CS 2043 – ASSIGNMENT #4

Assigned: **2/15/2023**
Due: **2/21/2023, by 11:59PM**

NOTE: Please make sure you are present for Friday's lecture that covers *process substitution* before attempting to complete this assignment!

PROBLEM:

Having been impressed with your ability to conquer the automated email system bug (last week's assignment), the HR department is now *thoroughly* impressed with your abilities as a shell script engineer. They have now asked you to complete another task for them that is much more complicated!

As you will find out in the real world, sometimes a company chooses to evaluate the effectiveness of employees (especially programmers) with metrics that are flawed. Such is the case in your company. HR wants to be able to determine who is making the most commits into the company's git repositories and want to see any contributor's longest commit message. They feel such things are **key** in determining the effectiveness of each employee!

Despite your misgivings about the overall objectives of this task, you don't want to disappoint the HR department, so you set to work on the task. You'll need to write a script that does the following:

1. `cd` into the git repository (specified as an argument on the command line)
2. Use `git log` to search and search for the target user (identified by their email address, also supplied as a command line argument) to obtain the following information:
    a. How many commits has this user made?
    b. What is their longest commit message?
3. The command execution should look something like this:

```
$ ./gitMetricsForUser.sh rd29@cornell.edu ./A4repo

User rd29@cornell.edu has made 9 commits
Longest message:

This is a really long commit message because I know that HR is watching and
thinks that I am a REALLY good programmer if I have REALLY long commit
messages.   I really didn't do anything special in this commit except add A
LOT of comments because HR is also counting the lines of code I write
somewhere else, I just know it.   Thanks for reading my REALLY long commit
message and have a nice day!
```

The command should be called `gitMetricsForUser.sh` and take two parameters. The first is the email address of the user that we are going to target and the second is the directory in which the repository exists. You should be able to just `cd` into the directory that is passed in and then start using `git log` to get information. Your script should detect the following error conditions and print out the following messages for each (use the EXACT message below as we'll have grading scripts looking for them):

Error condition #1: Email address specified, but not a second argument containing the path to the git repository:

```
$ ./gitMetricsForUser.sh rd29@cornell.edu
usage: gitMetricsForUser username repository
```

Error condition #2: Email address specified, but the second argument is a directory that doesn't exist. We use "noSuchDirectory" as an example of a directory that does not exist below:

```
$ ./gitMetricsForUser.sh rd29@cornell.edu noSuchDirectory
gitMetricsForUser: noSuchDirectory: Not a directory
```

GETTING THE REPOSITORY

Download the A4repo.zip file from CMS or find it on ugclinux at IP address 132.236.91.187 in the /tmp directory.

NOTES:

This will be our hardest assignment of the semester. You will need to pull together a number of commands we've learned about and use some programming concepts such as loops and reading from files. The `grep` command can be used for part of this assignment and old friends such as `wc`, `for`, `read`, and `[]` operations will come in handy as well. Expect that you'll likely need to make the `git log` command at least twice in your script, once to find the number of commits for the user and then again to go through the log looking for the longest commit message.

As always, come to office hours and utilize Ed discussions for assistance.

IMPORTANT: If you do use Ed, please DO NOT post your script code that you have questions about publicly for all to see. If you want us to take a look at code you've written, please post it as private between yourself and the instructors.

IMPORTANT: You'll likely need to employ process substitution in your solution. This will be covered in Lecture #12 on Friday, 2/18!

You will only need to submit your shell script to CMS for this assignment; you won't be submitting a git repository as you did an A3.

FILES TO SUBMIT:

1. gitMetricsForUser.sh