FINALS EXAM – Sections 501-504

(200 points)

This exam has a total of 8 questions, spread over 7 pages, including this cover page.

Date: December 12, 2014

Student Name: .....

Student ID: .....

You are allocated a maximum amount of space to answer each question. (We have provided sufficient lines.) Adhere to those limitations when you formulate your answers. Do not use the backside of the pages; no additional pages are allowed.
Make an effort to write in a readable fashion. We will skip over (and therefore not grade) non-readable portions.

"I have adhered to the Aggie Code of Honor."

Signature:
.....

1	(22)
2	(33)
3	(25)
4	(20)
5	(25)
6	(20)
7	(25)
8	(30)
T	(200)

1. [22 points] Match each term in the left column to the definition/description in the right column that **fits best**. Do this by filling in the void entries on the left:

( )	File Descriptor	(A)	Files share iNODE
( )	SIGSTOP	(B)	CTRL-Z
( )	Unix Pipe	(C)	Port 80
( )	Authentication	(D)	Big Endian Format
( )	IP	(E)	Files have unique inodes (one of the inodes only contains filename)
( )	FIFO	(F)	Integer number identifying a file connection
( )	Host Byte Order	(G)	One-way data channel in the Kernel
( )	Web Server	(H)	Lives in Kernel space and entries contain offset and ref count among other attributes
( )	Hard Link	(I)	Lives in Main Memory and facilitates IPC through file descriptors
( )	File Table	(J)	Protocol for Network Layer in Internet Programming
( )	Soft Link	(K)	Goal of a Security policy

2. [33 points] Circle the following statements TRUE or FALSE.

T F Masquerading is a security attack
T F The contents of a file descriptor table are pointers to entries in the file table
T F Doubling the block size of a unix file system will double the max file size
T F Every new file connection results in a unique entry in the file table.
T F Unix I/O is the most general and lowest overhead form of I/O
T F A file permission config of 766 allows "group" users write access to the file
T F A refcnt of 2 in a vnode table implies a file of size 2 blocks
T F A named pipe "foo" has an inode assigned on permanent storage
T F Calling open twice with the same filename results in 1 file table entry with a refcnt of 2
T F Shared memory for IPC can reside inside the address space of the creating process
T F When a signal handler is invoked that type of signal must remain unblocked to continue to catch new signals

3. [25 POINTS] What would be the output of the program below? Assume it writes to a file: RESULT

Rubric: 10points for correct answer and 15 points for supporting explanation

```
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <sys/types.h>
#include <fcntl.h>
#include <errno.h>
#include <sys/stat.h>
#define READ_FLAGS O_RDONLY
#define APPEND_FLAGS (O_APPEND | O_WRONLY)
#define WRITE_FLAGS (O_RDWR | O_CREAT | O_TRUNC)
#define WRITE_PERMS (S_IRUSR | S_IWUSR)
int main(int argc, char *argv[])
{
  int fd1, fd2, fd3;
  char *fname = argv[1];
  fd1 = open(fname, WRITE_FLAGS, WRITE_PERMS);
  write(fd1, "CSCE 313", 8);
  fd3 = open(fname, APPEND_FLAGS, 0);
  write(fd3, "teaches", 7);
  fd2 = dup(fd1);  /* Allocates new descriptor */
  write(fd2, "System", 6);
  write(fd3, "Programming", 6);
  close(fd1);
  close(fd2);
  close(fd3);
  return 0;
}
```

4. [20 POINTS] What would be the output of this program that reads an input file with content: **GIG'EM**
Rubric: 10points for correct answer and 10 points for supporting explanation

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <sys/types.h>
#include <fcntl.h>
#include <errno.h>
#include <sys/stat.h>

int main(int argc, char *argv[])
{
  int fd1;
  int s = 0x1;
  char c1, c2;
  char *fname = argv[1];
  fd1 = open(fname, O_RDONLY, 0);
  read(fd1, &c1, 1);
  if (fork()) { /* Parent */
    sleep(s);
    read(fd1, &c2, 2);
    printf("Parent: c1 = %c, c2 = %c\n", c2, c1);
  } else { /* Child */
    sleep(1-s);
    read(fd1, &c2, 1);
    printf("Child: c1 = %c, c2 = %c\n", c1, c2);
  }
  return 0;
}
```

5. [25 POINTS]

a. [15 POINTS] Using system calls discussed in class, write a pseudo code for a program that accomplishes the following in UNIX: % **whoami | less > result**.
Note: % is the UNIX shell prompt

b. [5 POINTS] What would be the following function result in:
kill (getppid (), SIGTERM)

c. [5 POINTS] What 3 actions can a process take in response to a SIGNAL?

6. [20 POINTS] Consider a file system with 2048 Byte blocks and 4 Byte disk block pointers to those blocks. Each file header has 12 direct pointers, 1 singly-indirect pointers, 1 doubly-indirect pointer, and 1 triply-indirect pointer. In the following, please be explicit about your work:

a.[5 points] How large of a disk can this filesystem support? Explain. You may leave your answer in symbolic form.

b.[10 points] What is the maximum file size? Explain. You may leave your answer in symbolic form.

c. [5 points] Make some reasonable assumptions and compute the number of INODES that can fit into a disk block.

7. [25 POINTS] List the set of disk blocks that must be read into memory in order to read the file /home/faculty/tyagi/csce313/demo.txt in its entirety from a UNIX file system which has 10 direct pointers, a singly-indirect pointer, a doubly-indirect pointer, and a triply-indirect pointer. Assume that the file demo.txt is 15,234 Bytes long and that disk blocks are 1024 Bytes long. Assume that the directories in question all fit into a single disk block each. Show your work.

8. [30 Points]

a. [10 Points] Draw a typical TCP client and server connection setup. Clearly show the system calls that must be executed on each side to establish and complete a TCP connection.

b. [20 Points] Write a client program and a server program to return the number of processes currently running on a specified host computer. Write your pseudo program using TCP system calls. **State all your assumptions clearly.**