July 24, 1997

V1

cs450 - Operating Systems 1
Summer 1997

Final Exam
closed books, closed notes

Starts: **6:20 pm** Ends**: 8:20 pm**

Name:______________________________________(please print)

ID:__________________________

Problem	Max points	Your mark	Comments
1	15		5 + 5 + 5
2	10		5 + 5
3	15		5 + 10
4	15		5 + 5 + 5
5	15		10 +5
6	10		
7	10		
8	10		
	100		

What happens with this paper (mark one):**Discard** **Mail** at the following address

1. A system with a 32 bit virtual address has a physical memory of 32 MB and uses demand paging with pages of 2 KB.

a) Show the size of various fields in the virtual address.

b) What is the size of the page table? You may assume that each entry in the table has a size which is a multiple of byte; each entry uses four bits for various purposes (the ‘Valid’ bit, ‘Dirty’, etc.).

c) Given the following list of references (addresses issued by the CPU), indicate which will generate page-faults: initially only the page which contains the reference **`0x400000`** is in the memory.

Reference	Page fault?	
`0x400020`		
`0x400024`		
`0x7fffffe0`		
`0x400028`		
`0x401000`		
`0x10000004`		

Reference	Page fault?	
`0x40002c`		
`0x401800`		

2. A simple UNIX file system has a block size of 512 bytes, and each block number is 4 bytes. The first 10 entries (from 0 to 9) in the i-node are direct pointers to disk blocks. Entry number 11 is a pointer to a block ('single indirect block') of 7 pointers to disk blocks. Entry number 12 points to a 'double indirect block' which contains 7 pointers to blocks of pointers (seven in each) to disk blocks. Finally, entry number 12 in the i-node is a pointer to a 'triple indirect block'.

a) What is the maximum file size (in bytes or multiple of byte)?

Suppose that the i-node structure for the file *myfile* contains the following block list (numbers are in decimal and the leftmost number is entry 0):

`50 51 52 53 77 78 79 80 99 31 32 96 81`

and the contents of some of these blocks are given in the following table:

Block #	Contents
31	**27 28 29 ...**
32	**65 97 24 ...**
80	**44 45 46 ...**
81	**89 92 91 ...**
89	**94 95 88 ...**
90	**41 42 43 ...**
91	**57 58 59 ...**
92	**60 61 62 ...**

Block #	Contents
93	**67 68 69 ...**
94	**36 37 39 ...**
95	**47 48 49 ...**
96	**93 98 90 ...**
97	**83 84 85 ...**
98	**72 73 74 ...**

Which disk blocks must be read for the following operations? Answer both in logical blocks and in physical blocks.

b) Offset 2000 in *myfile*: read 1024 bytes:

3. Fetch And Add (FAA), *atomically* increments a variable and returns the old value of the variable. `FAA(var, increment)` adds `increment` to the variable `var` and returns the initial value of `var`. We plan to use this for mutual exclusion:

```
shared int S;

// each process executes
while(TRUE){
      enter_section();
      ......                    // here is the critical section
      exit_section();
      ......                    // this is the remainder section
}

// enter_section() returns when it is safe to enter the critical
// section
void enter_section()
{
      while( FAA(S,-1) < 1 ) {
            FAA(S,1);
      }
}
```

a) What should be the initial value of S for this code to work?

S =

b) Write the code for **`exit_section()`**.

4. Two processes, P1 and P2, execute at the same time the following code:

```
for (i=0, i<2; i++)
      S = S+1;
```

where S is a shared integer with the initial value of 0.

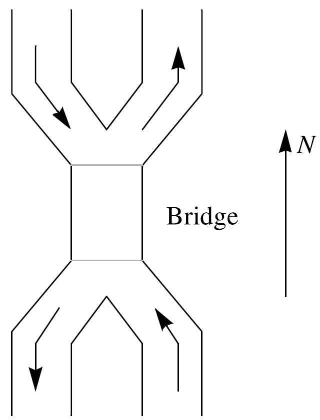
a) What would be the value of S if the two processes would execute to completion in the order P1, P2 (this means that P1 starts and runs to completion, then P2 executes)?

S =

b) What are possible values for S in reality? Hint: write the assembly code that implements the **`for`** loop.

c) Use semaphore(s) to fix the problem: in other words, modify the code in such a way as to always obtain the result described in part a.

5. At same place, the North-South Highway goes over a river. The bridge is in construction and the traffic on the bridge is on a single lane. At most MAX cars can be on the bridge at any time, all going the same direction (either North or South, but never in both directions at the same time).



a) Devise a solution with semaphores for this problem.

b) Is your solution starvation-free? Explain.

6. List the necessary conditions for deadlock

7. Explain the difference between external and internal fragmentation.

8. Assume a demand-paged memory. The page table is held in registers. It takes 8 ms to service a page fault if an empty page is available or the replaced page is not modified, and 20 ms if the replaced page is modified. The memory access time is 100 ns. You may also assume that the page to be replaced is modified in 70% of the time. What is the maximum acceptable access time for an effective access time of no more than 200 ns.