

Assignment #8

Repetition and Decomposition/Functions

Due: Sunday, 11/12/17, 11:59pm

(90 pts) **Problem Statement:**

Write a program, **my_math.py**, that defines 5 mathematical function and calculates the summation or integration of that function. You are creating your own library of commonly used mathematical functions and analyzing the different methods, rectangle vs. trapezoid methods, for solving integration problems to gain experience writing a short program, which uses selection, repetition, and functions/procedures.

Your program should define these 5 functions:

$$f1(x) = 2x^5 + x^3 - 10x + 2$$

$$f2(x) = 6x^2 - x + 10$$

$$f3(x) = 5x + 3$$

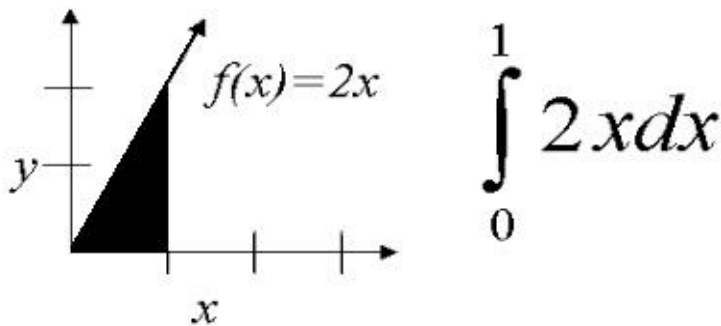
$$f4(x) = 2x^3 + 120$$

$$f5(x) = 2x^2$$

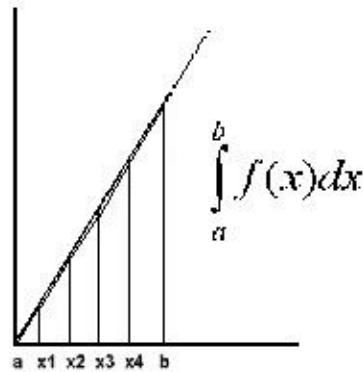
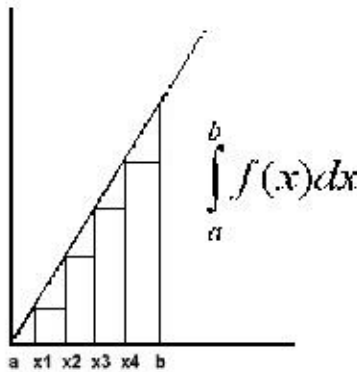
Integration Functions Explanation:

Your program should determine the area under the functions specified above. You will calculate the area under a curve using the rectangle and trapezoid method. Therefore, you need **integrate_rectangle()** and **integrate_trapezoid()** functions. **Don't freak out!!!** Programming integration into the computer doesn't consist of calculus!!!!!! It is simple math, and these two methods only require summing the area of a certain number of rectangles or trapezoids under a specific curve, $f(x)$.

For Example, if your $f(x) = 2x$:



Basic Numerical Methods (Rectangle vs. Trapezoid):



n = number of rectangles and/or trapezoids
a = beginning x value
b = boundary for rectangles
width = width of each rectangle/trapezoid, (b-a)/n
f(x) = height of rectangle
(f(x1)+f(x2))/2 = height of trapezoid
Area = width * height

The functions are bounded by any interval on the x-axis, including both positive and negative values!!!. You can check your answers to these functions using <http://www.wolframalpha.com/widgets/view.jsp?id=8ab70731b1553f17c11a3bbc87e0b605>

Here is an example run of your my_math.cpp program:

Choose a function (1, 2, 3, 4, 5, other(quit)): **5**
Would you like to calculate the area using the rectangle, trapezoid, or both (1, 2, 3): **2**
How many trapezoids do you want? **1000**
Please select a starting point, a = **1**
Please select an ending point, b = **2**
The area under **2x²** between 1 and 2 is 4.667

Program Input:

- Starting and ending points for the area or summation
- Function to calculate the area or summation, i.e. f1, f2, f3, f4, f5
- Function/Procedure(s) for calculating the area, i.e. rect, trap, both
- Number of rectangles and/or trapezoids to use

Program Output:

- The function being evaluated
- Starting and ending points for the integral or summation
- For integration: Number of rectangles and/or trapezoids used
- The area calculated by the method(s)

Program Description in more detail:

Your program needs to adhere to the following guidelines:

- If the user chooses to see the area calculated by both methods, each method should receive their own number of rectangles or trapezoids as input and return the value from the calculation.
- Your program should continue running until the user no longer wants to perform more calculations.
- You should use procedural decomposition and have functions for $f_1(x)$, $f_2(x)$, $f_3(x)$, $f_4(x)$, and $f_5(x)$, as well as functions for calculating the area using the rectangle vs. trapezoid method.
- All functions should be 10 lines or less, and you should not have any global variables.

(10 pts) Extra Credit:

Handle ALL erroneous input by the user associated with each of the options!!!!

(10 pts) Program Style/Comments

In your implementation, make sure that you include a program header in your program, in addition to proper indentation/spacing and other comments! Below is an example header to include. Make sure you review the style guidelines for this class, and begin trying to follow them, i.e. don't align everything on the left or put everything on one line!

http://classes.engr.oregonstate.edu/eecs/fall2017/cs160h-001/160_style_guideline.pdf

```
/******  
** Program: integration.py  
** Author: Your Name  
** Date: 11/12/2017  
** Description:  
** Input:  
** Output:  
*****/
```

Electronically submit your **Python program** (.py file, not your executable!!!) by the assignment due date, using TEACH.