CS232 Operating Systems
Assignment 1: Storage and Filing

CS Program
Habib University

Fall 2024
Due Date: 22 September 2023 @ 11:59PM

Note to Students

1. **chatGPT or AI tools:** You are not allowed to use any AI tool to obtain the code for this assignment. The faculty reserves the right to conduct a viva and use appropriate AI-tool checking software to verify original contribution. The faculty reserves the right to award a zero (0), and report the student to the Office of Academic Conduct should he/she be found using such tools.

2. **Plagiarism Policy**: Zero tolerance for plagiarism. Every submission will be screened using a plagiarism detection software. Offenders will be reported to the Office of Academic Conduct and obtain zero (0). This applies even if the code is obtained from an online repository like github without proper attribution. We expect you to credit all sources used for completion of this assignment.

3. **Submission:** You will submit one compressed zip file, naming it with your student ID (CSxxyyyy.zip), within the Assignment 1 submission module on LMS. The zipped file will contain the following four items:

   (1) `makefile` for compiling and running of your C code. It should contain `compile`, `build` and `clean` targets.

   (2) `.c files`: C codes corresponding to each question.

   (3) `.sh files`: Linux scripts corresponding to each question.

   (4) PDF report.

4. **Penalties** You will be penalized for the following:

   (a) code does not compile: -20% marks.

   (b) code has warnings (compile with -Wall): -10% marks.

   (c) makefile is absent: -10% marks.

   (d) program does not work properly or crashes: -20% marks

   (e) late submission: -10% marks for missing the deadline + -5% $\times$ num. of days.

5. **Viva:** The faculty reserves the right to call you for a viva.

6. **Marks:** Marks are highlighted in the table below.

Q1	Q2	Q3	Q4	Q5	Q6	Total
5	(5+5)	10	10	5	(5+5)	50

Table 1: Allocated marks for each question.

Feature	Hard drive	Solid State Drive
Storage Capacity	1 TB	250 GB
RPM	7200	-
Cache	64MB	64 MB
Data Transfer Rate	210 MB/s	600 MB/s
Average Seek	8 ms	-
Price	$50	$150

Table 2: Product Specifications of HDD and SDD.

1 Sequential and Random I/O rates

In chapters 37 and 38 we evaluated $S_{I/O}$ and $R_{I/O}$ for different hard drives and RAID [1]. Assuming a random read of 100KB and a sequential read of 100MB, determine the following for the two drives 2:

1. T_{Seek}
2. $T_{Rotation}$
3. $T_{Transfer}$
4. $T_{I/O}$
5. $R_{I/O}$
6. $S_{I/O}$

2 Storage

Write a (a) Linux script titled '`disk.sh`' and (b) a C code named '`disk.c`' where you:

1. Read the details about the hard disk, especially its total size *total_size*, and the free space available on the disk *free_space*.
2. Take two inputs from the user as arguments of the function, i.e., a sample sentence *sentence*, and a suggested name for a file *"name of file.txt"*.
3. Create a file named *"name of file.txt"* containing several *sentence* instances, equal to the size of $\frac{<free\_space>}{100{,}000}$

3 Virus 1: Angry Parent

In class we discussed the very powerful command `rm -rf *` which recursively deletes all files in a folder and all subsequent folders. You are to develop a virus called "*angry_parent*." The angry parent will remove all video files from your suggested folder and all subsequent folders. Video files have these files extensions (.mp4, .avi, .mov, .wmv, .mkv, .flv, .webm, .3gp, .m4v). The Linux version should have a front friendly face, and an evil hidden compartment. The front friendly face should use 'zenity' [2], to develop a GUI based simple calculator (with just four operations: +, -, *, and /), which (very politely) asks the user sudo privileges to download an install some random tools needed for the software to install. The evil hidden compartment then runs the *angry_parent* virus and removes all video files present in the `pwd` and all subsequent folders.

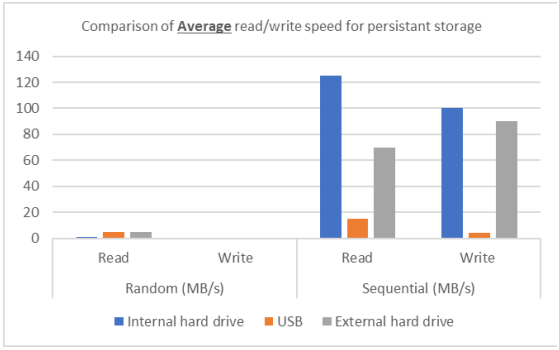

Figure 1: Sample graph for this question, showing a comparison of persistant storage devices.

4 Sequential and Random I/O rates

In chapters 37 and 38 we discussed hard drives and RAID [1]. Here you will analyze the average random and sequential read/write speeds of your internal hard drive, USB flash drive, and an external hard drive, (I am hoping you have all three, in the event you don't please feel free to share hardware for the purpose of this lab).

Note: Your read/write speed for a flash drive will differ significantly based on whether you use USB 2.0, or 3.0, or 3.1 for this homework. Similarly, your read/write speed will differ greatly among yourselves depending upon you having a hard drive with 7,200 RPM or 5,400 RPM, or whether you are using an SSD or not.

You will be using two Linux commands in your script: `dd`, and `fio` tool. Make sure `fio` is installed in your Linux system by typing: `> sudo apt-get install fio`

- Sequential Read/Write size: Use file sizes: 50MB, 100MB, 150MB, 200MB and 250MB.
- Random Read/Write size: Use these file sizes: 1KB, 5KB, 10KB, 15KB, 20KB.

Your submission must include necessary Linux script and a graph like the one shown in Figure 1, showing the average of the random/sequential read/write speeds.

Note: You are welcome to modify the code you developed in Q2 to develop the random and sequential files of different sizes.

5 Merge all files

Write a program in C language that merges many text files into one file. The program should ask the user for the names of the input files and the output file. The command prompt should look as follows:

```
> ./merge a.txt b.txt c.txt d.txt output.txt
```

Here files `a.txt` to `d.txt` are merged as one `output.txt`.

6 Virus 2: Envious friend

In class we discussed how reading a file takes more time, if the file is accessed after passing through several folders and subfolders. We also learnt the concept of a 'hard link' and a 'soft link.' Here we will make a virus called "envious_friend." Your envious friend doesn't like the fact that you finish your work early. Therefore, he/she wants your system to slow down significantly. The virus envious_friend does so, by having the following:

1. A friendly front face, where a simple calculator (with just four operations: +, -, *, and /) functions to operate, which (very politely) asks the user sudo privileges to download and install some random tools needed for the software to install. (You are welcome to reuse the code developed for Q4 here).

2. An evil hidden compartment will

   (a) recursively create folders within folders such that the first folder is named 'a' and the last folder is called 'z,' then.

   (b) moves (not copies, but moves) all files from the parent folder to this new location,

   (c) place 'soft links' (not hard links, but soft links) of these files to the original parent folder.

   (d) Hides this folder and all subfolders.

   In other words, the evil compartment will do the following:
   Original location: `> /home/Documents/Code/disk.c`
   Final location: `> /home/Documents/Code/a/b/c/d/.../x/y/z/disk.c`

You are to develop independent C and Linux codes for this question.

References:

1 R. H. Arpaci–Dusseau and A. C. Arpaci–Dusseau. "Operating Systems: Three Easy Pieces," Arpaci–Dusseau Books, LLC, 2019.

2 Chap. 11 – Gui Development: B. Wajid, H. Iqbal and M. Jamil. "Linux programming for the faint of heart," Sabz Qalam, 2020 (ISBN #: 978-969-7941-00-1). (The book is available in HULMS – OC course folder at "Home" tag)