University of California, Berkeley
College of Engineering
Computer Science Division – EECS

Spring 2003 Anthony D. Joseph

Midterm Exam #2

April 29, 2003
CS162 Operating Systems

Your Name:	
SID AND 162 Login:	
TA:	
Discussion Section:	

General Information:
This is a **closed book and notes** examination. You have two hours to answer as many questions as possible. The number in parentheses at the beginning of each question indicates the number of points given to the question; there are 100 points in all. You should read **all** of the questions before starting the exam, as some of the questions are substantially more time consuming.

Write all of your answers directly on this paper. ***Make your answers as concise as possible.*** If there is something in a question that you believe is open to interpretation, then please ask us about it!

Good Luck!!

Problem	Possible	Score
1	16	
2	25	
3	24	
4	10	
5	25	
Total	**100**	

1. (16 points total) Short answer questions:
   a. (2 points) What UNIX structure is used to keep track of the sectors allocated to a given file?

   b. (2 points) What is the smallest addressable piece of data on a disk drive?

   c. (2 points) What is a persistent, named collection of data?

   d. (2 points) What is a piece of hardware that caches virtual page → physical page mappings?

   e. (8 points) Consider two processes P and Q that are communicating using mailboxes. From P → Q they use mailbox A, and from Q → P they use mailbox B. Assume both mailboxes are currently empty and the communication link between P and Q is reliable. If P now wants to determine that Q has terminated, what is the sequence of instructions that P should execute?
   You may only use the following blocking instructions: `send`, `reply`, `re-send`, `receive`, `deleteMailbox`, `createMailbox`, and `setAlarm`. Note that an alarm will interrupt a blocking instruction.

2. (25 points total) Disk Drives and File Systems.

a. (6 points) Itsy Bitsy Machines Corporation develops a new disk drive that has a separate read/write head for each track and thus its disk heads don't move. Which of the following file-system features are no longer important?
State whether the feature is useful or not useful and ***in 1 – 2 sentences*** explain why *or* why not.
i) Cylinder groups (e.g., Unix 4.2 BSD):

ii) A bitmap free list (e.g., Unix 4.2 BSD)

b. (10 points) List the set of disk blocks that must be read into memory in order to read the file `/home/cs162/test.doc` in its entirety from a UNIX BSD 4.2 filesystem. Assume the file is 15,234 bytes long and that disk blocks are 1,024 bytes long. Assume that the directories in question all fit into a single disk block each. *Note that this is not always true in reality.*

Problem 2 (cont'd)

c. (9 Points) Suppose a file system can have three disk allocation strategies, contiguous, linked, and indexed. We have just read the information for a file from its parent directory. For contiguous and linked allocation, this gives the address of the first block, and for indexed allocation this gives the address of the index block. Now we want to read the 10^{th} data block into the memory. How many disk blocks (*R*) do we have to read for each of the allocation strategies? *For partial credit, explicitly list which block(s) you have to read.*

Contiguous allocation:

$R =$

Linked allocation:

$R =$

Indexed allocation:

$R =$

3. (24 points total) Adding Links to a File System.
This design question asks you to consider adding links to a file system. **This is a design question; you should not write code.**

The first set of questions is about adding **hard** links to the file system.

a. (4 points) What changes would you make to the internals (directory, fileheader, free map, etc.) of the filesystem to support hard links?

b. (5 points) How do the semantics of the `Remove` system call change, and how do you implement that change to `Remove` and any other affected system calls?

c. (3 points) What new system calls, if any, must be added to the system? Give the C language-style signature of any new calls, e.g., `int Open(char *file)`. List the signature and give a one sentence definition of each argument and return value.

Problem 3 (cont'd).

This set of questions is about adding **soft** links to the file system. Soft links are also called symbolic links.

d. (4 points) What changes would you make to the internals (directory, fileheader, free map, etc.) of the filesystem to support soft links?

e. (5 points) What existing system calls have to be modified and how to support soft links?

f. (3 points) What new system calls, if any, must be added to the system? Give the C language-style signature of any new calls, e.g., `int Open(char *file)`. List the signature and give a one sentence definition of each argument and return value.

4. (10 points total) Interactions between Operating System Components.

Generally we've talked about each operating system component in isolation. This question asks you to think about ways in which they interoperate. For each pair of systems below, ***in four sentences or less***, give a specific way that they interact (or that they could interact).

Writing that the file system and I/O system interact because they both use the disk is not worth more than a point, and may be worth none. Writing that the file system and I/O system interact when they determine the mapping from logical blocks → physical blocks which impacts the size of file system structures, and the efficiency of the disk usage because larger logical blocks imply more internal fragmentation on the disk is a more complete answer.

a. (6 points) How does a demand paged, lazy-loaded virtual memory system and the file system interact?

b. (4 points) How do the security system (e.g., file permissions) and the virtual memory system?

No Credit – Problem X: (000000000000 points)

The Story of Modern Math

Found on the Internet:

Mathematics in 1950:

Question: A logger sells a truckload of lumber for $100. His cost of production is 4/5 of the price. What is his profit?

Mathematics in 1960:

Question: A logger sells a truckload of lumber for $100. His cost of production is 4/5 of the price, or $80. What is his profit?

Mathematics in 1970:

Question: A logger exchanges a set "L" of lumber for a set "M" of money. The cardinality of set "M" is 100. Each element is worth one dollar. Make 100 dots representing the elements of the set "M." Set "C," the cost of production, contains 20 fewer points than set "M." Represent set "C" as a subset of set "M" and answer the following question: What is the cardinality of the set "P" of profits?

Mathematics in 1980:

Question: A logger sells a truckload of lumber for $100. His cost of production is $80 and his profit is $20. Your assignment: Underline the number 20.

Mathematics in 1990:

Question: By cutting down beautiful forest trees, the logger makes $20. What do you think of this way of making a living? Topic for class participation after answering the question: How did the forest's birds and squirrels feel as the logger cut down the trees? There are no wrong answers.

Mathematics in 2002:

Question: A logger sells a truckload of lumber for $100. His cost of production is $120. How does Arthur Andersen determine that the logger's profit margin is $60?

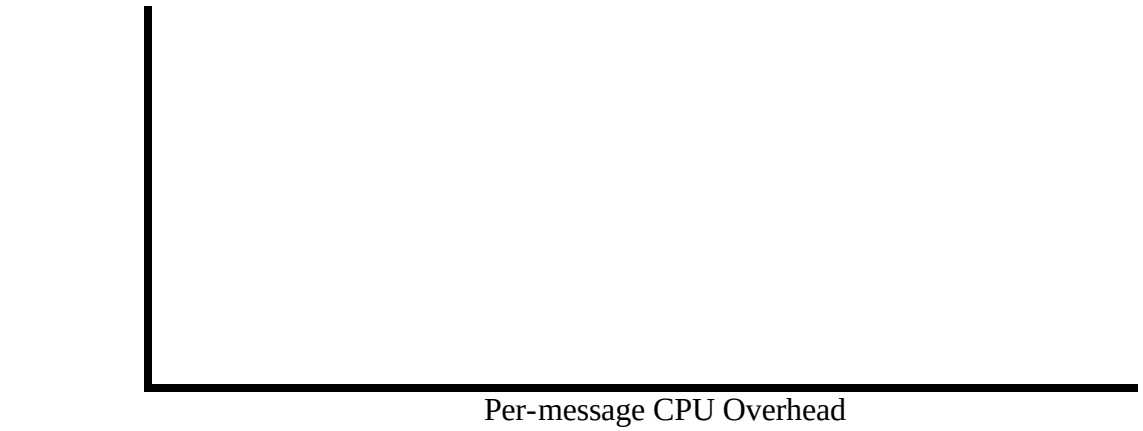
Bonus points: Explain how an Enron accountant can double this profit by selling the lumber to an out-of-state friend, repurchasing it, and selling it to the State of California.

5. (25 points total) Network Performance.

a. (9 points) Consider a TCP network connection with a current window size for unacknowledged bytes of 1,000 bytes, over a cross-country link with a one-way latency of 50 milliseconds, and a link bandwidth of 100 Mbit/second. You may assume that no packets are lost for this particular problem, and that the size of an acknowledgement is essentially 0 bytes long.

How long does it take TCP to transmit 100,000 bytes across the link? That is, how much time elapses from when the first byte is sent by the sender to when the sender *knows* that the receiver has received the last byte?

b. (6 points) Sketch a rough graph ("back of the envelope") illustrating *effective* network bandwidth as a function of per-message CPU overhead. Indicate any important relative values.



Problem 5 (cont'd)

c. (6 points) Assume that the receiver can process incoming data at greater than 100 Mbit/s, what is the optimal window size that the receiver should advertise?

d. (4 points) If the link is shared by N pairs of senders and receivers, does your answer for part (c) change? If so, how? If not, why?