McGill University
School of Computer Science

COMP-206

Mini Assignment #2

Due: January 30, 2019 on myCourses at 23:30

Command-line Bash Script

Create a script called **InformationExtractor.sh** that will be used by you to automate the process of searching and filtering information then logging it to a log file.

This bash script will be used at the command-line as follows:
./InformationExtractor.sh *directoryName*
Where ***directoryName*** is provided by the user at the command-line and is the directory name where the information will be stored. The directory name must be a single word (it cannot have spaces or other white characters).

The script will do the following in this order:

- Clear the screen
- Next, using absolute path, the script changes directory to your home directory.
- Next, it verifies whether a subdirectory called ***directoryName*** exists. If it does not exist it creates that directory.
- Next, it changes directory moving into ***directoryName*** using relative path.
- Next, it verifies whether a file called **extractor.log** exists. If it does not exist it creates that file which will serve as a log file where we log information regarding the execution of our script.
- Next, it verifies whether a file called **Information** exists. If it does not exist it creates that file which will serve as the file where we store the needed information.
- Next, the script should write to the log file the following message: "*Information file already exists*" if the file already existed or "*Information file is created*" otherwise.
- Next, the script should change the access rights to the file Information so that the owner can read and write, the group members can only read and no access rights for all the others.
- Next, the script should append to the log file the following message: "*Access right changed for Information file*".
- Next, the script will check the list of all the users connected to the server and write to the **Information** file the list of users of only those having one of the following letters in their user names: a, b, c or d. The list should consist of usernames only without any other related information.

- Next, the script should append to the log file the following message: "*List of user names has been added to the Information file*".
- Next, the script should append to the log file the number of users that were listed previously: "*The number of found users is: **xx***". Where xx is the number.
- Finally the script terminates by displaying to the screen: "Execution of InformationExtractor script has ended successfully!"

WHAT TO HAND IN

Everything must be submitted to My Courses before the due date. Remember that you can hand in your assignment up to two days late but there will be a penalty of 5% each day. After that, your assignment will not be accepted. Please hand in the following:

- A single Bash script called **InformationExtractor.sh**

HOW IT WILL BE GRADED

The assignment is worth a total of 20 points.

- 3 points - Creating the directory and files
- 3 points - Testing existence of directory and files
- 2 points - Setting the proper access rights
- 2 points - Logging messages to the log file
- 2 points - Writing the information to the information file
- 4 points - Searching and filtering the list of users
- 3 points - Finding the number of users that match the search pattern
- 1 point - Display of the "end of execution" message to the screen

GRADING RULES

The following rules are followed by the TA when grading assignments:

- A program must run in order to get a grade (even if it does not run well). If it does not run (does not compile) it will receive a zero. (Make sure to run your programs from Trottier – they sometimes do not run the same from home when logging in using putty.)
- The TA will grade using the mimi.cs.mcgill.ca server.

- All questions are graded proportionally (assuming the program runs at all). This means that if 40% of the question is correct, you will receive 40% of the grade.
- The TA will run the submitted script and observe if the directory and files have been created properly in their requested locations. Then, he/she will rerun it again in order to verify whether it handles the check for existing directory and files.
- The "information" file will be examined to verify whether it contains only the requested information (no more, no less). It will also be verified to check whether the file gets corrupted following multiple reruns of the same script.
- The TA will check the access rights of the "information" file. They need to be set as requested.
- The "log" file will be examined to verify whether it has the requested logging information in the proper order. It will also be verified to check whether the file gets corrupted following multiple reruns of the same script.
- The TA will finally check whether the script displays the "end of execution" message to the screen.