ENGR 102 - Lab Assignment #9

Spring 2019

General course learning outcomes:

- demonstrate programming techniques in the construction of computer programs, including:
  - collecting, creating, storing and manipulating data in larger structures such as lists,
  - using control structures such as conditionals and loops,
  - outputting processed results to a file, and
  - decomposing a complicated task into more manageable pieces.
- apply programming techniques to solve problems in engineering.
- complete a team programming assignment that ties together concepts learned in the class.

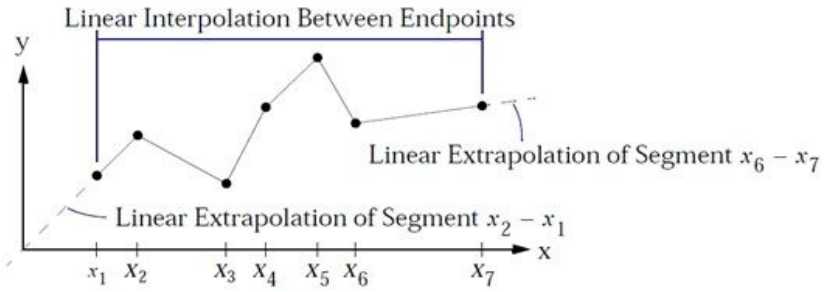
Activity 1: Top-Down Design of a Program - to do in lab (team)

☑ *Utilize the top-down design method in creating a Python program.*

Following the process described in the lecture, create a top-down design for a program. Then, construct code for the program.

Create a program that lets a user enter and arbitrary number of data points as x, y values. Assume the user will type only numbers or 'q' as an x-value to quit. Do not assume the user will input data in ascending order. Then calculate a linear interpolation/extrapolation from those points to provide the user with a value at any point a user requests. The user should be able to enter *any* x value and have a y value returned, such that the y value is the best estimate (using a linear estimation) from the *nearest* x value(s). Allow the user to repeatedly interpolate / extrapolate for new y-values until a 'q' is input.

Note: While a successfully implemented program will get you maximum points, this assignment is focused on the top-down design process. A correctly working program that was not created using the required process will not score as well as a well-designed process that has incomplete code or a poor plan.



http://aerospacengineering.net/wp-content/uploads/2014/04/figure_FR_8.jpg

Interpolation

For a value x in the interval (x_0, x_1), the value y along the straight line given by coordinates (x_0, y_0) and (x_1, y_1), is found from the equation:

$$y = y_0 + (x - x_0)\frac{y_1 - y_0}{x_1 - x_0} = \frac{y_0(x_1 - x) + y_1(x - x_0)}{x_1 - x_0},$$

Extrapolation

If the two data points nearest the point x to be extrapolated are (x_{k-1}, y_{k-1}) and (x_k, y_k), linear extrapolation gives the function:

$$y(x_*) = y_{k-1} + \frac{x_* - x_{k-1}}{x_k - x_{k-1}}(y_k - y_{k-1}).$$

(continued, next page)

A. Using a top-down design process, develop a hierarchy for portions of the program. Break each part into smaller pieces until the code for that portion of the program is "obvious". Record your work in a document to submit as a PDF.

B. Determine what variables you will use for the main sections of your code. You should decide on the main data that you will need to keep that will be used in more than one "node" of the design. Write a list of these, along with a brief description of what that variable will store. You do not need to decide on variables used only within one "node" of a program (e.g., a loop iterator if it is not going to be used outside the loop). Include this in your document.

C. Code the program. Translate your hierarchy into steps, and divide the coding tasks among all team members—each person will work on a different section of the program.

   i. Divide the program so each team member has approximately the same number of "leaf" nodes to implement.

   ii. A good top-down design and well-defined specifications for variables that carry over from one section to the next, will allow each member to write code for only their own section without seeing other sections of code!

   iii. Each person should write and execute tests that focus on *(only)* their own coding.

D. Once each person has written his or her own separate code, combine the code into one program. You will then need to test and debug your combined code together.

E. As a team, write a paragraph in your document describing the benefits and drawbacks your team encountered using this method. While combining your code, did your team find insight into how your design could have been improved?

Activity 2: Writing to a file - to do in lab (team)

☑ *Output processed results to a file*

Create a copy of your program from Activity 1, and modify it in the following way:

1. Before the user inputs x, y data, ask what the dependent variable values represent (e.g., Number of rabid elephants, kilograms of cheese consumed by llamas, quantity of squid ink in reservoir). Assume the x-values will be time.

2. Once the user has finished typing in x, y data points, write the following to an external file titled "nailedIt.txt":
   a. First line includes your team number and team names
   b. Second line includes the date you created the file
   c. Third line is blank
   d. Fourth line includes what the x-values represents (time), and what the y-values represent
   e. Fifth line contains a series of dashes (----------)
   f. Subsequent lines contain an x, y value pair. One x, y pair per line with a comma separating the values.
   g. After the last x, y value pair has printed, print another series of dashes

3. Your program should then begin asking the user to provide x-values for interpolation or extrapolation. After every new x value that the program interpolates or extrapolates, append an additional line to the file. Include first the new x, y value pair, and then print whether this was an interpolation or extrapolation.