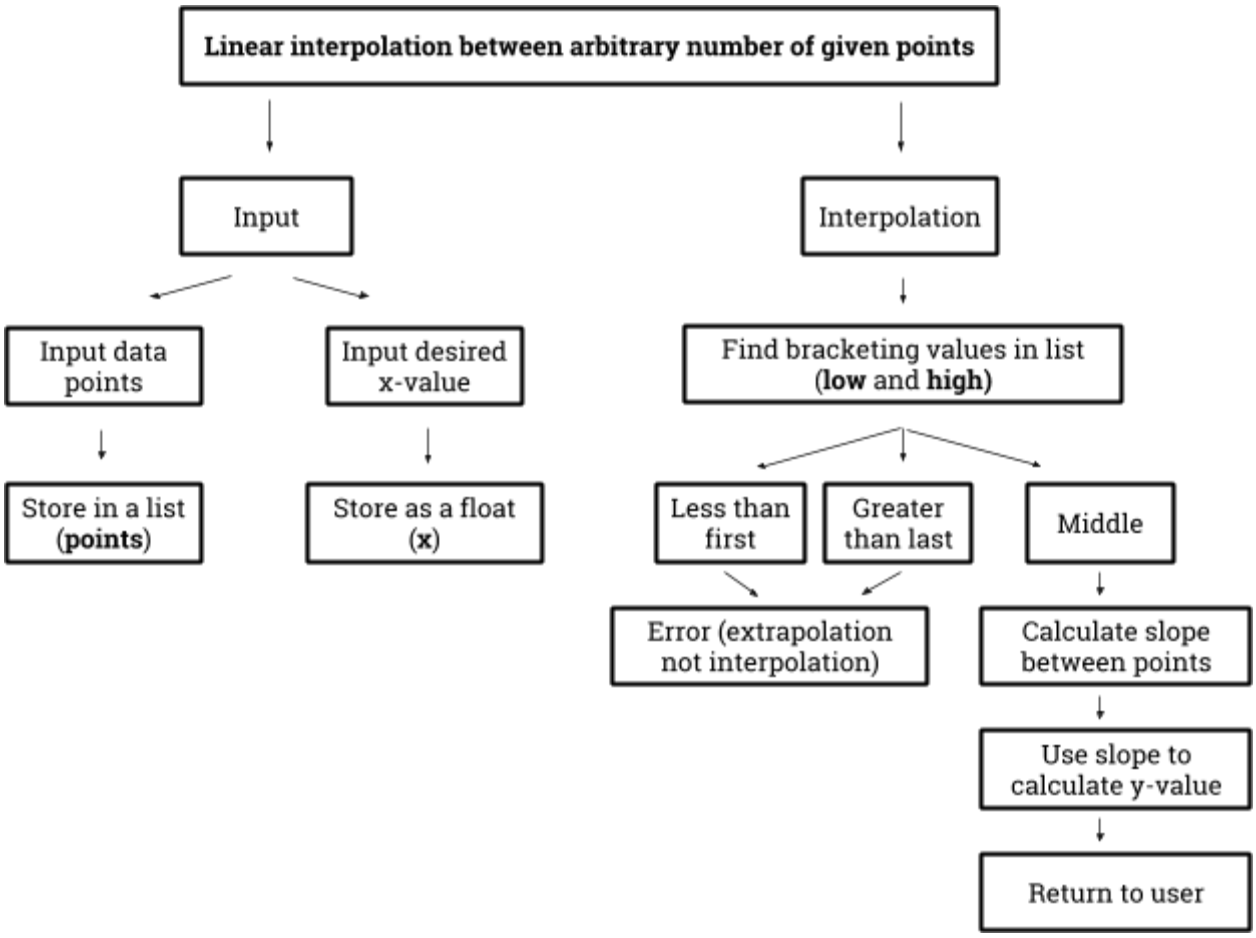
Muhammad Nauman
Daniel Huck
Patrick Zhong
Alexander Bogdan
ENGR 102-213

Lab 8.2: Top-Down Design of a Program

Top-down design:



Variables:

- **points** (list of arrays of floats): list of given x-y pairs
- **x** (float): given x-value to estimate the y-value at
- **low** (array of floats): point closest and under in x-value to **x**
- **high** (array of floats): point closest and over in x-value to **x**

Test cases:

- ([1, 2]), ([2, 6]), ([3, 1]), ([9, 8]), x = 2

    - Output: One of the given points was found at the desired x-value. The y-value there is exactly 6.
- ([1, 2]), ([3, 6]), ([4, 1]), ([8, 8]), x = 1.3
    - Output: The linear interpolation yields 2.6 as the estimated y-value at 1.3.
- ([1, 2]), ([3, 2]), ([5, 30]), ([8, 2]), x = 1.5
    - Output: The linear interpolation yields 2.0 as the estimated y-value at 1.5.
- ([1, 1]), ([2, 2]), ([4, 4]), ([5, 5]), x = 2
    - Output: One of the given points was found at the desired x-value. The y-value there is exactly 2.
- ([2, 1]), ([5, 2]), ([6, 4]), ([18, 5]), x = 2.2
    - Output: The linear interpolation yields 1.0666666666666667 as the estimated y-value at 2.2.
- ([2, 1]), ([5, 4]), ([6, 7]), ([7, 5]), x = 2
    - Output: One of the given points was found at the desired x-value. The y-value there is exactly 1.
- ([2, 1]), ([10, 9]), ([24, 7]), ([31, 8]), x = 5
    - Output: The linear interpolation yields 4.0 as the estimated y-value at 5.
- ([2, 1]), ([7, 9]), ([7.5, 7]), ([8, 8]), x = 5
    - Output: The linear interpolation yields 5.800000000000001 as the estimated y-value at 5.

Division of coding tasks:

- Taking points input
- Taking x-value input
- Finding bracketing values
- Using bracket to calculate y-value and return it

Summary:

Each person thinks differently so there were some slight merge conflicts and slight tweaks had to be made to each person's code. Dividing code like this speeds up development as people are working in parallel and not depending on each other, but also has a large chance of causing messes and conflicts when the division of coding tasks is not properly understood by the people working on them that take effort and time to resolve. For large projects with many meetings this would likely be very effective but forcing students to go home and work on segments by themselves they end up getting a fractured picture of what they were supposed to learn. If the project description was to change in the real world (e.g. finicky customer) small groups with few meeting times would probably also have a very hard time adjusting because lack of hierarchy to redistribute work and everyone would have already attempted to specialize in their original niche.