BCSE – 3rd year – 1st Semester – 2022

Assignment Sheet – I
Operating Systems Laboratory

1. Write a shell script that has 2 user created variables, uv1 and uv2. Ask for the values of the variables from the user and take in any values (real/integer/character) for the 2 variables. Test the program for different types of uv1 and uv2.
   (a) Print them as: (i) value of uv1 followed by value of uv2 separated by a comma and (ii) value of uv2 followed by value of uv1 separated by the word "and".
   (b) Print the variables in reverse order [If uv1 is 1234, then output should be 4321]
2. Write a shell script to count the number of lines in a file. Test if the file is present. If not, create and write.
3. Write a shell script that counts the number of ordinary files (not directories) in the current working directory and its sub-directories. Repeat the count of files including the sub-directories that the current working directory has.
4. Write a shell program to duplicate the UNIX **rm** command with the following features:
   a. Instead of deleting the files, it will move them to a **my-deleted-files** directory. If the file already exists in the **my-deleted-files** directory, then the existing file (in the **my-deleted-files**) will have the version number zero (0) appended to it and the newly deleted file will have version number one (1) appended to it. Go on incrementing the version nos., if required.
   b. The command will have a switch **-c** that will clear the entire **my-deleted-files** directory after asking for confirmation.
5. Write a script called birthday_match.sh that takes two birthdays of the form DD/MM/YYYY (e.g., 15/05/2000) and returns whether there is a match if the two people were born on the same day of the week (e.g., Friday). And then find out the age/s in years/months/days.
6. Write a shell script that accepts a file name as an input and performs the following activities on the given file. The program asks for a string of characters (that is, any word) to be provided by the user. The file will be searched to find whether it contains the given word. If the file contains the given word, the program will display (a) the number of occurrences of the word. The program is also required to display (b) the line number in which the word has occurred and no. of times the word has occurred in that line (Note: the word may occur more than once in a given line). If the file does not contain the word, an appropriate error message will be displayed.
7. Extend the shell script written in (6) to perform the following task: User is asked to enter two different patterns or words. The first pattern will have to be matched with the contents of the file and replaced by the second pattern if a match occurs. If the first pattern does not occur in the file, an appropriate error message will be displayed.

Last date of submission of Assignment I: by August 29th /30th, 2022
Test/Viva voce on Assignment I: August 29th /30th, 2022

**Additional credit will be given if you complete the following assignments in two weeks following the last date as mentioned above.**

**1) [Don't use Linux commands to write the shell script.]**

Write a shell script with the name <*last four digits of your roll no.*>_*firstname*.sh that repeatedly displays the following menu:

[1] Display greetings [2] List large files [3] Disk usage [4] View Log File [5] Exit

Your choice >

---

If you enter any number other than [1 – 5], the script gives you an error message.

---

If you enter 1 (Display greetings), the script will:

- Display "Hello <*user_name*> good morning/evening" .
- morning/evening will be displayed depending on time.

---

If you enter 2 (List large files), the script will:

- Prompt for a size in bytes
- List the names and sizes of all files greater than or equal to the specified size. If there are no files of that size, a blank list with headings only will be printed.

---

If you enter 3 (Disk usage), result will be displayed appropriately.

---

When you choose command 4 (View Log File), the script will display a file named logfile.txt with appropriate headings.

Your script will create this file as follows: Every time a command is entered, you will append a line to a file named logfile.txt. The line will have the user name, the menu choice, and the output of the date command, separated by percent signs. The menu choice must be given in words, as shown in the sample output. Don't just give the menu choice as a number—when you look at the log file a month from now, you want to know that a user did a "disk usage" command, not "3." Invalid commands and the exit command do not have to be entered in the log file if you don't want to.

---

If you enter 5 (Exit), the script will exit.

---

Menu options 1, 2, and 4 (find user, list large files, and view log file) must be implemented as separate functions. Add any other functions that you feel will make your job easier.

2)

Add a system call (developed by you) to your linux shell.

https://medium.com/anubhav-shrimal/adding-a-hello-world-system-call-to-linux-kernel-dad32875872

https://www.kernel.org/doc/html/v4.10/process/adding-syscalls.html