Advanced Programming in the Unix Environment
CS765B — Fall 2001
Final Examination

INSTRUCTIONS: Answer all questions. Answers for the multiple-choice section should be *written neatly in a blue book*, NOT circled or otherwise indicated on the exam printout. Write the letter of your answer, not the text of the answer itself. Answers for the long-form section should also be written neatly in a blue book. You may use more than one blue book if necessary. Write your name and student ID number on the cover of all materials.

This exam is open-book, but you may not share another student's book. The only permitted books are *Advanced Programming in the Unix Environment* by Stevens and *The C Programming Language* by Kernighan and Ritchie. You may not use your class notes or any other materials for reference.

Multiple-Choice Section

1) What is another system call that provides the same functionality as creat()?
   a) link()
   b) send()
   c) open()
   d) dup()

2) What is the difference between the dup() and dup2() system calls?
   a) dup() copies a file; dup2() renames it.
   b) dup() closes its argument; dup2() copies its first argument to its second argument.
   c) dup2() copies its first file descriptor argument to its second file descriptor argument, closing the second file descriptor if it is open. dup() returns a new file descriptor which is a copy of the file descriptor given as its argument.
   d) dup()copies its first file descriptor argument to its second file descriptor argument, closing the second file descriptor if it is open. dup2() duplicates a file descriptor given as its first argument to two new file descriptors given as its second and third arguments.

3) What is the difference between a symbolic link and a Òhard linkÓ?
   a) A hard link can point to any kind of file, whereas a symbolic link can point only to a regular file or a directory.
   b) A symbolic link is another name for an inode in the filesystem; it can reference any file. A hard link stores the pathname of another filesystem node as its file data and can refer to only a plain file.
   c) Pre-POSIX systems support only symbolic links.
   d) A symbolic link stores a pathname as its file data and may reference any filesystem node, or even no filesystem node. A hard link is an additional directory entry referring to the same filesystem node as the first name given to any particular file.

4) A file is created with mode S_IWUSR|S_IRGRP|S_IROTH and umask 0. Who can read the file?
   a) Any user except the user who created the file.
   b) Only the user who created the file.
   c) Any user.
   d) Only users in the same group as the user who created the file.

5) Who can write to the file described in the previous question?
   a) Any user except the user who created the file.
   b) Only the user who created the file.
   c) Any user.
   d) Only users in the same group as the user who created the file.

6) How is a file with a "hole" in it created?
   a) The file is opened with the O_LARGEFILE flag.
   b) lseek() is used to move the file pointer beyond the current end of the file.
   c) The truncate() system call is used to remove space from the file.
   d) Both b) and c) are correct.

7) How is it possible to rename a file without using the rename() system call?
   a) Using the pathalias() and unlink() system calls
   b) Using the link() and unlink() system calls
   c) Using the open() and fchdir() system calls.
   d) Using the stat() and namei() system calls.

8) What does the fchdir() function do?
   a) Set the process' current working directory to the directory referred to by the file descriptor argument to the function
   b) Set the process' current working directory to the filesystem default path.
   c) Move the file referred to by the file descriptor argument to the directory given by the pathname argument.
   d) Move the file referred to by the pathname argument to the directory given by the file descriptor argument.

9) What function can be used to set the modification time of a file?
   a) mtime()
   b) maketime()
   c) stat()
   d) utimes()

10) Which two functions could be used to convert a time_t into a struct tm?
   a) utimes() and gettimeofday()
   b) time() and gettimeofday()
   c) gmtime() and localtime()
   d) strftime() and times()

11) How is it possible to find all members of group "bob"?
   a) By using getgroups() to walk the group database
   b) By using getpwent() to walk the password database
   c) By using getgrnam("bob") and using getpwent() to walk the password database
   d) By using getgroups() to walk the group database and getpwent() to walk the password database

12) Calling abort() is similar to:
   a) kill(getpid(), SIGABRT)
   b) kill(getppid(), SIGABRT)
   c) kill(getpgid(), SIGABRT)
   d) kill(getpid(), SIGKILL)

13) How are the wait() and waitpid() functions related?
    a) wait() provides a superset of the functionality of waitpid()
    b) waitpid() provides a supserset of the functionality of wait()
    c) wait and waitpid() are both implemented using waitfor()
    d) waitpid() can use an alternate process stack.

14) Process A forks, creating process B. Process A then exits, as does its parent, and the parent of its parent, and the parent of its parent. What will probably happen to Process B when it exits?
    a) Process B will become a zombie.
    b) Process A will become a zombie.
    c) Process B will be reaped by process 1, init.
    d) Process A will return from the zombie state.

15) What are two ways to overlay the current running process with a new executable?
    a) execve(), execlp()
    b) execle(), execvep()
    c) execle(), system()
    d) execve(), vfork()

16) What does the spawn() system call do?
    a) spawn() is another name for vfork()
    b) spawn() creates a new task
    c) spawn() creates a new thread
    d) There is no spawn() system call.

17) What does the signal() system call do?
    a) Sends a signal to another process
    b) Installs a signal handler
    c) Sends a signal to a related process
    d) There is no signal() system call

18) What does the sigprocmask() system call do?
    a) Examines and/or changes the set of signals blocked from current delivery
    b) Examines and/or changes the set of signals that may be sent
    c) Polls for signals which may have been blocked upon delivery
    d) There is no sigprocmask() system call

19) How is a terminal put into Òcanonical Ó or Òcooked Ó mode?
    a) By turning off all input processing, including ISIG, IEXTEN, and ICRNL.
    b) By setting ICANON in c_lflag in struct termios, and installing the terminal modes.
    c) Using tcsetent()
    d) Using ioctl(fd, 0, TCSANOW)

20) What function is used to open a a TCP connection?
    a) open()
    b) connect()
    c) sendmsg()
    d) send()
21) What is a correct prototype for the select() function?
    a) int select(int nfds, fd_set readfds, fd_set writefds, fd_set exceptfds, struct timeval timeout);
    b) int select(int nfds, fd_set *reafds, fd_set *writefds, fd_set *exceptfds, time_t *timeout);
    c) int select(int nfds, FD_SET *readfds, FD_SET *writefds, FD_SET *exceptfds, struct timeval *timeout);
    d) int select(int, fd_set *, fd_set *, fd_set *, struct timeval *);

22) Which two functions can be used to receive a datagram message?
    a) read(), recvfrom()
    b) recvmsg(), recvfrom()
    c) read(), select()
    d) listen(), accept()

23) How does a socketpair differ from a pipe?
    a) A socketpair is bidirectional; a pipe is unidirectional.
    b) A pipe is full-duplex, whereas a socketpair is half-duplex.
    c) A socketpair preserves record boundaries, whereas a pipe is a stream file.
    d) There is no difference between a pipe and a socketpair.

24) What does the accept() system call do?
    a) The accept() system call returns a new file descriptor for a pending connection to a stream socket, removing that connection from the queue of pending connections.
    b) The accept() system call listens for a new connection to a stream socket.
    c) The accept() system call listens for a new connection to a datagram socket.
    d) The accept() system call sets the permissions for an AF_UNIX socket, so that only authorized connections can be made.

25) What does the ntohl() function do?
    a) Converts a 16-bit value from host to network byte order
    b) Converts a value of type ÒlongÓ from little-endian to big- endian byte order.
    c) Converts a 32-bit value from network to host byte order
    d) Converts a 32-bit value from little-endian to big-endian byte order.

Long-Form Section

Write a small Unix shell. The shell is required to wait for command-line input and support the following syntax:

command	**run the executable named ÔcommandÕ**
command >file	**run the executable named ÔcommandÕ with its standard output redirected into the file named ÔfileÕ, which is created if necessary.**
command1 | command2	**run the executables named ÔcommandÕ and Ôcommand2Õ, connecting the standard output of Ôcommand1Õ to the standard input of Ôcommand2Õ**
cd dir	**change the current working directory of the shell to directory ÔdirÕ**

Your shell must emit an appropriate error message if a file or directory does not exist or cannot be created or executed. Your shell must print the exit status and reason of each of its child processes.

Your shell should exit when it receives EOF on its standard input, or if an error occurs reading from standard input or writing to standard output or standard error.

No credit will be given for copying the example program from the textbook into your answer.