CSP203: Software Tools & Technologies Lab

Assignment-1

Due Date: 31-August-2024 11:59 PM

Maximum Points: 100

Instructions:

1. Prepare all your solution files in a zip file and name it as <ROLLNO.zip> and submit on canvas.
2. Your zip file should also include a readme file, which should have instructions to execute your programs. Readme carried 5 points.

Question-1:

Consider an input file dir.txt, which has many lines in the following format

name:department:type

Where name represents the username, department represents the department in which a user belongs, type can be either student, staff, or faculty.

One example of the input file is
user1:CSE:student
user2:ME:faculty
user3:CSE:staff
user4:ME:student
user5:CSE:faculty

Write a shell script called dir.sh which takes the input dir.txt and produces a directory structure /type/department/name for each entry in the dir.txt

For example, the directories produced for the input dir.txt should be as follows.

student/CSE/user1, student/ME/user4, staff/CSE/user3, faculty/ME/user2, faculty/CSE/user5.

Question-2:

Write a script, **digits.sh**, to count how many of the numbers between 2000 and 7500 (inclusive) that contain 4 in one or more of their digits, and print those numbers.

Question-3:

Implement a shellATM machine using the shell script with the following features.

1. *Account.txt* contains several lines in the below format
   Account Name, Card Number, Email ID, Balance

Account name: It should contain First name followed by Last name (with space separated). First and Last names start with capital letters followed by any number of lower letter alphabets.

Card Number: It has 16 digits in the below format <4 Digits><space><4 Digits><space><4 Digits><space><4 Digits>

Email ID: It has the format username@Domain.
The username should start with an alphabet (lower/upper) followed by any number of alphabet or digits. Domain should use <string>.<string>.<string>, where strings contain only lower Alphabets.

Balance: Any number >= 0.

2. *Credentials.txt* has the below format.
   Card Number, Password.

Card number format follows as listed in *Account.txt.* Passwords can be any string.

Step-0: Your shellATM script upon execution should show a welcome screen as follows.

**** Welcome to My shellATM ****
**** Press any key to continue ***

Step-1: Once any key is pressed, the terminal screen should seek the credentials of Card Number and Password. If both the Card Number and Password match as listed in Credentials.txt, it should authenticate the users. Otherwise, it should ask to enter the details again. [Hint: You can use grep commands here]

Step-2: A successful authenticated user should get the following options.

1. Withdraw cash
2. Deposit cash
3. Settings
4. Exit

Withdraw Cash/Deposit cash options should accept a non-negative number and subtract/add the cash amount entered by the user. Once the operation is complete, your terminal should show the updated balance on the screen. *[Hint: Use grep with regular expression to validate the number and sed for updation of balance].*

Settings: Option should provide the user the user to change the email address. The new account and email address should use the same syntax as listed in the *Account.txt.* Any invalid entry should throw an error. If the new entries are valid, you should update the Account.txt. *[Hint: You can use grep for checking the patterns and sed for replacement]*

After completion of any of the options in Step-2, the terminal should go to the welcome screen.

Suggestion:

1. Your shell script should use as many linux commands as possible to minimize the work.
2. You can use **read** to get the input from the user.