Lab 2 - Scripting

NERS/ENGR 570 Fall 2020
September 11th, 2020
DUE BY: September 22nd, 2020 -- 11:59 pm

Exercise 1 (30 pts) - Searching & Parsing

Download and unzip the `Ex1.tar.gz` file and enter the `Ex1` directory. In `Ex1`, you should see 5 folders labeled `Folder1, Folder2, Folder3, Folder4,` and `Folder5`.
In this exercise, you are required to create a shell script named `<user>_Lab2Ex1.sh`. It should contain all of the Linux commands needed to perform the following steps ((a)-(d)). Test your script and ensure that it does what you expect before submission.

(a) Use a command to download or copy the `Ex1.tar.gz` file from http://www-personal.umich.edu/~bkochuna/NERS570/Lab2/Ex1.tar.gz
(b) Use a command to unzip the `Ex1.tar.gz` file and enter the `Ex1` directory.
(c) Use the `find` command to find the locations of two files named `nuclear_secrets1.txt` and `nuclear_secrets2.txt`. You should type the 'find' command once for multiple files. Do not repeat it for each file. You should only search at most one level down the current directory (`Ex1`). For example, if there is a directory under `Folder2`, do not search it.
(d) Find the differences between the content of the two files and output this to a file named `launch_code.txt`.
(e) Perform the following steps:
    (i) Search all of the given files in the `Ex1` directory for the word "Jackpot" and send the output to `jackpot_locations.txt`.
    (ii) Repeat the search in (**i**) but only search all the `.doc` and `.xml` files and add the output to the end of `jackpot_locations.txt` (follow the output from (i)).
    (iii) Repeat the search in (**i**) without case sensitivity and for an arbitrary number of spaces between "Jack" and "pot". That is, it should find all instances of things such as "Jack pot", "Jack Pot" and "jackpot". This time **only** search all the `.txt` files. Add the output to the end of `jackpot_locations.txt` (follow the output from (ii)).

Note that your script should not overwrite any existing file content when adding output to the `jackpot_locations.txt`. When performing the searches in (ii) and (iii), your script should be able to bypass the `jackpot_locations.txt`, i.e. only search the files that we gave you in Ex1, do not search your output file.

Exercise 2 (70 pts) - Parametric Calculations and Function Approximation

Often in engineering simulation and analysis some material property is needed to provide the coefficients in your governing equation. Usually, properties of materials are represented by *semi-empirical* formulas or correlations. A semi-empirical formula is developed with two main ingredients. One is a partial theoretical understanding of the form of the equation describing the macroscopic property. The other ingredient is experimentally measured data. Given the generally accepted form of the equation the coefficients in this equation are determined by fitting the equation to the experimentally measured data, resulting in a semi-empirical formula.

Conversely, some scientific simulations turn this procedure "on its head" and do the opposite. In some fields of science and engineering, so called *ab initio* simulations are performed based on first principles to predict macro (or micro) scale properties of materials. Thus, it is necessary to run the same simulation repeatedly while varying one parameter (or sometimes more). This is often referred to as a *parametric study*.

In this exercise you will produce scripts in both bash and python that demonstrate common workflows of

1. Performing a parametric study
2. Fitting an equation to data

By developing scripts in both languages you will gain insight into the advantages and disadvantages of each language for each type of workflow.

For this exercise we are providing you with a simple program that calculates select thermophysical properties of water. The thermophysical properties of water are used in many fields of engineering and the primary reference for them is the International Association for the Properties of Water and Steam (www.iapws.org). This program represents your "experimental apparatus" to provide you with data.

The executable may be obtained here:
http://www-personal.umich.edu/~bkochuna/NERS570/Lab2/thermo_water

NOTE: THIS PROGRAM MUST BE RUN IN THE CAEN LINUX ENVIRONMENT
(otherwise it probably will not work, but it might. So are you feeling lucky?)

The program usage is:

```
$ ./thermo_water <temperature> <pressure> <property_name>
```

`<temperature>` is (273.15 K, 1000 K)
`<pressure>` is any real value on the interval (0,100] MPa
`<property_name>` is one of "`density`" g/ml, "`viscosity`" Pa*s, or "`enthalpy`" kJ/kg

The specific tasks for this lab exercise are:

(a) Write a script in bash that performs a parametric study. The script should

(i) Accept as an input the number of equi-spaced intervals in pressure at which the thermophysical properties of water will be evaluated.

(ii) Create three directories called "`300K`", "`600K`", "`800K`" for each temperature

(iii) In each directory create an individual file called "`water_prop_<N>.csv`" where `<N>` is the parameter input to the script for the number of intervals of pressure.

The content of the file should appear with a column for each property as:
`"pressure [MPa]", "<property_name> [<property_units>]", …`
`<pressure_value>,<property_value>, …`

(iv) Run your script to generate files evaluating the water properties with 10, 100, 1000, and 10000 intervals.

(b) Write a script in python that reads the files output in part (a) of this exercise and generates coefficients for a polynomial of a given order up to 8th order. The input should be the data file name and the polynomial order. The output should be a list of the coefficients, one coefficient on each line. You must implement your own linear least squares routine (python has an existing routine that does exactly this, but that's too easy, useful for verifying your implementation though! Feel free to use the NumPy linear algebra functions). The output should look like:

```
Density:
a0
a1

Viscosity:
a0
a1

Enthalpy:
a0
a1
```

(c) Consider and give written answers for the following:

(i) Provide an equation to estimate the maximum interpolation error of the viscosity of water at 600 K for linear and second order interpolation in terms of the number of intervals. Assume the exact expression for the viscosity is:

$$\nu(P) = 73.3 + 0.3785P + -0.001229P^2 + 2.949 \times 10^{-6}P^3 - 4.247 \times 10^{-9}P^4 + 3.12 \times 10^{-12}P^5 - 9.076 \times 10^{-16}P^6$$

(ii) Equi-spaced intervals are used to develop the polynomial coefficients in part (b). Is there a better or "smarter" way to develop a polynomial approximation by choosing different intervals?

(d) ***EXTRA-CREDIT*** (10 pts)
Write a script in python that performs the same operation as the script in part (a) of this exercise. In this script utilize the `argparse` package available in Python to handle the input arguments.

Deliverables and how to submit:

Please pack all of your files for Lab2 into a single tar(zip) file.
Name it as `<user>_Lab2.tgz` and submit it through canvas.
Your tar file for Lab2 should include:

- For Ex1: `<user>_Lab2Ex1.sh`.

- For Ex2: `<user>_Lab2Ex2a.sh`
  `<user>_Lab2Ex2b.py`
  `<user>_Lab2Ex2c.pdf`
  `<user>_Lab2Ex2d.py`

  For 2a, 2c, and 2d we will run your scripts in the CAEN linux environment and compare the output to the expected result.

Note: To create a zipped tar file for all files in a directory the command is

```
$ tar -cvzf <user>_Lab2.tgz ./*
```