CSci 402 - Operating Systems

Final Exam (DEN Section)

Fall 2023

[9:00:00am-9:40:00am), Monday, December 11)

Instructor: Bill Cheng

Teaching Assistant: Zhuojin Li

(This exam is open book and open notes.
Remember what you have promised when you signed your
Academic Integrity Honor Code Pledge.)

Time: 40 minutes

_____-
Name (please print)

Total: 38 points

_____-
Signature

Instructions

1. This is the first page of your exam. The previous page is a title page and does not have a page number. Since this is a take-home exam, no need to sign above since you won't submit this file.
2. Read problem descriptions carefully. You may not receive any credit if you answer the wrong question. Furthermore, if a problem says *"in N words or less"*, use that as a hint that N words or less are expected in the answer (your answer can be longer if you want). Please note that points may get *deducted* if you put in wrong stuff in your answer.
3. If a question doesn't say `weenix`, please do not give `weenix`-specific answers.
4. Write answers to all problems in the **answers text file**.
5. For non-multiple-choice and non-fill-in-the blank questions, please show all work (if applicable and appropriate). If you cannot finish a problem, your written work may help us to give you partial credit. We may not give full credit for answers only (i.e., for answers that do not show any work). Grading can only be based on what you wrote and cannot be based on what's on your mind when you wrote your answers.
6. Please do *not* just draw pictures to answer questions (unless you are specifically asked to draw pictures). Pictures will not be considered for grading unless they are clearly explained with words, equations, and/or formulas. It's very difficult to draw pictures in a text file and you are not permitted to submit additional files other than the answers text file.
7. For problems that have multiple parts, please clearly *label* which part you are providing answers for.
8. Please ignore minor spelling and grammatical errors. They do not make an answer invalid or incorrect.
9. During the exam, please only ask questions to *clarify* problems. Questions such as "would it be okay if I answer it this way" will not be answered (unless it can be answered to the whole class). Also, you are suppose to know the definitions and abbreviations/acronyms of *all technical terms*. We cannot "clarify" them for you. We also will **not** answer any clarification-type question for multiple choice problems since that would often give answers away.
10. Unless otherwise specified and stated explicitly, multiple choice questions have one or more correct answers. You will get points for selecting correct ones and you will lose points for selecting wrong ones.
11. When we grade your exam, we must assume that you wrote what you meant and you meant what you wrote. So, please write your answers accordingly.

(Q1) (2 points) What are the **OS design approaches** to fix the **reliability** problem of a **monolithic kernel** to reduce kernel crashes?

(1) use dynamically loaded kernel modules so that unreliable kernel module can be unloaded while the rest of the kernel is running
(2) reduce the size of the kernel code that must run in the privileged mode
(3) run transactions in the file system to improve kernel crash resiliency
(4) run the entire kernel in user mode
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q2) (2 points) Which of the following are approaches used to **reduce page fault latency**?

(1) use a pageout daemon
(2) use a FIFO page replacement policy
(3) lazy evaluation
(4) increase the size of translation lookaside buffer
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q3) (2 points) Which of the following statements are correct about **paravirtualization**?

(1) a paravirtualized OS is indistinguishable from the real OS in the sense that it can also run on the hardware the real OS was designed to run on
(2) one way to implement paravirtualization is to modify the hardware so that an OS can run inside a virtual machine without modification
(3) inside a commercial paravirtualized OS, usually there are no device drivers and there are no file systems in the guest/virtualized OS
(4) VMware is well-known for its paravirtualization patent
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q4) (3 points) Let's say that you have four threads A, B, C, and D and you are using the basic **round-robin (RR) / time-slicing** scheduler with a very small time slice. At time zero, all four threads are in the run queue and their processing times are shown in the table below. Assuming that there are no future arrivals into the run queue, please complete the table below with the "waiting time" of all four threads and the "average waiting time" (AWT) of these four threads and write the results on your answer sheet. Please make it very clear which waiting time is for which thread and which one is the AWT. For non-integer answers, you can use fractions or decimals with two digits after the decimal point. Your answer must not contain plus or multiplication symbols. You must use the definition of "waiting time" given in lectures.

	A	B	C	D	AWT (1 pt)
T (hrs)	15	14	13	11	-
wt (hrs)					

(Q5) (2 points) Which of the following statements are correct about **directed processing**?

(1) the "asynchronous procedure call" mechanism in Windows is a form of "directed processing"
(2) in "directed processing" in Unix, a kernel thread is used to execute code for a user process without going into the user space
(3) the "signals" mechanism in Unix is more general than the Windows APC mechanism because it uses an upcall
(4) in "directed processing" in Unix, a user thread is used to execute code inside the kernel
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q6) (2 points) What did **FFS** do to **improve the effective disk access time** of **S5FS**?

(1) reduce block size
(2) use bit-interleaving to improve seek time
(3) use cylinder group to improve rotational latency
(4) use RAID
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q7) (2 points) Which of the following statements are correct about a **B+ tree of order** $m = 7$?

(1) since m is 7, the height of the B+ tree must be strictly greater than 4
(2) since m is 7, it's okay for an intermediate node (i.e., neither a root node nor a leaf node) to have 4 child nodes
(3) since m is 7, it's okay for the root node to have 7 child nodes
(4) since m is 7, it's okay for an intermediate node (i.e., neither a root node nor a leaf node) to have 3 child nodes
(5) since m is 7, the height of the B+ tree must be strictly less than 7

Answer (just give numbers): _____

(Q8) (3 points) Let's say that you have four threads A, B, C, and D and you are using **stride scheduling**. You have decided to give thread A 8 ticket, thread B 15 tickets, thread C 16 tickets, and thread D 18 tickets. The initial pass values that **you must used** for the four threads are shown below along with the "winner" of the iteration 1. Please run **stride scheduling** to fill out all the entries (pass values) in the table and keep track of the "winner" in each round. For **iterations 2 through 7**, please write on your answer sheet the "winner" and the winning pass value of that iteration. (For example, you would write "C:7" for iteration 1 since C is the "winner" of iteration 1 and the winning pass value is 7.) You must use the **smallest possible integer stride values** when calculating all the pass values. If you get the stride values wrong, you will not get any partial credit for this problem.

itr	A	B	C	D
1	17	12	(7)	21
2				
3				
4				
5				
6				
7				

(Q9) (2 points) Let's say that the address space of a user space process in **weenix** looks like the following:

```
    VADDR RANGE       PROT FLAGS    MMOBJ    OFFSET    VFN RANGE
0x08029000-0x08038000 rw-  PRIVATE 0xcfe0c034 0x0001a 0x08029-0x08038
0x0803d000-0x0804c000 r-x  PRIVATE 0xcfe0c004 0x0001c 0x0803d-0x0804c
0x0804d000-0x08060000 rw-  PRIVATE 0xcfe0c064 0x0003c 0x0804d-0x08060
```

If you get a page fault with vaddr = 0x0804b984, what **pagenum** would you use to lookup a page frame when you are handling a page fault? Please just give an integer value answer (no partial credit for this problem).

(Q10) (2 points) Which of the following statements are correct about the **scheduler activations model**?

(1) scheduler activations model is not popular because it's undesirable to let user-space schedulers to make scheduling decisions
(2) the down side of the scheduler activations model is that it's difficult to make priority-based scheduling work well
(3) the down side of the scheduler activations model is that if a user thread makes a system call, another user thread in in the user process cannot make a system call until the first thread has returned from the kernel
(4) in the scheduler activations model, the kernel does not schedule/assign CPUs to processes; instead, the kernel schedules/assigns CPUs to threads
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q11) (2 points) A **disk map** of a S5FS **inode** contains 13 disk block pointers. If a disk block is 1KB in size, which of the following are **possible file sizes** if an inode's **disk map** contains non-null block numbers in the first 12 entries and the last entry is null?

(1) 128 MB
(2) 960 MB
(3) 96 MB
(4) 32 MB
(5) 98 KB

Answer (just give numbers): _____

(Q12) (2 points) Which of the following statements are correct about the **popf machine instruction** in a **traditional Intel x86 processor** (i.e., non-Vanderpool)?

(1) executing `popf` in the user mode will not cause a trap while executing it in the privileged mode will cause a trap
(2) executing `popf` will not cause a trap in either user mode or privileged mode
(3) executing `popf` in the user mode causes a trap while executing it in the privileged mode does not
(4) the `popf` machine instruction would behave differently when it is executed in different CPU modes (i.e., user mode vs. privilege mode)
(5) `popf` is a privileged instruction and not a "sensitive instruction"

Answer (just give numbers): _____

(Q13) (2 points) Which of the following statements are correct about the naive spin lock implementation vs. the **"better" spin lock implementation**?

(1) both the naive spin lock and the "better" spin lock do busy-waiting
(2) the naive spin lock does busy-waiting while the "better" spin lock does not do busy-waiting
(3) no matter which spin lock implementations you use, the impact on system performance is the same if the spin lock is currently unavailable
(4) using the "better" spin lock can lock the spin lock faster than the naive spin lock if the spin lock is currently available
(5) none of the above is a correct answer

Answer (just give numbers): _____

(Q14) (2 points) Which of the following are maintained in a **S5FS superblock**?

(1) first node of the free disk block list
(2) root node of the B+ tree for locating all the inodes
(3) inode numbers of all the free inodes
(4) inode cache
(5) disk map

Answer (just give numbers): _____

(Q15) (2 points) Which of the following statements are correct about **LFS (log-structured file system**?

(1) the two checkpoint files in LFS achieves the same functionality as the superblock in S5FS
(2) the inode map in LFS achieves the same functionality as the disk map in S5FS
(3) LFS's append-only and never delete/modify requirements make the idea of LFS not very useful in practice for a file system on a hard drive
(4) LFS's append-only and never delete/modify requirements make the idea of LFS not very useful in practice for any type of file systems
(5) LFS is designed to use close to 100% of the disk transfer capacity when writing to the disk

Answer (just give numbers): _____

(Q16) (2 points) A correct mutex implementation for **straight-threads** (i.e., no interrupt) **synchronization** on a single CPU is shown here:

```
void mutex_lock(mutex_t *m)
{
  if (m->locked) {
    enqueue(m->queue,
        CurrentThread);
    thread_switch();
  } else
    m->locked = 1;
}
```

```
void mutex_unlock(mutex_t *m)
{
  if (queue_empty(m->queue))
    m->locked = 0;
  else
    enqueue(runqueue,
        dequeue(m->queue));
}
```

Let's say that thread X owns the mutex **m** (i.e., has it locked). If thread X calls **mutex_unlock()** and the mutex queue is **not** empty, the thread at the head of the mutex queue (let's call it thread Y) is supposed to own the mutex next. The above code would dequeue thread Y from the mutex queue and enqueue thread Y to the run queue **without unlocking the mutex**. Referring to the above code, which of the following statements are correct about **the next time thread Y will run** in the CPU?

(1) thread Y will return from the **thread_switch()** function inside **mutex_lock()** but thread X is still the owner of mutex **m**
(2) even though thread Y is the new mutex owner, thread Y will still need to call **mutex_lock()** and may go to sleep again in **thread_switch()**
(3) thread Y will return from the **thread_switch()** function inside **mutex_lock()** as the owner of mutex **m**
(4) since thread Y is the new mutex owner, thread Y will call **mutex_lock()** again and is guaranteed to be the new owner of mutex **m**
(5) thread Y may wake up inside **thread_switch()** but will go to sleep again insdie **thread_switch()** without returning from **thread_switch()**

Answer (just give numbers): _____

(Q17) (2 points) Which of the following statements are correct about virtual machine (VM) and virtual machine monitor (VMM)?

(1) VMM is a user space program that runs inside a virtual machine
(2) VMM is also known as "hypervisor"
(3) when virtual machine is used, the OS of the real machine is often referred to as the guest OS
(4) when virtual machine is used, the guest OS runs in the user mode of the real machine
(5) a "virtual machine mointor" is a terminal device that's used to interact with a virtual machine

Answer (just give numbers): _____

(Q18) (2 points) Let's say that you are using a **rate-monotonic scheduler** to schedule 4 periodic tasks with $T_1 = 0.5$, $P_1 = 5$, $T_2 = 1$, $P_2 = 6.5$, $T_3 = 0.5$, $P_3 = 7.5$, and $T_4 = 1$, $P_4 = 8$ (all in seconds). If you were to simulate the **rate-monotonic scheduler** to see if it can schedule these 4 periodic tasks without any of them missing a deadline, assuming that you schedule all 4 periodic tasks to start a time = 0, how many seconds into the simulation would be the first time all 4 jobs would be scheduled to start executing at exactly the same time again? Please just give a numeric answer (no partial credit for this problem).