Part 1 (50%)

Finish the function `num_integrate` by implementing the left and right rectangle rule and the trapezoidal rule of numerical integration. Only this function will be called by the automated grader. You are free to modify the rest of the code as you see fit, but remember to keep your work below the line `if __name__ == "__main__":`, since anything outside would be imported by the automated grader and could potentially mess it up.

Part 2 (50%)

In this part, we will conduct an analysis of the error margin of each numerical integration method.

1. (10%) Using Python, calculate the error incurred by estimating the area under the curve with a rectangle at each point. Mathematically, we define this error as $f(x) \cdot \Delta - \int_x^{x+\Delta} f(x)dx$. The function $f(x) = x^3 - 2x^2$ has been given as an example and will be used for the questions below. Plot the error margin of each rectangular approximation at x versus x.
2. (5%) Plot $\frac{d}{dx}f(x)$. Compare this plot to the previous plot and state your observation.
3. (10%) When the plots of f_1 and f_2 have the same shape, it means there exists a constant k such that $f_1(x) = kf_2(x)$. We call this the scaling constant. Find the scaling constant between the error plot and the derivative plot. Confirm that this constant is approximately $\frac{\Delta^2}{2}$. Does this observation hold when we use the right endpoint instead of left?
4. (10%) Provide a sketch proof of the result obtained above.
5. (15%) Repeat the experimental procedure above to determine the error formula for the trapezoidal rule and provide a sketch proof to justify the result. Hint: can we still use the first order derivative? Or is it something else this time?

Submission

Write your answers for Part 2 on a separate PDF file and submit it to Gradescope. The submission method for Part 1 is still being decided, and will be announced on Campuswire.