The simplest scheduling algorithm is
A: shortest job first.
B: longest job first.
C: round robin.
D: priority based scheduling.

A process is blocked when it is
A: stuck in an infinite loop.
B: performing a system call.
C: executing the **sleep** system call.
D: waiting for an external event to occur.

The **fork** system call
A: returns 0 to the parent process.
B: loads a new program into memory.
C: creates a new process.
D: splits the program control sequence.

The **dup** system call
A: duplicates a process.
B: duplicates a file descriptor.
C: duplicates a pipe.
D: duplicates a file.