NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES ISLAMABAD

OPERATING SYSTEMS Spring, 2021

ASSIGNMENT 01

Given Date: March 12th, 2021 **Due Date: March 20th,2021**

Instructions

- Zero marks will be awarded to the students involved in plagiarism.
- All the submissions will be done on google classroom.
- You have to submit zip folder named as ROLL_NUM_SEC (e.g., i20-1234_A) containing three script files named as Q1, Q2, Q3. Naming convention has to be followed strictly. As, it will carry 2 marks for proper naming convention and correct submission.
- Each part will carry different marks. Marks distribution is given with each part.
- Read the complete instructions given in each part. In case of any query you can email to mehran.khan@nu.edu.pk shehr.bano@.nu.edu.pk or zia.urrehman@nu.edu.pk
- Be prepared for viva or anything else after the submission of assignment for two weeks.

Shell Scripting

Question 1.

Write a script that takes two parametric variables: **[10]**

./script pattern_option number

where

pattern_option = {left, right, full}, number = could be any positive integer

and output the pattern depending on the input parameters as shown in Fig 1. Generate error messages for invalid input parameter.

left	*right*	*full*
*	*	*
**	**	***
***	***	*****
****	****	*******
*****	*****	*********
****	****	*******
***	***	*****
**	**	***
*	*	*

Figure 1: Output for number=5

QUESTION NO. 02 [20]

Write a script that displays a main menu and perform tasks based on the input value.
Valid *input values = {1, 2, 3, 4, exit}*.
The different options *1,2,3,4* will display the output as follow:

1. Input a filename from user and display permissions of that particular file. Then invert the permissions e.g. If permissions were *r-x* change them to *-w-*. Then again display the updated permissions of that file. [5]

2. Input a *filename* and a *string* and search it in the file. Output the lines of file where that *string* is found. But if the *string* contains a *dot(.)* it means any character can fill the place. For example: [5]

   string = c.t = {cat, cot, c t,}

3. Create a file *dummy.txt* and add the content of all the files in the current directory to *dummy*. But copy the content in such a way that if files in current directory = {f1, f2, f3, f4, ..., fn}. Then copy first *N* lines of files at even location *{f2, f4,.}* and last *N* lines of files at odd location *{f1,f3,. }*. Input value of *N* from user. [5]

4. Input a filename from user and check modified date of that file. If modified date is greater than 24 hours from the current time change the modified date to current date. Along with displaying the output on terminal, maintain a log file that contains the information of the script. Format of the log file is given below: [5]

Format of the log File

Option 01 selected at *date and time*
File name: *filename.txt*
Permissions of *filename.txt*: *Show permissions*
Permissions changed
Updated Permissions of *filename.txt*: *Show permissions*

Option 02 selected at *date and time*
Filename: *filename.txt*
String: *string*
Output all the lines in *filename* where *string* is found.

Option 03 selected at *date and time*
Files at odd location: *f1, f3, f5*
Files at even location: *f2, f4*
Dummy.txt is created and *N* lines of each file copied in it.

Option 04 selected at *date and time*
Filename: *filename.txt*
Current modified date: *date*
Modified time updated or not

Option exit
Script terminated at *date and time*.

QUESTION NO. 03 [18]

Fig 2 shows the basic hierarchy of the files. Input (read) the name of the *Main Directory*. In the *Main Directory* there are multiple directories *Dir_1, Dir_2, ..., Dir_N*. In each subdirectory there are multiple files with different extensions. Now insert a loop that takes a string value as input and terminate when user enter *any string starting with e*. The hierarchy of the files will change when user will enter either *forward or backward*.

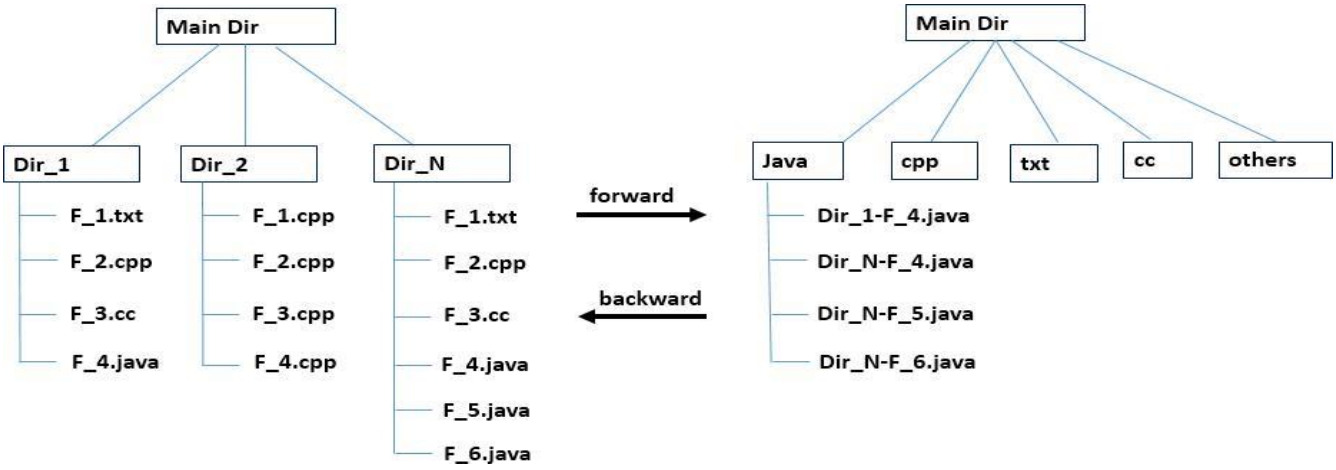

Figure 2: Hierarchy of the main directory

1. **Forward** **[7]**

   Create 5 different directories namely *java, txt, cpp, cc and other*. Copy each file in the directories depending on their file extension. Before copying the file follow the protocol to name the file as *DirectoryName-Filename.extension*. Delete all the extra folders from *main Dir* namely *Dir_1, Dir_2,. , Dir_n.*

2. **backward** **[7]**

   Now in the backward cycle, extract name of directories from the file name. Create the directories again and copy the files to their original directory restoring their default names.