CS232 Operating Systems
Assignment 01: Introduction to bash scripting
Due : 23h55 Wednesday 2nd October, 2019.

CS Program
Habib University

Fall 2019

1 Objectives

1. Test the student's bash command line and scripting skills.
2. Test the student's ability to follow written instructions.

2 Description

In this assignment you are supposed to submit a series of shell scripts.

You might find these links helpful:

1. Linux basics: https://ryanstutorials.net/linuxtutorial/#welcome
2. Bash scripts: https://ryanstutorials.net/bash-scripting-tutorial/
3. Google

3 Tasks

Q1. Make a script called cleanup.sh
This script would take two arguments:

1. a filename or a path to a file which contains rows of comma-separated data. The second column contains the student ID and the third column contains the student_name e.g.: 1., 3456, "student name", gpa, ...
2. a directory name of a path to a directory.

The script should exit with a helpful error message if:

(a) num arguments less than or greater than 2

(b) arg1 doesn't exist or is not a file or a file of size 0

(c) arg2 doesn't exist or is not a directory

The script should:

(a) read all the student IDs from the file and display their count.

(b) go to the directory specified by arg2 and see if there's a directory for each student in arg2. The directory names would follow the pattern stXXXX where XXXX is the student ID. In case the directory with a certain student ID is not found, it should display an error saying "student_names's directory was not found."

(c) for all the directories it finds, it should:

- delete all the temporary files (ending in .o, ~) in that directory. if there are subdirectories, it should recurse
- rename all the .txt files containing #/bin/bash as their first line as .sh (replace their extensions) recursively
- display the number of deleted and renamed files

Q2. Make the a script tictactoe.sh
This should simulate a 3x3 game between the computer and the user taking turns. The user should be able to specify their move via an x,y input (0-indexed). The computer will choose the coordinates for their move randomly.

Use the tput utility mentioned in the tutorials to display the game in the center of the screen.

At the end of the game the script should congratulate the winner and in the absence of a conclusion announce a draw.

Q3. Write a script test_passwd.sh
The script will check and validate passwords. The object is to flag "weak" or easily guessed password candidates.

A trial password will be input to the script as a command-line parameter. To be considered acceptable, a password must meet the following minimum qualifications:

- Minimum length of 8 characters
- Must contain at least one numeric character
- Must contain at least one of the following non-alphabetic characters: , #, $, %, &, *, +, -, =

You might find regular expressions useful here.

4 Submission Instructions

1. All scripts should be submitted as separate .sh files.
2. The scripts should be named Lec1_gp01_A1Q1_cleanup.sh:
   - this has fields separated by underscores (_)
   - the first field indicates your section number
   - the second field indicates your group number
   - the third field is a combination of assignment no. and question no.
   - the fourth field is the scriptname terminating in .sh
3. You'll also submit a PDF file combining all these scripts and the submitters' info. The name should follow the pattern Lec1_gp01_A1.pdf. A template will be provided later.
4. The final submission should combine all these file together and submit one .tar.gz file e.g. Lec1_gp01_A1.tar.gz. Explore the linux 'tar' command to archive and zip multiple files/directories together.

5 Rubric

files named correctly - 10 marks
PDF submitted correctly - 10 marks
Q1 working properly - 30 marks
Q2 working properly - 30 marks
Q3 working properly - 20 marks
Total - 100 marks