ECE5725: Homework 1

Hongxi Chen (hc2325)

September 8, 2025

Question 1: Academic Integrity

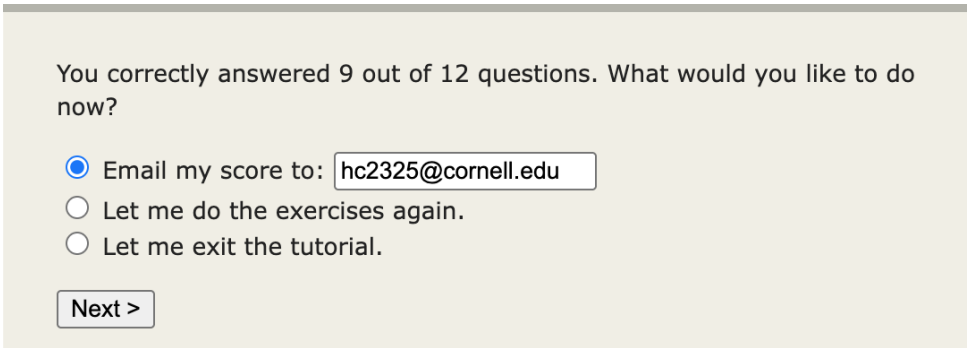


Figure 1: Question 1 Screenshot

Question 2: SD Card Setup

This task has been completed according to the Canvas guides.

Question 3: Linux File Permissions

In Linux, file permissions control the actions that users can perform on files and directories. Permissions are assigned to three user classes: the **owner** (u), the **group** (g), and **other** (o). There are three primary permissions:

- **Read (r):** View the contents of a file or list the contents of a directory.
- **Write (w):** Modify a file or add/remove files in a directory.
- **Execute (x):** Run a file as a program or enter (cd into) a directory.

These are often represented by an octal (numeric) code, where each digit corresponds to the owner, group, and others, respectively. The digit is the sum of the values for its permissions: read=4, write=2, and execute=1.

Permission `777`

Permission `777` grants full rights to everyone.

- **Owner**: $7 = 4 + 2 + 1 \rightarrow (rwx)$
- **Group**: $7 = 4 + 2 + 1 \rightarrow (rwx)$
- **Others**: $7 = 4 + 2 + 1 \rightarrow (rwx)$

Danger: This permission is dangerous because it allows any user on the system to read, modify, delete, or execute the file/directory. This can lead to accidental or malicious modification of important data or the introduction of harmful scripts onto the system.

Permission `644`

Permission `644` is a common default for files. It allows the owner to edit the file, while others can only read it.

- **Owner**: $6 = 4 + 2 + 0 \rightarrow (rw-)$
- **Group**: $4 = 4 + 0 + 0 \rightarrow (r--)$
- **Others**: $4 = 4 + 0 + 0 \rightarrow (r--)$

Permission `700`

Permission `700` provides private access exclusively to the owner.

- **Owner**: $7 = 4 + 2 + 1 \rightarrow (rwx)$
- **Group**: $0 = 0 + 0 + 0 \rightarrow (---)$
- **Others**: $0 = 0 + 0 + 0 \rightarrow (---)$

Question 4: Server Login and Directory Permissions

The screenshot is Figure 2.

Question 5: File Creation and Permissions

The screenshot is Figure 3.

Question 6: The `df` Command

The `df` (disk free) command displays the amount of available disk space for the system's file systems. The `-h` flag makes the output human-readable by showing sizes in units like Gigabytes (G) and Megabytes (M).

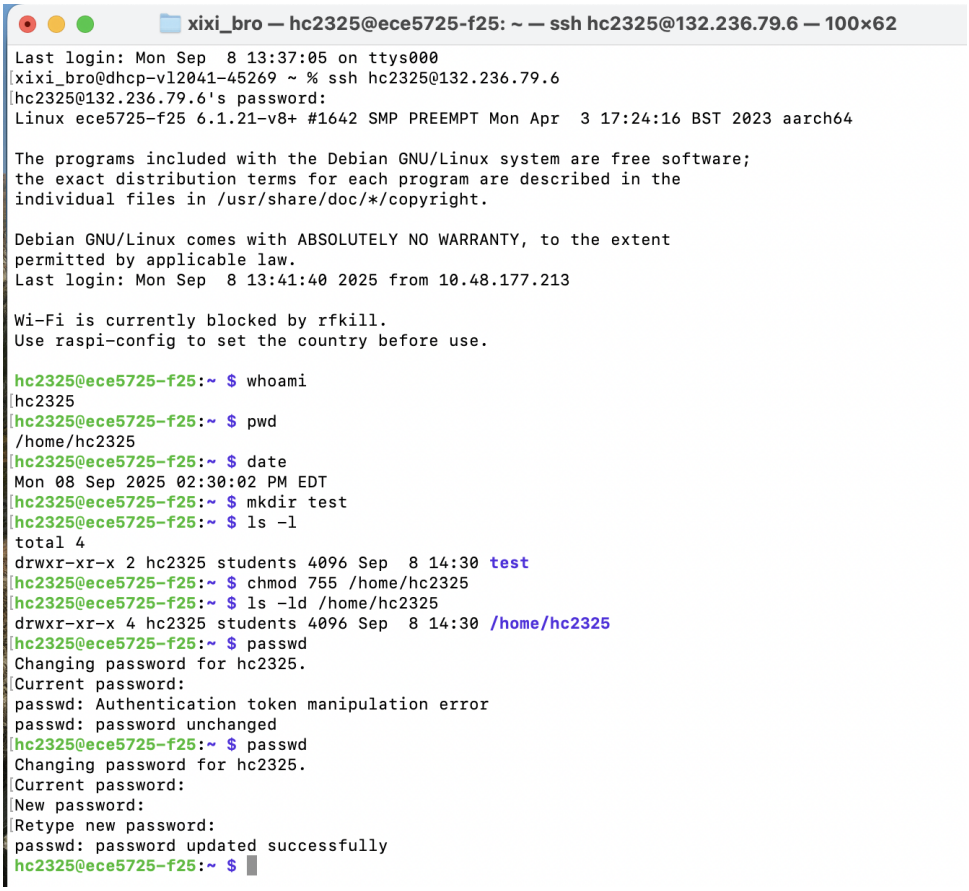


Figure 2: Question 4 Screenshot

```
hc2325@ece5725-f25:~ $ ls
test
hc2325@ece5725-f25:~ $ cd test
hc2325@ece5725-f25:~/test $ echo "hc2325, Hongxi Chen" > HW1.txt
hc2325@ece5725-f25:~/test $ ls
HW1.txt
hc2325@ece5725-f25:~/test $ chmod 600 HW1.txt
hc2325@ece5725-f25:~/test $ ls -l
total 4
-rw------- 1 hc2325 students 20 Sep  8 14:38 HW1.txt
hc2325@ece5725-f25:~/test $ cat HW1.txt
hc2325, Hongxi Chen
hc2325@ece5725-f25:~/test $
```

Figure 3: Question 5 Screenshot

Analysis of the /home entry

The screenshot(Figure 4) shows the output of the `df -h` command. The line for the `/home` directory is:

`/dev/sda1  916G  3.7G  866G  1% /home`

This output shows that the `/home` partition has a total **size** of 916G, of which 3.7G is **used** and 866G is **available**.

What is **unique** about this server's setup is that the user home directories (`/home`) are located on a separate physical partition (`/dev/sda1`) from the operating system's root filesystem (`/dev/root`). This is a standard practice for managing multi-user systems

because it isolates user data from system files. If the user partition fills up, it will not crash the operating system, making the server more stable.

```
[hc2325@ece5725-f25:/home $ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        15G   11G  3.0G  78% /
devtmpfs        667M     0  667M   0% /dev
tmpfs           925M  8.0K  925M   1% /dev/shm
tmpfs           370M  1.6M  369M   1% /run
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
/dev/sda1       916G  3.7G  866G   1% /home
/dev/mmcblk0p1  255M   33M  223M  13% /boot
tmpfs           185M   24K  185M   1% /run/user/1000
tmpfs           185M   20K  185M   1% /run/user/1033
tmpfs           185M   24K  185M   1% /run/user/1052
tmpfs           185M   24K  185M   1% /run/user/1040
tmpfs           185M   24K  185M   1% /run/user/1064
tmpfs           185M   24K  185M   1% /run/user/1027
tmpfs           185M   24K  185M   1% /run/user/1048
tmpfs           185M   24K  185M   1% /run/user/1026
tmpfs           185M   24K  185M   1% /run/user/1008
tmpfs           185M   24K  185M   1% /run/user/1023
tmpfs           185M   24K  185M   1% /run/user/1024
tmpfs           185M   24K  185M   1% /run/user/1022
```

Figure 4: Question 6 Screenshot.

Question 7: The ps Command

The `ps` command reports a snapshot of the current processes. I used the following commands.

1. `ps aux`: Show all processes for all users.
2. `ps aux | wc -l`: Pipe the output to `wc -l` to count the total lines (processes).
3. `ps aux | grep hc2325`: Pipe the output to `grep` to search for lines containing hc2325.

The screenshot is Figure 5.

Question 8: Raspberry Pi vs. Laptop

Component Correspondence

- **Laptop Disk (SSD/HDD):** Corresponds to the **MicroSD Card** on the Raspberry Pi, which holds the operating system and user data.
- **Laptop Memory (RAM):** Corresponds to the onboard **LPDDR SDRAM** which is integrated directly into the Pis main chip.

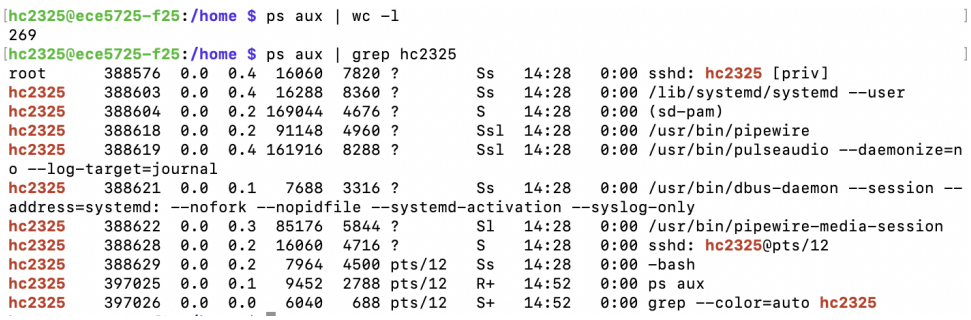


Figure 5: Question 7 Screenshot.

- **Laptop Processor (CPU):** Corresponds to the **ARM-based System on a Chip (SoC)**, which integrates the CPU, GPU, and other components into a single package.

Advantages of Raspberry Pi over a Laptop

- **Cost:** Significantly cheaper than any laptop.
- **Size and Portability:** Extremely small and lightweight, making it ideal for embedded and space-constrained projects.
- **Power Consumption:** Uses very little power, allowing it to run off a phone charger or battery pack for extended periods.
- **GPIO Pins:** The GPIO pins allow it to directly interface with sensors, motors, and other electronic components, which is not possible on a standard laptop.

Disadvantages of Raspberry Pi versus a Laptop

- **Performance:** The processing power and memory are much lower than a modern laptop, making it unsuitable for computationally intensive tasks.
- **Storage:** MicroSD cards are slower and have lower capacity and a shorter lifespan than laptop SSDs or HDDs.
- **Compatibility:** The ARM architecture means it cannot run software compiled for standard x86 processors (though this is becoming less of an issue).
- **Integrated Peripherals:** It lacks a built-in screen, keyboard, and battery, which must all be supplied externally.

Question 9: top vs. htop

top (table of processes) is the traditional command-line utility for displaying a real-time view of running system processes. It is pre-installed on nearly all Unix-like systems. It is functional but basic in its presentation and interaction. **htop** is an enhanced, interactive process viewer. It is considered a more user-friendly and powerful alternative to top.

htop is generally preferable for several reasons:

- **User Interface:** htop presents data in a colorful, clearer layout.
- **Interactivity:** htop allows you to directly interact with processes using function keys or a mouse, such as killing, renicing (changing priority), or filtering processes without typing commands.
- **System Information:** htop provides a more comprehensive, graphical view of system resources like CPU, memory, and swap usage at the top of the screen.

Question 10: Debian Release Characters

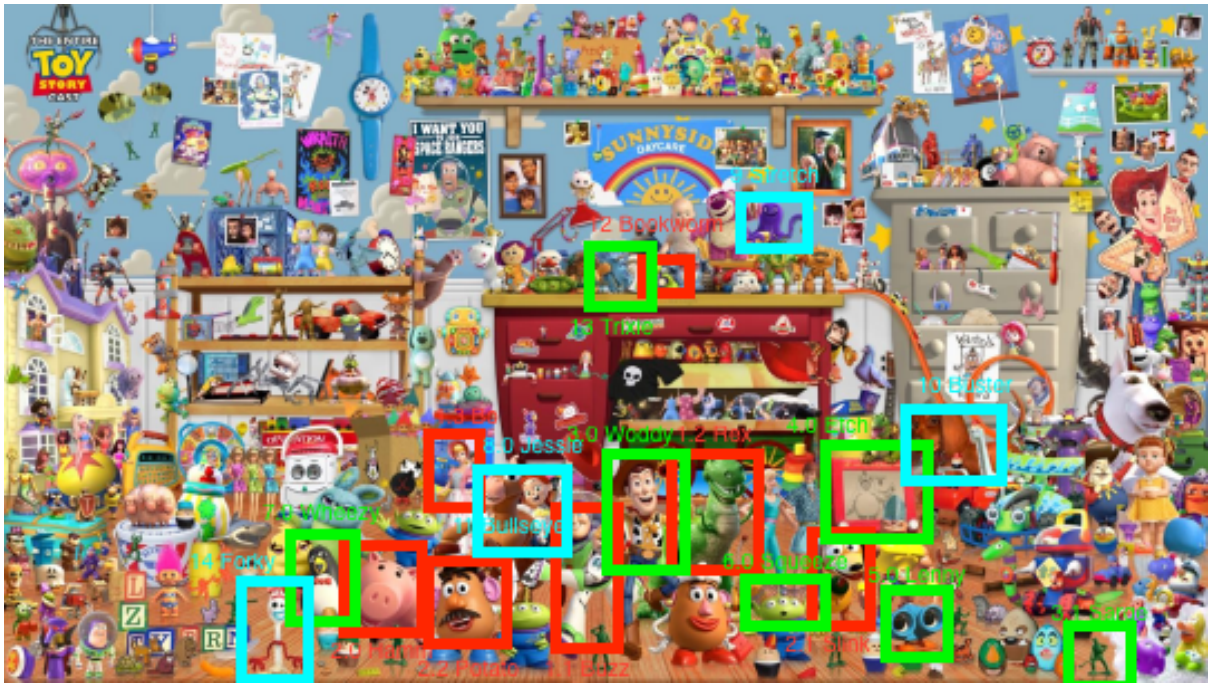
Stable Releases (Beginning with 1.1):

- 1.1: **Buzz** (Buzz Lightyear)
- 1.2: **Rex** (Rex the Dinosaur)
- 1.3: **Bo** (Bo Peep)
- 2.0: **Hamm** (Hamm the Piggy Bank)
- 2.1: **Slink** (Slinky Dog)
- 2.2: **Potato** (Mr. Potato Head)
- 3.0: **Woody** (Sheriff Woody)
- 3.1: **Sarge** (Sergeant of the Green Army Men)
- 4.0: **Etch** (Etch A Sketch)
- 5.0: **Lenny** (Lenny the Binoculars)
- 6.0: **Squeeze** (Squeeze Toy Aliens)
- 7.0: **Wheezy** (Wheezy the Penguin) - *RPi Foundation based here*
- 8.0: **Jessie** (Jessie the Cowgirl) - *RPi Foundation based here*
- 9: **Stretch** (Stretch the Octopus) - *RPi Foundation based here.*
- 10: **Buster** (Buster the Dog) - *RPi Foundation based here.*
- 11: **Bullseye** (Bullseye the Horse) - *RPi Foundation based here.*
- 12: **Bookworm** (Bookworm) - *Current RPi OS basis.*

Future and Codename-Only Releases:

- 13: **Trixie** (Trixie the Dinosaur)

- 14: **Forky** (Forky)

Special Releases:

- **Sid** (Sid Phillips)