CSC 369H1 F 2019 Midterm Test
Duration — 50 minutes
Aids allowed: none

Student Number: |_|_|_|_|_|_|_|_|_|_|_|_|_|

Last Name: _____ **First Name:** _____

Lecture Section: L5101 **Instructor: Reid**

---

*Do **not** turn this page until you have received the signal to start.*
(Please fill out the identification section above, **write your name on the back of the test**, and read the instructions below.)
Good Luck!

This midterm consists of 5 questions on 8 pages (including this one). *When you receive the signal to start, please make sure that your copy is complete.* Pseudo-code is acceptable where code is required. Answer the questions concisely and legibly. Answers that include both correct and incorrect or irrelevant statements will not receive full marks.
If you use any space for rough work, indicate clearly what you want marked.

# 1:	_____/ 6
# 2:	_____/ 7
# 3:	_____/ 2
# 4:	_____/ 6
# 5:	_____/ 9
TOTAL:	_____/30

Question 1. [6 MARKS]

Each of the following statements is false. In one sentence, explain why. (If you disagree that the statement is false, provide your reasoning.)

Part (a) [1 MARK]

A system call is invoked in the same manner as a user function.

Part (b) [1 MARK]

If two threads access a shared variable at the same time, there will be a concurrency error.

Part (c) [1 MARK]

Multiple threads in the same process share all code and data resources.

Part (d) [1 MARK]

A context switch from one process to another can be accomplished without executing OS code in kernel mode.

Part (e) [1 MARK]

External fragmentation occurs when a file is broken up into many pieces on a disk.

Part (f) [1 MARK]

It is important to store blocks of a file close together on a Solid State Drive to speed up sequential access.

Question 2. [7 MARKS]

Part (a) [1 MARK]

Briefly explain what causes a thread to move from the ready state to running state.

Part (b) [1 MARK]

Briefly explain what causes a thread to move from the running state to the blocked state.

Part (c) [1 MARK]

Briefly explain what causes a thread to move from the blocked state to the ready state.

Part (d) [2 MARKS]

Identify and briefly explain two types of operations that result in a mode switch from user to kernel mode.

Part (e) [2 MARKS]

Give one concrete example of how the operating system virtualizes the hardware resources that we have seen so far in the course. Identify which resource is virtualized and how the operating system achieves this virtualization.

Question 3. [2 MARKS]

Consider the following excerpt from the solution to the reader/writer problem using semaphores:

```
// assume all variables are correctly initialized

sem_wait(mutex);
readcount += 1;

if(readcount == 1)
    sem_wait(w_or_r);

sem_signal(mutex);
```

Which of the following statements are true?

- ☐ `readcount` must be 1 when `sem_wait(w_or_r)` returns.
- ☐ `readcount` cannot be 0 when `sem_wait(w_or_r)` returns.
- ☐ If thread A is blocked in `sem_wait(w_or_r)` other threads will block on `sem_wait(mutex)` because thread A "holds" the mutex.
- ☐ Multiple threads might pass through this "critical section" while a thread is blocked in `sem_wait(w_or_r)`

Question 4. [6 MARKS]

A program issues the following system calls on the original Fast File System (FFS) file system. The file system block size is 4KB and the size of an inode is 128 bytes. Assume that the file system is already mounted when the program starts and that all system calls succeed.

```
char buf[4096];
int fd = open("/cs/dw.txt", 0); // open in read-only mode
lseek(fd, (13*4096) + 2048, 0);    // seek to position (13*4096)+2048 from start of file
read(fd, buf, 4096);        // read 4k of data from file
```

Assume that the file buffer cache is empty. State the *minimum* number of the following types of blocks that will be read when the program above is run. Explain your answer for each block type.

1. Inode block(s)

2. Directory block(s)

3. Indirect block(s) (include single, double or triple indirect)

4. Other data block(s)

Question 5. File system consistency [9 MARKS]

On an ext2 or FFS (Fast File System) file system, consider the operation of removing a file of size 1024 bytes. Assume the directory that contains the file occupies one block. Assume the directory inode and the file inode are in different disk blocks.

Part (a) [4 MARKS]

Which of the following blocks must be updated? Check all that apply. For each block that must be updated, explain briefly how the values in the block change.

- ☐ inode bitmap
- ☐ data block bitmap
- ☐ file inode
- ☐ file data block
- ☐ directory inode
- ☐ directory data block

Part (b) [1 MARK]

What data is updated in the directory inode?

Part (c) [4 MARKS]

In each of the remaining questions, check all of the boxes that most closely explain what happens if a crash occurs after updating only the block(s) specified.

Inode Bitmap

- ☐ No inconsistency (it simply appears that the operation was not performed)
- ☐ Data leak (data block is lost for any future use)
- ☐ Inode leak (Inode is lost for any future use)
- ☐ Multiple file paths may point to same inode
- ☐ Inconsistent inode data (Some inode field does not match what is stored in data blocks)
- ☐ Something points to garbage

Data Block Bitmap

- ☐ No inconsistency (it simply appears that the operation was not performed)
- ☐ Data leak (data block is lost for any future use)
- ☐ Inode leak (Inode is lost for any future use)
- ☐ Multiple file paths may point to same inode
- ☐ Inconsistent inode data (Some inode field does not match what is stored in data blocks)
- ☐ Something points to garbage

Directory Inode

- ☐ No inconsistency (it simply appears that the operation was not performed)
- ☐ Data leak (data block is lost for any future use)
- ☐ Inode leak (Inode is lost for any future use)
- ☐ Multiple file paths may point to same inode
- ☐ Inconsistent inode data (Some inode field does not match what is stored in data blocks)
- ☐ Something points to garbage

Directory inode and Directory data

- ☐ No inconsistency (it simply appears that the operation was not performed)
- ☐ Data leak (data block is lost for any future use)
- ☐ Inode leak (Inode is lost for any future use)
- ☐ Multiple file paths may point to same inode
- ☐ Inconsistent inode data (Some inode field does not match what is stored in data blocks)
- ☐ Something points to garbage

Print your name in this box.

End of Examination