CSE 120: Final

July 27, 2005—Day 10

Name: _____

Student ID:_____

Write your name on *all* pages because the pages will be separated for grading.

No books, no notes, but calculators are allowed. If you need to make an assumption to solve a problem, state the assumption.

Problem	Score
1	/5
2	/5
3	/5
4	/5
5	/5
6	/5
7	/5
8	/5
9	/10
10	/10
11	/10
12	/10
13	/10
14	/10
15	/10
16	/10
17	/5
Total	/125

1. *5 pts.* In your second Nachos project, what class held the page table?

   Solution: UserProcess

2. *5 pts.* In a C program using the Nachos OS, what is the code to spawn another process that executes the "sh.coff" program?

   Solution: int pid = execute("sh.coff", argv, argc);

   What is the code that would wait for it to be completed?

   Solution: int result = join(pid, &status)

3. *5 pts.* Which of the following best describes the difference between global and local page replacement algorithms?

   (a) If a process p page faults, a local page replacement algorithm will page out a page belonging to p, while a global replacement algorithm can page out any page.

   (b) If a process p page faults, a local page replacement algorithm will consider only the recent history of page references (local), while a global page replacement algorithm will consider the entire history of all page references (global).

   (c) Global page replacement algorithms can't exhibit *Belady's anomaly*, while local page replacement algorithms can.

   (d) Global and local page replacement algorithms both try to reduce the frequency of page faults, but global algorithms are easier to implement

   Solution: a) is correct

4. *5 pts.* Which of the following best describes external fragmentation:

   (a) Memory allocators allocate a larger chunk than requested to make it easier for the memory allocation algorithm.

   (b) When paging, allocations have to be an even multiple of the page size, wasting memory

(c) A pattern of memory allocations and deallocations causes free memory to be discontiguous.

Solution: c) is correct. The others are examples of internal fragmentation.

5. *5 pts.* A non-maskable interrupt (NMI):

   (a) Can be generated on some machines by pushing a halt button.
   (b) Causes the hardware to immediately jump to the interrupt service routine assigned to that NMI.
   (c) Can't be disabled by disabling interrupts.
   (d) All of the above
   (e) None of the above

   Solution: d) is correct. They're all examples of NMIs

6. *5 pts.* Why are multilevel page table often used?

   (a) Because they use less memory than a single-level page table.
   (b) The tables that are in-memory can be smaller than a single-level page table.
   (c) Fewer memory accesses are needed to find a page frame than with a single-level page table.
   (d) Multilevel page tables are shared across processes, unlike single-level page tables where one is needed per process.

   Solution: b) is correct. The table in total are bigger, but many of the multi-level page tables can be stored on disk. Only those needed for the pages in the working set need be present in memory.

7. *5 pts.* A working set is best described as:

   (a) The set of pages that that CPU will reference in the near future.
   (b) The set of pages that the CPU has referenced in the recent history
   (c) The set of pages that a process will reference in its near future.
   (d) The set of pages that a process has referenced in its recent history

Solution: d) is correct. The working set is based on looking back in time for a given process

8. *5 pts.* A multicomputer is best described as:

   (a) Loosely-coupled CPUs that don't share memory
   (b) Tightly-coupled CPUs that share memory
   (c) Tightly-coupled CPUs that don't share memory

   Solution: a) is correct.

9. *10 pts.* Consider a simple paging system with the following parameters: 2^{31} bytes of physical memory; page size of 2^{10} bytes; 2^{16} bytes of logical address space.

   (a) How many bits are in a logical address?
      Solution: 16: there are 2^{16} bytes of logical address space, so each one requires 16 bits to address

   (b) How many bytes in a frame?
      Solution: 2^{10}. There are that many bytes in a page, and the size of a page is equal to the size of a frame.

   (c) How many bits in the physical address specify the frame?
      Solution: 21: Since 10 of the 31 bits in a physical address specify the address, the other 21 specify the frame.

   (d) How many entries in the page table?
      Solution: 2^6. In a 16-bit logical address, 10 of the bits specify the offset, leaving 6 for the page. Thus, there are 2^6 pages.

   (e) How many bits in each page table entry (assume each page table entry includes a valid/invalid bit)?
      Solution: 22: Each page table entry contains the frame number (21 bits) plus a valid bit (1 bit).

10. *10 pts.* We can keep track of free disk space on a disk using a bitmap or a free list. If there are 2^D blocks on disk, with F of them free, under what conditions does the free list take fewer bits than the bitmap?

   If D is 16, express your answer as a fraction of the total disk space that must be free.

   Solution: Using a bitmap takes 2^D bits. The free list takes $F * D$. We want $F * D < 2^D$. The fraction of the total disk space that is free is $\frac{F}{2^D}$, so $\frac{F}{2^D} < 1/D$. If D is 16, the fraction must be less than 1/16.

11. *10 pts.* An inverted page table has one entry for each physical page. How is this inverted page table stored and how is it used?

   Solution: The inverted page table is stored as a hash table with key (process ID, virtual page number) and value: physical page number.

On TLB miss, the hash table is searched with key to find matching physical page number. Hash table entries are added (as virtual page numbers are mapped to physical page numbers) and deleted (as virtual pages are evicted).

12. *10 pts.* Unix files can be *sparse*, with holes in them not referencing any allocated disk blocks. Such files can be created by seeking beyond the end of the file and then writing: any intermediate un-written disk blocks are not allocated. If an attempt is made to read them, they appear to be filled with 0's.

    Show how a unix inode might represent such a sparse file.

    Solution: The disk block numbers in the inodes could have a special value representing a non-valid block number (-1, perhaps). When an attempt is made to read from a non-valid block number, 0's are returned.

13. *10 pts.* We have two processes, each of which increments a shared variable

$$\text{shared Integer shared = 0;}$$

    Process A   Process B

    ```
        shared++;                        shared++;
    ```

    Give the possible values for the shared variable when the processes finish, along with a detailed analysis of how each value can occur.

    Solution: The shared variable can be 2 if A runs, then B runs. The shared variable can be 1 if A does a read, then increment, then B does a read, then increment, then A writes, then B writes.

14. *10 pts.* In a public key encryption system, when would you encrypt something with a private key rather than a public key?

    Solution: You'd encrypt with a private key when you are trying to sign a document. Add a hash and encrypt the hash with the private key. Anyone can decrypt the encrypted hash with the corresponding public key and then verify the hash.

15. *10 pts.* What atomic (hardware) instruction can be used to implement a spin-lock? Show how you'd use that instruction to protect a critical section.

    Solution: You can use a test-and-set instruction. Here's an example:
    while (TestAndSet(&var) == 1) {
    }

    critical section

    var = 0

16. *10 pts.* Define the following terms: *trojan horse*, *worm*, and *virus*. Make sure that you are clear in how they differ from one another.

    Solution:

    trojan horse Program that contains code that an adversary wishes to run using your permissions.

    virus Code that infects other programs.

    worm Code that spreads itself across a network.

17. *5 pts.*

    (a) What are three things you liked most about this class?

    (b) What are three things you liked least about this class?

    (c) What would you like to see changed?