Homework 5
COM S 352
Fall 2021

1. (10 points) Linux provides a command-line utility `rm` that has the purpose of removing a file from a directory. When executing `rm` inside of the `strace` utility we can see that `rm` makes a system call to `unlink()`. Here is an example.

```
$ strace rm test.txt
...
unlinkat(AT_FDCWD, "test.txt", 0)       = 0
...
```

Explain why `rm` unlinks the file. How does unlinking result in the removal of the file? Why does Linux not have a system call such as `delete()` that `rm` could use to directly remove the file?

2. (10 points) Linux provides the command `mv` that has the purpose of "moving" a file from one file path name to another file path name. When executing `mv` inside of the `strace` utility we can see that `mv` makes a system call to `rename()`.

```
$ strace mv oldname.txt newname.txt
...
renameat(AT_FDCWD, "oldname.txt", AT_FDCWD, "newname.txt") = 0
...
```

Is there any difference at all between renaming a file and just copying the file to a new file with the new name, followed by deleting the old one? Explain.

3. (10 points) How many disk operations are needed to fetch the inode for the file `/var/log/boot.log`? Assume that the inode for the root directory is in memory, but nothing else along the path is in memory. Also assume that all directories fit in one disk block.

4. (20 points) In the Very Simple File System (VSFS) the inodes are kept together near the start of the disk. An alternative design is to allocate an inode when a file is created and put the inode at the start of the first data blocks of the file.

a. Assume a hard disk drive (HDD) is being used. What is an advantage to putting inodes at the start of the disk? What is an advantage to putting inodes at the start of the first data blocks of the file?

b. In the Fast File System (FFS) the inodes for a file are placed near the start of the same block group as the file's data blocks. Again assuming a HDD, explain an advantage to this approach rather than placing all the inodes near the start of the disk.

c. Now assume a Solid State Drive (SSD) is being used. Does the FFS approach still have any advantages or would the alternative design of putting the inode at the start of the first data blocks of the file be better?

5. (10 points) What would happen if the bitmap containing the information about the free disk blocks was completely lost due to a crash? Is there any way to recover from this disaster, or is it bye-bye disk?

6. (10 points) Some file systems allow disk storage to be allocated at different levels of granularity. For instance, a file system could allocate 4 KB of disk space as a single 4-KB block or as eight 512-byte blocks.

a. How could we take advantage of this flexibility to improve performance?

b. What modifications could be made to the bitmap free-space management scheme in order to support this feature?

7. (20 points) Consider a file system that uses Unix-style inodes (index nodes) to represent files. The system uses 16-KB blocks and 8-byte pointers. Multi-level indexing is used with an inode containing 12 direct pointers, 1 single indirect pointer, 1 double indirect pointer and 1 triple indirect pointer.

a. What is the maximum size of a file that can be stored in the file system? Show calculations.

b. For each of the following addresses describe how many blocks (including the inode and data block) need to be read to perform a read of the data at the address. Explain your answers.

4,096
32,768
1,048,576
268,435,456

8. (10 points) Suppose an SSD consists of raw flash with the following performance characteristics: reading one page takes 25µs, programming one page takes 200µs and erasing one block takes 1500µs (note these refer to physical SSD pages and blocks). The following figure represents the current state of the SSD.

Table: 100->3 101->4 102->5

Block:	0				1			
Page:	00	01	02	03	04	05	06	07
Content:	a1	a2	a3	b1	b2	b3		
State:	V	V	V	V	V	V	E	E

How much time will it take for garbage collection to reclaim physical block zero? Assume all operations must be performed sequentially (no parallel hardware optimization) and none of the pages are cached. Explain your result.