System Programming Lab Course Assignments

Assignment-1:

Problem Description:

You will be given variable number of TXT files having different types of ASCII sentences. A python program must be developed to accept all these file names as arguments and the output shall provide all the word counts in single output.

Sample Files are:

Output can be shown as:

{'am': 1, 'python': 1, 'afraid': 1, 'want': 1, 'learn': 1, 'of': 1, 'to': 1, 'I': 2, 'python.': 1, 'But': 1}

Instruction to be followed:

1. Testing procedure: <yourpythoncode.py> <file1> <file2> <file3>…<fileN>
2. The python binaries shall be 3.x
3. Files can be in your current directory or in different directory
4. You need to show the running time of your program in seconds
5. You can't use any external library to satisfy the requirement except basic python libraries.

Assignment-2:

Solve the Assignment-1 using shell script.

Instruction to be followed:

1. Testing procedure: ./<yourshellscriptname.sh> <file1> <file2> <file3>
2. Please use UNIX distribution as your operating system
3. Files can be in your current directory or in different directory
4. Please make sure you have assigned execute permission for your script
5. User shall get an error while running the script if the argument is not given correctly.

When you run your file, the output should as follow:

./yourshellscriptname.sh file1.txt file2.txt file3.txt

am 1
python 1
afraid 1
.... so on

Assignment-3:

Problem Description:

You have been given with a UNIX passwd file having all the user's details. Now you need to develop a shell script which will be taking out only duplicate usernames and their unique shell names as output from the given file. In addition, you need to take the passwd file as input in the shell script.

Sample Files are:


passwd

Output can be shown as:

Duplicate users are as follows:

adm

halt

games

[…continue printing usernames in same fashion above for more duplicate users]

Unique shell used among all the duplicate users above:

/bin/sh

/bin/bash

/bin/csh

[…continue printing shell names in same fashion above for more cases]

Instruction to be followed:

6. Testing procedure: <yourshellscriptname.sh> <passwdfile>
7. Please use UNIX distribution as your operating system
8. Files can be in your current directory or in different directory
9. Please make sure you have assigned execute permission for your script
10. User shall get an error while running the script if the argument is not given correctly.

Important Note:

- No other clarifications will be given except the above information related to the assignment.
- Use base python and shell scripting knowledge to solve the assignments.
- Please submit your files (into zipped format) within given timeline in the google class room