Summer 2003 CS170 - Operating Systems
Midterm 2

August 27, 2003

Instructor: Zoran Dimitrijević (*zoran@cs.ucsb.edu*)
Department of Computer Science
University of California, Santa Barbara

Name:						
CSIL e-mail:						
Question:	1	2	3	4	5	Total
Points:						
Out of:	20	20	20	20	20	100

This is a closed book/notes examination. Write the answers directly on this paper. You are not allowed to use any other scratch paper. **Make your answers as concise as possible.** You have **one hour and 25 minutes** to finish your exam. Please, use the time wisely.

GOOD LUCK!

And remember: **DON'T PANIC!**

1. True/false questions (wrong −2, correct +2, total 20 points)

   1) **True** **False** All processes share the same logical address space.

   2) **True** **False** Multiple threads share both code and data segments.

   3) **True** **False** Multiple threads share the same stack.

   4) **True** **False** Different processes can share physical memory pages.

   5) **True** **False** Limit register is used to limit the number of running processes.

   6) **True** **False** DMA is used to offload the main CPU for large IOs.

   7) **True** **False** Polling is used to offload the main CPU for large IOs.

   8) **True** **False** A victim page is a page selected for swap-in to main memory.

   9) **True** **False** Disk sector size is usually 512 B.

   10) **True** **False** RAID level 0 with 1024 disks is more reliable than a single disk.

2. Virtual memory (total 20 points).

a. (10 points) Explain the steps in handling a page fault. Depict it on a figure with a page table (with valid bits), an OS module for handling page faults (both swap-out and swap-in), a swap file on the disk, and a physical memory.

b. (10 points) Explain the copy-on-write method. If an OS uses paging and copy-on-write, explain what happens during the fork() system call. Explain what is shared between the parent and child process, and what is not.

3. Replacement algorithms (total 20 points).

a. (10 points) Explain LRU algorithm. Depict it on a system with 4 physical memory pages when the processes access the virtual memory pages in the following order: 10, 3, 100, 10, 1, 2, 4, 3, 4, 3, 100, 10, 100, 3.

b. (5 points) Define and explain thrashing.

c. (5 points) A system uses TLB cache, single-level paging, and the virtual memory on the disk. The TLB access requires 10 ns, the memory access requires 100 ns, and the page fault handling requires 10 ms. Probability for a TLB hit is 90%. Probability for a regular TLB miss is 9.9%, and for a page fault miss is 0.1%. Calculate the effective memory access time for this system.

4. File systems (total 20 points)

a. (6 points) Explain memory-mapped files. **Briefly** depict it on a figure containing a process page table, OS page cache, and a file on the disk.

b. (4 points) What is file-system meta-data? Give example.

c. (5 points) Explain how to delete a file if it can have multiple names. Assume that the OS uses a simple counter associated with each file.

d. (5 points) A RAID system has four data disks and a single parity disk. It uses RAID level 5 (distributed parity) organization. Draw a picture to depict the organization (the picture should contain logical blocks $0, 1, 2, ..., 11$ and relevant parity blocks $p_{0,3}$, $p_{4,7}$, and $p_{8,11}$).

5. File system implementation (total 20 points).

a. (8 points) An i-node for a file in indexed allocation contains 10 direct block pointers, two single indirect, and one triple indirect pointer (**no double indirect pointers**). The size of all blocks is 4 kB, and pointers are 4 B each. Explain using a figure how to locate the following data bytes in the file: 128, 5096, 5000000 (approximately).

b. (4 points) What is the maximum file size for this indexed allocation scheme?

c. (4 points) Explain the free space allocation in linked FAT method. Depict it on a figure.

d. (4 points) Explain the external and internal fragmentation. What is the average wasted space due to internal fragmentation in a file system which uses 4 kB blocks and has $100,000$ files.

Relax. These are just few jokes for those of you who finish early and are wise enough to stay a little bit longer and check again for some trivial mistakes. There is NO extra credit for being the fastest one.

Some funny Unix shell messages:

$ drink matter
matter: cannot create

$ ^ What is saccharine?
Bad substitute.

$ "How would you rate your Professors' incompetence?
Unmatched ".

$ rm lawyer-ethics
rm: lawyer-ethics nonexistent

$ ar m God
ar: God does not exist

$ If I had a (for every $ Boesak stole, what would I have?
Too many ('s.

$ make love
Make Don't know how to make love. Stop.

$ got a light?
No match.

$ man Why did you get a divorce?
man Too many arguments.

$ ^ How did the^ sex change operation go?
Modifier failed.

$ make sense
Make Don't know how to make sense. Stop.

$ make mistake
Make Don't know how to make mistake. Stop.

$ date me
You are not superuser date not set Mon Sep 11 155230 PDT 2000