Previous Year Practice Problems on

Shell Commands and Shell Scripting

1. The Semester Scramble: Taming the Course Chaos with Shell Power!

This semester is proving to be a real challenge! Between lectures, assignments, and projects, your course materials are scattered across your computer like leaves in a hurricane. To combat the chaos and reclaim your sanity, you've decided to use the power of the Linux shell to organize your course content.

Basic things to do-

    a. Suppose, initially your folder looks like this.
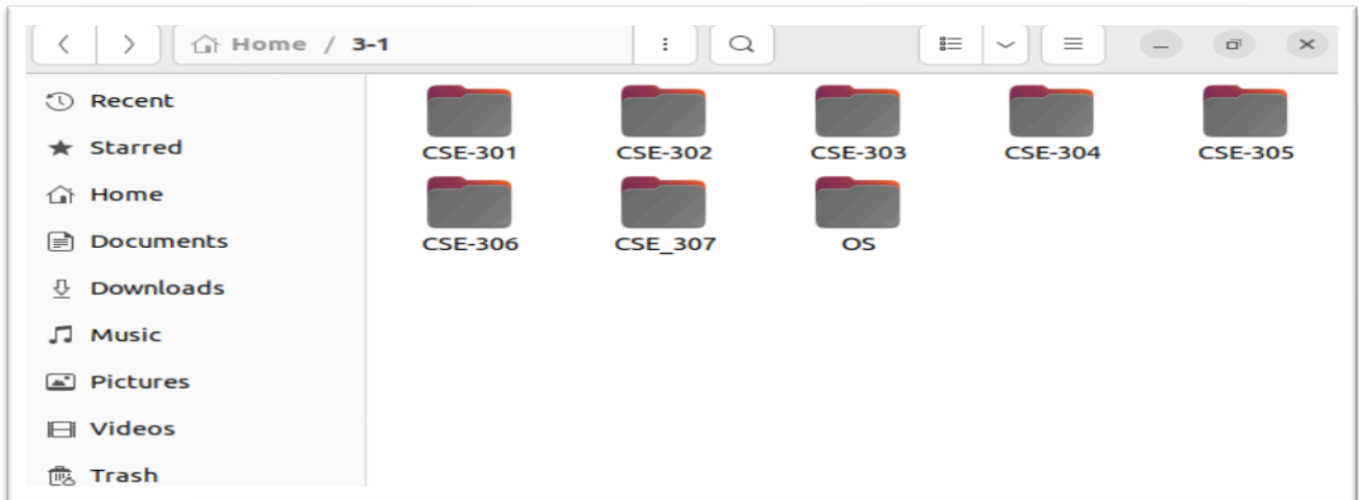
    b. Inside each folder are two directories named SecA and SecB. Inside, you can keep any materials you like (using terminal ofc, creating text files, shell files, other folders, etc). You must keep something in each folder as you like.

    c. Change the permission of some of your folders/ files.

    d. Rename the name of the OS folder with the course code.

    e. Create a text file listing all the folders and files within the "3-1" directory. This text file should be saved in a separate directory, renamed in point 4.

    f. Now, copy this file into another directory named "Backup" located in another directory under Top Level Directory (root), but not under the "Home" directory. (Use relative path to do it)

    g. Sort the file contents alphabetically in ascending order and count the lines and words. Save the line and word count in that text file too.

    h. Delete the files located in all of your theory course folders.

    i. Last but not least, update your file with the **system name**, **your full name, ID, Section,** and **all of your commands used in this assignment.**

    j. Rename the text file as "your_student_id_1.txt" and save it.

Additional instructions:
- Use all the commands taught in your class.
- Any customization/addition is highly appreciated.

2. Write a bash script that prompts the user to choose their favourite fruit from a list of options (e.g., apple, banana, mango, etc). Depending on the user's choice, the script should print a corresponding message. Use at least 5 fruits including individual messages.

3. Write a bash script to calculate the factorial of a given number. Implement a function named factorial that takes a single argument and returns the factorial of that number. Then, prompt the user to enter a number, calculate its factorial, and display the result. Then develop another function named nPr and nCr that will take arguments n and r to calculate their corresponding values. nPr and nCr functions shall use the factorial function.

4. Write a bash script to perform basic arithmetic operations (addition, subtraction, multiplication, division) on two numbers entered by the user. Define functions for each operation and use a while loop to repeatedly prompt the user for input until they choose to exit.

5.

> Write a shell script that **takes two integers from the command prompt, and determines the prime factors** of the first argument **between these two integers.** Assume, the first argument is larger than the second one.
>
> **Example:** Arguments: 70 4; Prime factors (of 70): 5, 7

6.

> Write a shell script that will **display the following series up to N (take user input) terms:**
>
> 0 1 8 9 64 25 216 49 · · ·

7. Given an array of floating points arr[ ] of size n, the task is to rotate the array elements to the left by d positions.

8.

Imagine that by exploiting some weakness in the system, you got access to the terminal of a Linux system of a very valuable PC that you were trying to hack for a while!

However, you only have access to the terminal of the system for 20 minutes. Someone else is currently using that system so you can't use the GUI anyway.

However, that shouldn't be a problem for you now that you know shell commands. Now, **use appropriate shell commands to complete the following steps** and get some important information out of that system in the 20 minutes that you have!

A. 1. Create a Directory Named DataRead on *Desktop*.
    2. Now write the names of all directories and files inside the "*sys*" directory which is inside root of the OS. Write it in a file named "fileinfo.txt" inside the folder DataRead.
    3. Then, In the same way, append the list of Directories and Files inside the directory "*devices*" and "*kernel*" inside *sys* to the same file ("fileinfo.txt").
    4. Then copy the folder named DataRead to the folder named "*tmp*" inside root.
    5. **Rename** it to Tempfile.
    6. Cover your tracks and Delete the folder named DataRead from Desktop!

B. 1. Now, calculate how many folders and direct files were inside sys, devices, and kernel from the file you stole (fileinfo.txt). Save that count at the bottom of fileinfo.txt.
    2. Change the permission of fileinfo.txt in such a way that "others" may read, write and execute it. Keep all permissions for users and groups unchanged. Save the newly modified permission status at the bottom of the same file (fileinfo.txt).