ENGR 102 - Lab #9b

These activities are meant to help give you practice reading and writing files, as well as processing larger amounts of data, which is one of the common tasks that computer programs are written for.

Activity #1:

In a previous lab, your team designed and constructed a program for performing interpolation calculations. As a team, you should now work to build essentially the same program, extended somewhat (as described below) but in two different parts, one to write and store data in a file, and another to read data from the file and answer queries.

a) First, determine as a team how you will store the data in the file. You want to make sure you have agreement on how entries in the data file will be stored.
b) Create a program that will read in data and store it in a file. Your program should prompt the user for the name of the file to use, ask what is being interpolated (the independent and dependent variables, how many data entries there are, and then request values be entered (similar to the way the previous program worked) until the user is done. This data should be written to the specified file in the format your team chose.
    a. As an example, you might first ask for a file name,
    b. Then you might ask what the independent variable (i.e. the "x" value) is. The user might enter "time".
    c. Then, you might ask what the dependent variable (i.e. the "y" value) is. The user might enter "price".
    d. Then you would repeatedly ask the user to enter information (time and price) until done.

    All of that should be saved to the file in the format you describe.
c) Next, create a program that will prompt the user for a data file to read, and then will read in the data file, in the format given above. You should prompt the user for the file name. Then, repeatedly ask for values that the user wants to interpolate. You should print out each of the interpolated values.
    a. For example, following the example above, you would ask the user for a time, and then you would give the price at that time (interpolated or extrapolated from the data you read in from the file).

Submit your two programs, clearly indicating by their names which one is to collect data, and which is to interpolate data.

Activity #2:

Write a program that will take a file named Celsius.dat that contains a list of temperatures in Celsius (one per line), and will create a file Fahrenheit.dat that contains the same temperatures (one per line, in the same order) in Fahrenheit.

Activity #3:

One of the most common ways that data can be stored to be loaded in a spreadsheet or other similar table is with a CSV (Comma Separated Value) file. These often are given a .csv extension. A csv file is a

way of representing a table in a file. Each line represents a row of the table, and the cells in each column are separated by commas. CSV files can usually be read into spreadsheet programs (such as Excel), and most spreadsheets can output their data in a CSV format (sometimes called "comma delimited" format). You are going to practice writing and reading CSV data directly.

Write a program that will save, to a file, a list of amortized values for a loan. Specifically:

a. Ask the user for the amount of the loan, the annual interest rate, and the amount being paid monthly. Also, ask for the name of the file to write the results to. For whatever name is given, your program should add a ".csv" extension to the name, to indicate that it is a CSV file.
b. Each month, you should calculate the amount remaining on the loan by first applying the monthly payment (reducing the loan value), then increasing the loan amount by 1/12 of the annual interest rate.
c. For each month, write to the output file the month number, the total amount of interest accrued so far, and the amount remaining on the loan, separated by commas.
    i. Start with month 0, when there is no payment and no interest, with month 1 being the first payment and first interest accumulation
    ii. If the loan eventually will be paid off (i.e. if the loan amount is decreasing), write out values until the loan amount is 0
    iii. If it will not be paid off (i.e. the loan amount increases or stays the same each month), then write 30 months worth of data.
d. Be sure that you write out column headers for the table, indicating what each column is.

Note: if you write your .csv file correctly, you should be able to open it in a spreadsheet program that can read .csv files.

Activity #4:

On the class website is a CSV file containing weather data from Coulter Field (in Bryan) for 3 years (1 day is missing for some reason!); the data was taken from Weather Underground (wunderground.com). There are different versions of the file for Windows and Mac; the only difference is whether the end of a line contains just a new-line character, or both a new-line and a carriage return (you don't need to worry about the difference). Open the file in any text browser and you should see what it is. Note that the first line of the file contains the column headers explaining what each column is.

Download the file to your system, and write a program that will read in the file and do the following:

a. Output the maximum and minimum temperature seen over the 3 year period
b. Output the average daily precipitation
c. Pick any 3 other "interesting" data analysis questions (of your choice) and output the answer to that question. For at least one, make use of the date information. Here are some ideas, but you can pick whatever you want:
    a. For some particular day, such as December 25, find the maximum and minimum temperatures reached among the 3 years of data.
    b. For some particular month, such as July 2015, calculate the average high temperature.
    c. Calculate how frequently the pressure increases from one day to another vs. how frequently it decreases.
    d. Calculate the percentage of days when the humidity was above some value, like 90%.
    e. Calculate the mean and standard deviation of precipitation levels.

    Be sure to include a descriptive sentence for what you are printing out in each case.

Note that the “interesting” analysis questions you choose should be different analyses from each other; for example, you should not find the min/max temperature for just 3 different dates, or find the min/max pressure for December 25, since those are essentially the identical computation.