Numerical Integration Project: Riemann Sum Visualization

Course: CST-305

Institution: Grand Canyon University

Professor: Ricardo Citro

Programmer: Christian Nshuti Manzi

Date: June 19, 2025

# a. Responsibilities and Completed Tasks

This project was completed individually by Christian Nshuti Manzi. Responsibilities included:  
- Researching Riemann Sum concepts  
- Writing Python code to calculate and visualize numerical integration  
- Generating and analyzing plots for three different functions  
- Writing documentation and README files

# b. System Performance Context Description

The Riemann Sum Visualizer was developed and tested on a standard personal computer running Python 3.x.  
All computations were done using NumPy for numerical operations and Matplotlib for plotting, ensuring fast execution and accurate graphical output even with hundreds of plotted data points.  
Graphs rendered within 1–3 seconds each on an average CPU.

# c. Specific Problem Solved

The task was to develop a Python tool to approximate definite integrals of continuous functions using Riemann sums.   
The tool was required to display Left, Right, and Midpoint sum approximations for each function over a defined interval, providing graphical and numerical outputs.  
This assists in visual learning and approximation of integral values in calculus.

# d. Mathematical Approach

The Riemann Sum approximation method was applied using the formulas:  
Δx = (b - a)/n  
Left Sum: Sum[f(xᵢ)Δx] where xᵢ = a + iΔx  
Right Sum: xᵢ = a + (i+1)Δx  
Midpoint Sum: xᵢ = a + (i+0.5)Δx  
  
Functions evaluated:  
- f(x) = sin(x) + 1 on [-π, π]  
- f(x) = 3x + 2x² on [0, 1]  
- f(x) = ln(x) on [1, e]  
  
Each function was graphed using the three Riemann methods and compared with the true integral (where known).

Graph on lined paper with writing

AI-generated content may be incorrect. A graph on a piece of paper

AI-generated content may be incorrect.

A piece of paper with writing on it

AI-generated content may be incorrect. A paper with math equations

AI-generated content may be incorrect. A math equations on a piece of paper

AI-generated content may be incorrect.

A piece of paper with math equations

AI-generated content may be incorrect.

A piece of paper with math equations

AI-generated content may be incorrect.

Part 2: Download rates (in Mbps) were recorded over 30 seconds

|  |  |
| --- | --- |
| **Minute** | **R(t) Mbps** |
| 0 | 46.15 |
| 1 | 47.969 |
| 2 | 49.376 |
| 3 | 50.421 |
| 4 | 51.154 |
| 5 | 51.621 |
| 6 | 51.865 |
| 7 | 51.926 |
| 8 | 51.841 |
| 9 | 51.644 |
| 10 | 51.366 |
| 11 | 51.036 |
| 12 | 50.678 |
| 13 | 50.315 |
| 14 | 49.965 |
| 15 | 49.645 |
| 16 | 49.368 |
| 17 | 49.144 |
| 18 | 48.98 |
| 19 | 48.881 |
| 20 | 48.847 |
| 21 | 48.877 |
| 22 | 48.966 |
| 23 | 49.106 |
| 24 | 49.287 |
| 25 | 49.495 |
| 26 | 49.712 |
| 27 | 49.919 |
| 28 | 50.095 |
| 29 | 50.211 |
| 30 | 50.241 |

# A graph on a white sheet AI-generated content may be incorrect.

y = -0.0001x4 + 0.0094x3 - 0.2339x2 + 2.0441x + 46.15  
R² = 1

Total downloaded data over 30 minutes ≈ 90005.49 Megabits

Which is ≈ 11250.69 Megabytes

# e. Code Implementation Approach

The Python script follows these steps:  
1. Define the function f(x) and interval [a, b]  
2. Select number of subintervals (n)  
3. For each Riemann method (left, right, midpoint), compute sum using loop logic  
4. Plot rectangles and curve using Matplotlib  
5. Display Riemann sum value in the plot title  
  
Each method was modularized into reusable functions and executed via main function calls per task.

# f. Screenshots of Program Execution

Screenshots of graphs are included in the project directory as part of the output and submission files.

Each function generates 4 graphs:  
- Left Riemann  
- Right Riemann  
- Midpoint Riemann  
- Combined View

A graph with a blue line

AI-generated content may be incorrect.A graph with a blue line

AI-generated content may be incorrect.A graph with a blue line

AI-generated content may be incorrect.A graph with a blue line

AI-generated content may be incorrect.

# A graph with a blue line AI-generated content may be incorrect.A graph with a blue line AI-generated content may be incorrect.A graph with a blue line AI-generated content may be incorrect.A graph with a blue line AI-generated content may be incorrect.

# A graph with a blue line AI-generated content may be incorrect.A graph with a blue line AI-generated content may be incorrect.A graph with a blue line AI-generated content may be incorrect.A graph with a blue line AI-generated content may be incorrect.

# g. References

* Khan Academy. (n.d.). *L’Hôpital's rule and Riemann sums*. Retrieved June 19, 2025, from <https://www.khanacademy.org>
* Bronson, R. and Costa, G. (2022). *Schaum's outline of differential equations* (5th ed.). McGraw-Hill: New York, NY. ISBN-13: 9781264258826
* Stewart, J. (2015). *Calculus: Early transcendentals* (7th ed.). Cengage Learning.
* The NumPy Development Team. (2023). *NumPy documentation*. <https://numpy.org/doc/>
* The Matplotlib Development Team. (2023). *Matplotlib documentation*. <https://matplotlib.org/stable/contents.html>