Midterm 2–April 6
Closed book section (64 points)

The exam is to be turned in at 2:50 pm. 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.

University regulations require that you sign the following pledge on the first page of your turned-in exam.

I have neither received nor given any unauthorized aid on this exam. _____

Problem 1. (40 points–2.5 points each)

Circle the most appropriate completion for each sentence.

Question i. A mode switch *always* occurs when
A. a signal is delivered.
B. a user process makes a system call.
C. a device interrupt happens.
D. a process modifies `argv[0]`.

Question ii. The page size of a modern computer is likely to be
A. one-tenth the size of available memory.
B. a power of two.
C. a prime number.
D. the size of a track on the paging disk.

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

Question iv. In Unix, the number of context layers is limited by
A. avoiding recursive calls in the kernel.
B. the number of runnable processes
C. the number of interrupt levels.
D. having a fixed number of arguments to all system calls.

Question v. The *environment,* passed as the third argument to the `main` routine, is likely to contain
A. the code of the process.
B. the current working directory.
C. the name of prefered printer of the owner of the process.
D. the open files of a process.

Question vi. The Unix system call `kill` will
A. stop a process (or processes).
B. terminate a process (or processes).
C. send a signal to a process (or processes).
D. delete a file.

Question vii. Setuid bits are permently stored in
A. the buffer pool.
B. file inodes.
C. home directories.
D. superuser installed programs.

Question viii. Usually, the code of the operating system kernel is
A. loaded by a *bootstrap* program.
B. loaded by the superuser.
C. stored in read-only memory.
D. stored on a read-only file system.

Question ix. A datagram socket may be used to
A. send and receive messages.
B. send messages.
C. receive messages.
D. pack data into records.

Question x. A likely value for a *time slice* is
A. 10 seconds.
B. 100 milliseconds.
C. 1 millisecond.
D. 10 microseconds.

Question xi. In most Unix schedulers, the priority of a running process decreases
A. logarithmicly.
B. linearly.
C. exponentially.
D. randomly.

Question xii. Most likely, the line printer daemon is started when
A. Duke and Wake Forest play basketball.
B. the system is booted.
C. the superuser logs in.
D. a file is printed.

Question xiii. When the parent of a process terminates, the child process
A. is adopted by the *init* (#1) process.
B. is terminated.
C. is adopted by a grandparent process.
D. becomes a zombie.

Question xiv. Within a process, the `wait` system call may be used to
A. change user id.
B. block until terminal I/O is available.
C. sleep until wakeup occurs.
D. suspend until a child process terminates.

Question xv. The two major socket domains in the BSD system are
A. Unix and System V.
B. stream and datagram.
C. Unix and internet.
D. local and remote.

Question xvi. In BSD-based operating systems, a socket is
A. a small integer identifying a pipe.
B. an endpoint for communication.
C. a place to plug in the computer.
D. a function for handling signals.

Problem 2. (24 points–4 points each)

Question i. List three possible *events* on which a Unix process may sleep.

Question ii. List three kinds of kernel data structures whose reference counts will be incremented with a new process is `fork`ed.

Question iii. List three ways two processes on the same machine can communicate.

Question iv. List the names of the three main regions of a Unix process.

Question v. List three possible states, other than "zombie," of a process.

Question vi. List the three system calls that must be executed by a stream server process *before* it can accept a connection.