CS 162 Operating Systems and System Programming

Fall 2024

Midterm 3

INSTRUCTIONS

Please do not open this exam until instructed to do so. Do not discuss exam questions for at least 24 hours after the exam ends, as some students may be taking the exam at a different time.

For questions with **circular bubbles**, you should select exactly *one* choice.

◯ You must choose either this option

◯ Or this one, but not both!

For questions with **square checkboxes**, you may select *multiple* choices.

☐ You could select this choice.

☐ You could select this one too!

GENERAL INFORMATION

This is a **closed book** exam. You are allowed 1 page of notes (both sides). You have 110 minutes to complete as much of the exam as possible. Make sure to read all of the questions first, 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.* On programming questions, we will be looking for performance as well as correctness, so think through your answers carefully. If there is something about the questions that you believe is open to interpretation, please ask us about it!

Problem	**Possible**
1	16
2	18
3	22
4	11
5	11
6	13
Total	90

Preliminaries

This is a proctored, closed-book exam. You are allowed 1 page of notes (both sides). You may not use a calculator. You have 110 minutes to complete as much of the exam as possible. This exam is out of 100 points. Make sure to read all the questions first, as some are substantially more time-consuming.

If there is something about the questions you believe is open to interpretation, please ask us about it.

We will overlook minor syntax errors when grading coding questions. **There is a reference sheet at the end of the exam that you may find helpful.**

(a)

Name

(b)

Student ID

(c)

Discussion TA's Full Name

(d)

Please read the following honor code: "I understand this is a closed-book exam. I promise the answers I give on this exam are my own. I understand that I am allowed to use one 8.5x11, double-sided, handwritten cheat-sheets of my own making, but otherwise promise not to consult other people, physical resources (e.g. textbooks), or the internet in constructing my answers."

Write your full name below to acknowledge that you've read and agreed to this statement.

The Servers have hit a Utilization of 100%, becoming unresponsive to our request for this page.

1. **(10.0 points) True/False**

Please explain your answer in **TWO SENTENCES OR FEWER**.
Select **only one of True or False** and provide an appropriate explanation for the answer of your choice. You must fill out **at most one** box for each question. **If you believe a choice to be INCORRECT, you DO NOT NEED TO FILL OUT THE BOX CORRESPONDING TO THE INCORRECT CHOICE**.

(a) **(2.0 pt)** Response time will grow unboundedly as utilization approaches 1, regardless of the system's distribution of job arrival and servicing times.

○ True. Explain why every system with utilization 1 will have an unbounded response time.

○ False. Explain why some systems with utilization 1 will have a bounded response time.

(b) **(2.0 pt)** The Byzantine General's Problem can be solved as long as only a minority of players (i.e. lieutenants and generals) are malicious.

○ True. Explain why this is true.

○ False. Make a correct claim on what the maximum number of malicious players can be.

(c) (2.0 pt) Disk controllers allow you to write an individual byte at a time to disk.

◯ True. Explain how disk controllers support writing in byte-sized granularities.

◯ False. Explain why writing individual bytes to disk is not supported.

(d) (2.0 pt) Multiple file path names can refer to the same underlying inode.

◯ True. Explain what mechanism allows this.

◯ False. Explain why this would be impossible on a typical operating system.

(e) (2.0 pt) In a journaling file system, appending a `COMMIT` log to the transaction indicates that all changes to the transaction have been reflected on disk.

◯ True. Explain problems that may arise if this were not the case in case of a crash.

◯ False. Explain why this does not result in loss of data in case of a crash.

(f) (2.0 pt) PintOS allows processes to invoke the `read` syscall on a directory.

◯ True. Explain why reading the raw bytes of a directory is allowed.

◯ False. Explain why reading the raw bytes of a directory is not allowed.

(g) (2.0 pt) In NTFS, reading very small files requires a minimum of two disk accesses when neither the Master File Table (MFT) nor the file is cached: one access for the MFT, and one for the file data.

◯ True. Explain why these 2 disk accesses are unavoidable.

◯ False. Explain how you can access a file with fewer than 2 disk accesses.

(h) (2.0 pt) All transformations on Spark immediately materialize their output on disk for fault tolerance.

◯ True. Explain why not doing so will lead to irrecoverable loss of data.

◯ False. Explain what Spark actually does on transformations and mention what mechanism provides fault tolerance despite not immediately materializing results.

(i) (2.0 pt)

○ True. From Homework 4, state what dynamic memory allocation can lead to.

○ False. Explain what Spark actually does on transformations and mention what mechanism provides fault tolerance despite not immediately materializing results.

2. (18.0 points) Multiple Select

Follow the directions for each question. If a question does not have a choice for "None of the above", it means that at least one choice is correct. **There is no partial credit given to leaving the entire question blank.**

(a) (2.0 pt) Select all true advantages of an HDD over an SSD.

- ☐ It is typically cheaper.
- ☐ It is more robust and resistant against physical damage.
- ☐ It can sustain a larger number of writes.
- ☐ It is typically more performant, offering faster reads and writes.
- ☐ None of the above.

(b) (2.5 pt) Select all true statements about the FTL (Flash Translation Layer).

- ☐ SSDs use it to map logical blocks to physical blocks.
- ☐ It helps prevent wear out of a flash chip by distributing erase and write operations across all blocks.
- ☐ It buffers small writes and aggregates them into one large write.
- ☐ The FTL allows low-level flash blocks to be overwritten directly without having to erase it first.
- ☐ The FTL can relocate data on flash without the OS knowing.

(c) (2.5 pt) Select all true statements about file systems.

- ☐ A directory can contain files and other directories as its entries.
- ☐ Most files in a file system tend to be small.
- ☐ Most of the storage space in a file system is used for large files.
- ☐ In inode-based file systems, every entry in the file descriptor table corresponds to a unique inode.
- ☐ A file system translates a path name into a "file number".

(d) (2.5 pt) Select all true statements regarding I/O devices.

- ☐ Device drivers have a set of registers that can be written and read to interact with the device.
- ☐ Memory mapped I/O maps each device's control registers to a range of physical addresses on the memory bus.
- ☐ Both memory mapped and port mapped I/O use the same physical address space as the system's main memory.
- ☐ Port-mapped I/O uses `in` and `out` instructions to interface with periphery devices.
- ☐ Memory-mapped I/O uses `in` and `out` instructions to interface with periphery devices.

(e) (2.0 pt) Select all true statements about 4.2 BSD FFS.

- ☐ It implements skip sectors, which interleaves other data in between sequential blocks of a file.
- ☐ It reserves 10% of disk space for efficiency.
- ☐ It places all inodes on the same track for read efficiency.
- ☐ It uses a bitmap over a free-list.
- ☐ None of the above.

(f) **(2.5 pt)** Select all true statements about FFS and FAT.

- ☐ FAT can support hard links.
- ☐ Each entry in the FAT corresponds to either a free block, the end of a file, or the block number of the next data block in the file.
- ☐ FFS is better than FAT for portable storage media such as flash drives.
- ☐ Reading sequentially from a large file is typically more efficient in FAT than in FFS.
- ☐ Entries in the FAT table contain no metadata (size, name, file type etc.) pertaining to a file.

(g) **(2.0 pt)** Select all true statements about the buffer cache in Project 2.

- ☐ The buffer cache can merge multiple writes to a single sector into one disk write.
- ☐ You can cache the inode array, which would lead to a boost in performance.
- ☐ Small temporary files may be created and deleted without having any of its data make it to the disk.
- ☐ The buffer cache is a write-through cache.
- ☐ None of the above.

(h) **(2.0 pt)** Select all true statements about worker crashes in Homework 5.

- ☐ If the crashed worker completed a reduce task, you have to reassign this task to another worker.
- ☐ If a worker crashes while working on a job, the entire job should fail without reassignment.
- ☐ Crashed workers will invoke the FailTask RPC in the coordinator.
- ☐ The coordinator guesses that a worker crashed if it hasn't been contacted by the worker for a given timeout duration.
- ☐ None of the above.

3. (21.0 points) Short Answer

(a) **(3.0 pt)** A graphics card is a periphery device that receives large buffers of data from the processor at a very high frequency. What mechanism would allow efficient transmitting data to a graphics card with minimal involvement of the processor? Explain in detail how the processor can initiate this mechanism.

(b) **(6.0 pt)** Compare RAID 1 and RAID 5 setups regarding failure tolerance, write performance, and storage efficiency. **Which RAID setup is better and why?** For each subpart, explain your answer in 1-2 sentences. Assume individual disks are equal to one another in terms of performance. There is no partial credit for a correct selection but an incorrect explanation.

i. **(2.0 pt)** Failure tolerance (i.e. how many disk failures they tolerate):

○ RAID 1 ○ RAID 5 ○ Same

ii. **(2.0 pt)** Write performance:

○ RAID 1 ○ RAID 5 ○ Same

iii. **(2.0 pt)** Storage efficiency:

○ RAID 1 ○ RAID 5 ○ Same

(c) **(8.0 pt)** Page Replacement

For the following problem, assume we have a machine with 4 pages of physical memory and 7 pages of virtual memory.
We are given the following access pattern: **A B C D E F A B E G F A E D A F G**

Complete the following table to mark which pages are mapped to which physical pages for each of the following page replacement examples. Assume that each blank box matches the page to its left.

Access →		A	B	C	D	E	F	A	B	E	G	F	A	E	D	A	F	G
FIFO	1	A				E				+	G						F	
	2		B				F					+		E				G
	3			C				A					+		D			
	4				D				B							A		
MIN	1																	
	2																	
	3																	
	4																	
LRU	1																	
	2																	
	3																	
	4																	

(d) **(4.0 pt)** Answer the following questions about transactions and journaling file systems.

i. **(2.0 pt)** Define what a transaction is, and explain why it is important for it to be "atomic."

ii. **(2.0 pt)** We learned in lecture that journaling file systems can log metadata updates, rather than all updates to the file system. What is the utility of such a system, even though we cannot roll back and recover corrupted data?

4. (12.0 points) A Tale of 2 PCs ... and a Mac

Consider the following scenario in the 2PC protocol:

```
1. Jacob writes PREPARE to his log.
2. Jacob sends Diana and Ashwin VOTE-REQ to play League.
3. Jacob sends Diana VOTE-REQ to play League
4. Ashwin writes VOTE-COMMIT to his log.
5. Diana writes VOTE-COMMIT to her log.
6. Ashwin sends Jacob VOTE-COMMIT in favor of playing League.
7. Diana sends Jacob VOTE-COMMIT in favor of playing League.
8. Jacob writes GLOBAL-COMMIT to his log.
9. Jacob sends Ashwin GLOBAL-COMMIT to play League.
10. Ashwin writes GLOBAL-COMMIT to his log.
11. Ashwin sends Jacob ACK.
12. Jacob sends Diana GLOBAL-COMMIT to play League.
13. Diana writes GLOBAL-COMMIT to her log.
14. Diana sends Jacob ACK.
15. Diana, Ashwin, and Jacob start playing League.
```

Only assume the situations specified in each question (i.e. situations don't carry over into the next question). Assume that all transmissions over the network are instantaneous and guaranteed to succeed. Answer each question in three sentences or fewer. Any answer without explanation may not receive credit.

(a) **(2.0 pt)** Suppose that Diana crashes **immediately after step 5** for a few seconds (i.e. *after* logging `VOTE-COMMIT`). She wakes up before Jacob times out. Assume no further crashes.

Are the three guaranteed to still play League?

○ Yes. ○ No.

Explain, and specify what action Diana takes upon waking up.

(b) **(2.0 pt)** Suppose that Jacob crashes **immediately *before* step 8** for a few seconds (i.e. *before* logging `GLOBAL-COMMIT`). Are the three guaranteed to still play League?

○ Yes. ○ No.

Explain, and specify what action Jacob takes upon waking up.

(c) **(2.0 pt)** Suppose that Jacob crashes **immediately *after* step 8** for a few seconds (i.e. *after* logging `GLOBAL-COMMIT`). Assume all nodes eventually recover after a crash.

Are the three guaranteed to still play League?

◯ Yes. ◯ No.

Explain, and specify what action Jacob takes upon waking up.

(d) **(2.0 pt)** Suppose that Ashwin crashes **immediately before step 11**.

Are the three guaranteed to still play League?

◯ Yes. ◯ No.

Explain, and specify what action Ashwin takes upon waking up.

(e) **(3.0 pt)** Suppose in the above problem that Jacob instead sends a vote to meet at Soda Hall tomorrow at 6 pm. Diana and Ashwin send `VOTE-COMMIT`, Jacob makes a decision to `GLOBAL-COMMIT`, and Jacob receives ACKs from both Diana and Ashwin before 6pm.

Recall that the General's Paradox states that coordinating simultaneous action is impossible with an unreliable communication channel, no matter how many acknowledgements you send. Does 2PC solve the General's Paradox in this case? Explain your answer. (Hint: It has nothing to do with recovery.)

5. (11.0 points) Wait, it's all files?

Sriram finally did his job and updated the PintOS filesystem to support 64-bit disk and block pointers and 4KiB blocks. Additionally, with Sriram's implementation of Project 3, each inode contains 8 direct pointers, 1 indirect pointer, 1 doubly-indirect pointer, and 1 triply-indirect pointer.

```
struct inode_disk {
    uint64_t direct[8];
    uint64_t indirect;
    uint64_t doubly_indirect;
    uint64_t triply_indirect;
}
```

(a) **(1.0 pt)** What is the maximum disk size that this file system can support, in bytes?

You may leave your answer unsimplified (i.e. sum/product of powers of 2).

(Hint: Recall that disk pointers are now 64 bits.)

(b) **(2.0 pt)** How many data blocks does a doubly-indirect pointer contain?

(c) **(2.0 pt)** What is the maximum file size this filesystem can support?

You may leave your answer unsimplified (i.e. sum/product of powers of 2).

Suppose we want to read `/a/b/c.txt` in this modified PintOS file system. Assume that all directory entries fit into exactly one disk block.

(d) **(3.0 pt)** How many disk accesses are incurred to find the file number (inumber) for c.txt (`/a/b/c.txt`)?

Show your work below.

(e) **(3.0 pt)** Suppose you have now determined the file number (inumber) for c.txt (`/a/b/c.txt`).

How many disk accesses would it take, given `c.txt`'s file number, to read in 165 KiB of data from `c.txt`?

Show your work below. Specify how many disk accesses incur from direct pointers, the indirect pointer, etc.

6. (13.0 points) Should've bought an SSD

Suppose we have a hard drive with the following specifications:

Controller delay: 1 ms
Seek time: 6 ms
Rotation speed: 3,000 RPM
Disk head transfer rate: 4000 KiB / s
Sector size: 2 KiB
Effective transfer rate: (size of data) / (total service time)

(a) **(3.0 pt)** What is the average latency of reading a single sector from disk at random?

(b) **(4.0 pt)** How many sequential sectors would we have to combine together into a block in order to achieve an effective transfer rate of at least 25% (so 1/4th) of the disk transfer rate when reading a block of data?

Suppose you ran analytics on disk usage with `iostat` and saw that the average length of the disk I/O queue was 5 requests. For parts (c) and (d), assume that the average service time of all disk requests is 50 ms. Also, the average amount of time spent in the queue is 200 ms (between being queued and being sent to the disk controller).

(c) **(3.0 pt)** What is the average arrival rate of disk requests? Answer with appropriate units.

(d) **(3.0 pt)** Suppose instead that the average arrival rate of disk requests is 10 requests per second. What is the utilization of the disk?

Goodbye CS 162!