CSCI 4210- Operating System Fundamentals

Sample Final Exam:

Name: ______________________________

Email Address: ________________________

This is a open-book, open notes exam. You have 180 minutes to answer as many questions as possible. The number in parenthesis at the beginning of each question indicates the number of points given to the questions (and that also indicates approximately how many minutes you should spend on the question). Write all your answers directly on this paper. Make your answers as concise as possible.

Problem	Points Scored	Points Possible
1	________	20
2	________	10
3	________	10
4	________	10
5	________	10
6	________	10
7	________	10
8	________	10
9	________	10
Total		

1. (20 points) Indicate whether each of the statements below is True(**T**) or False(**F**), and explain your answer *briefly*.

(a) A page table provides protection in the capability style, not in the access-list style.

(b) A multi-level indexed file descriptor permits faster random access than a file descriptor with a single level of indexing.

(c) In UNIX, user processes can read and write directories just like ordinary files.

(d) The dispatcher is the part of the Operating System that determines the priority of each process.

(e) Binary semaphores are those that are used by no more than two processes.

(f) In a system with only one independent processes, deadlock cannot occur.

(g) If the working set mechanism is used for demand paging, the processor can sit idle even though there are runnable processes.

(h) Different threads can share global memory locations.

(i) When communication is through messages it is possible for processes to communicate even if both send and receive are non-blocking.

(j) Page sizes are always a power of 2.

(k) The more cache memory the machine has, better performance it provides.

(l) Reference count algorithm for reclaiming memory overcomes the problem of recovering memory allocated to circular structures.

(m) A garbage collection algorithm requires at least two passes

(n) A fully connected (links between every pair of nodes) is more reliable than a ring network.

(o) Security and Protection are one and the same.

(p) In disk scheduling, the SSTF (shortest seek-time first) algorithm can lead to starvation.

(q) Network load will NOT affect the response time in a distributed system.

(r) The routing tables mechanism for Internet is an example of an adaptive system.

(s) Efficiency of Algorithms for Distributed Computation problems depends upon the network interconnection

(t) In UNIX system, the remote procedure calls are used to execute commands on different machines in a network.

(2a) (5 points) List five services provided by an operating system. Explain how each provides convenience to the users.

(2b) (5 points) Using semaphores, write a program (in some pseudo language with concurrent begin and concurrent end) to compute the following program graphs. The nodes are tasks and an edge from t_i to t_j means that task t_j has to wait for the completion of task t_i.

(3) (10 points) A file is to be shared among different processes, each of which has a unique number. The file can be accessed simultaneously by several processes, subject to the following constraint: The sum of all unique numbers associated with all processes currently accessing the file must be less than n. Write a monitor to co-ordinate the access to the file. You may want to write start_access and end_access monitor procedures. (The monitor solution can be written in a pseudo language.)

(4a) (6 points) Consider a paging system with the page table stored in memory.

1. If a memory reference takes 200 nanoseconds, how long does a paged memory reference take?
2. If we add associative registers, and 75% of all page-table references are found in the associative registers, what is the effective memory reference time? (assume that finding a page-table entry in the associative registers takes zero time, if the entry is there.)

(4b) (4 points) Consider a logical address space of eight pages of 1024 bytes each, mapped onto a physical memory of 32 frames.

1. How many bits are there in the logical address?
2. How many bits are there in the physical address?

(5a) (2 points) Assume you have a page reference string for a process with m pages (initially all empty). The page reference string has length p with n distinct page numbers occurring in it. For any page replacement algorithm

(i) What is the minimum number of page faults?

(ii) What is the maximum number of page faults?

(5b) (8 points) Consider the following sequence of memory references from a 460 byte program: 10, 11, 104, 170, 73, 309, 185, 245, 246, 434, 458, 364.

(i) Give the reference string, assuming a page size of 100 bytes.

(ii) Find the page-fault rate for the reference string in part a, assuming 200 bytes (2 pages) of main memory i available to the program and a FIFO replacement algorithm.

(iii) Calculate the page-fault rate if the page replacement algorithm is LRU.

(iv) Calculate the page-fault rate for the MIN page replacement algorithm.

(7a) (6 points) Consider a system that supports 5000 users. Suppose that you want to allow 4990 of these users to be able to access one file.

1. How could you specify this protection in UNIX?
2. Could you suggest another protection scheme that can be used more effectively than the one provided by UNIX?

(7b) (4 points) What are the main differences between capability lists and access lists? How do you change the privelages in capability lists?

(8a) (2 points) What security measures will you use to protect your files?

(8b) (8 points) Consider a distributed system with two sites A and B. Consider whether site A can distinguish among the following:

1. B goes down.
2. The link between A and B goes down.
3. B is extremely overloaded and response time is 100 times longer than normal.

What implications does your answer have for recovery in distributed systems?

(9a) (5 points) What are the advantages and distributed of having a distributed file systems?

(9b) (5 points) The following algorithm computes the maximum of pids in a unidirectional ring network of processes. Modify this algorithm so that it calculates the maximum in a bidirectional ring network. Compare the time steps it takes to compute the maximum with that of unidirectional ring network algorithm.

```
Each Process: sends send to its neighbor:

send:=null
if the incoming message is v, a pid, then
case
v>u: send := v
v=u: status:= leader
v<u: do nothing
endcase
```