

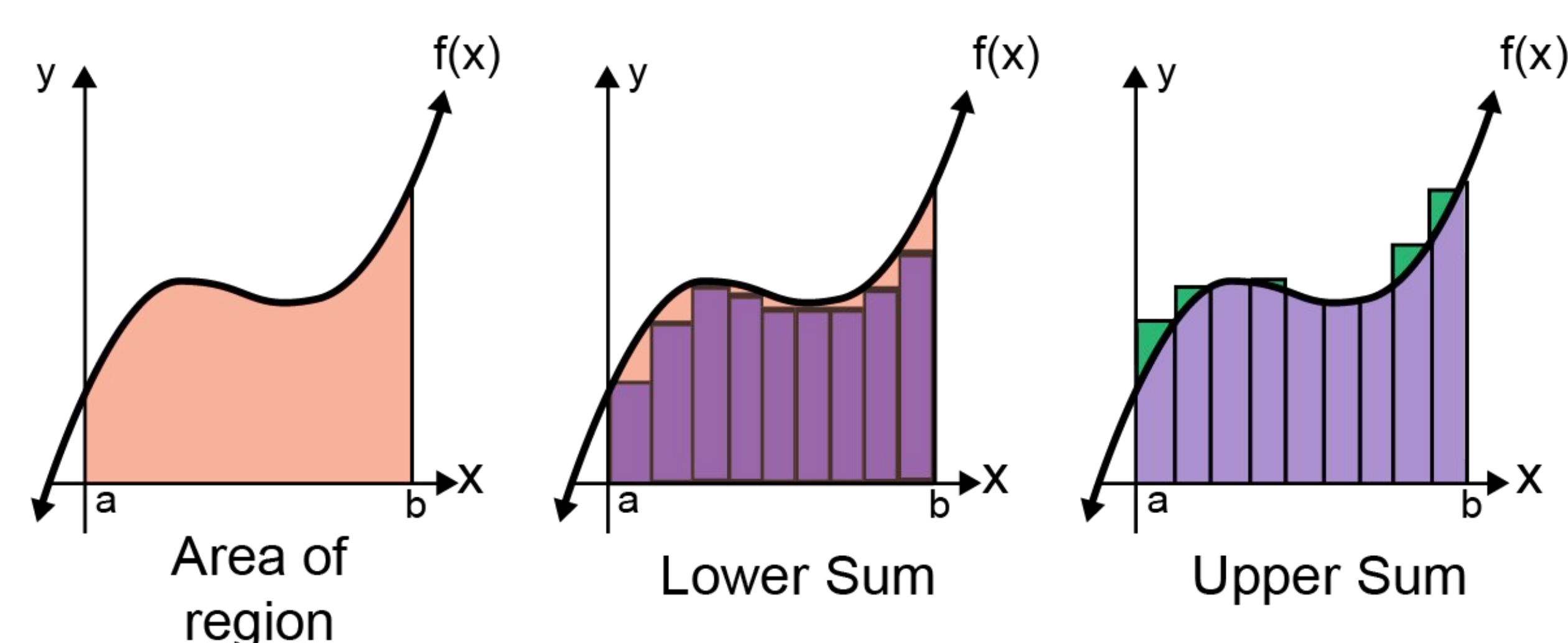
# Programming Riemann Sum Accuracy

Michael Liogky, Aaron Herrmann, & Jordan Bogdan

Choose **Ohio** First

## INTRODUCTION

While many people don't think of Calculus as something we use daily, for engineers, physicists, and other hard science majors, calculus is something they rely on daily on the job. We wanted to take the Riemann sum formula, that is commonly used to estimate the area under the curve of a function, and make a program that could give you this estimate quickly based on your desired degree of accuracy.



Calcworkshop.com

## Abstract

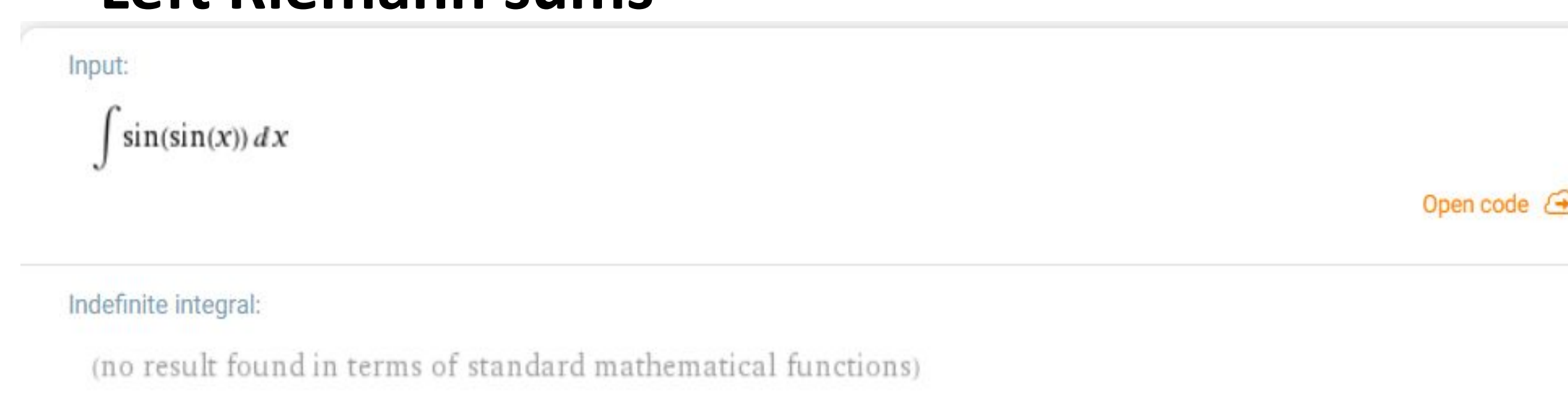
Taking the integral of an equation is a facet of Calculus people use every day on the job. When an equation cannot be integrated the Riemann Sum formula is then used to estimate the area of the graph. The problem with that though is when a job requires the graph to be accurate to a certain decimal degree. This project seeks to solve that problem by creating a computer program to calculate how many Riemann sum rectangles must be used to estimate the area of a graph to a certain decimal accuracy. The more rectangles that are used, the more accurate the estimation is. To do this a formula was created in Matlab so a person may plug in their equation values into certain specified variables in the formula program such as the formula itself, the two endpoints, and the desired degree of accuracy, to calculate how many Riemann sum rectangles are needed for the decimal accuracy they specified, and then calculate the final answer. This program should then help save time for engineers using the Riemann sum formula on the job.

## OBJECTIVES

In short we needed to make a program that took the variables  $f$  (the function),  $a$  (the start point),  $b$  (the endpoint), and  $n$  (the number of decimal places of accuracy), and then integrated the function into the Riemann sum formula from points  $a$  to  $b$  to the specified accuracy.

## METHODS

- Integrals and integration in calculus
- Right Riemann sums
- Left Riemann sums



## RESULTS

```
>> IntegralError(f,0,1,0.0001)
Integ =
0.4305
```

```
function r=rightsum(f,a,b,n)
dx=(b-a)/n;

% initialize r to f(a) (the right-end point of the first sub-interval)
% [a,a+dx]
r=f(a);

% need only consider the n-1 remaining sub-intervals
% 1. k=1:n represents where x is if we were to consider this being done
% by hand.
% 2. c=a+k*dx. a is the starting point. k*dx is the "progress." each
% new sum to be added to r is found with this.
% 3. r is the total value found so far.

for k=1:n
    c=a+k*dx;
    r=r+f(c);
end
r=r*dx;
```

- This code calculates the right hand Riemann Sum.
- The only difference between the right and left Riemann Sum code is that  $k=1:n-1$  as you exclude the first endpoint.

```
function Integ=IntegralError(f,a,b,err)
n=(f(b)-f(a)*(b-a))/(err);

% calculates amount of subdivisions needed for desired error

dx=(b-a)/n;

% calculates width of each riemann sum

r=f(a);

for k=1:n-1
    c=a+k*dx;
    r=r+f(c);
end
r=r*dx;
```

- For the program function one must just plug in the equation they have, the left and right bounds, and the decimal error rate that is required.

## CONCLUSIONS

- A person may know to a certain decimal point accuracy the area of the increasing function using Riemann Sums calculated by the Matlab program function by entering the required values.
- For now the program function can only be used on graphs that are continuously increasing. For future research the Matlab program will be steadily modified to be able to estimate the area of all the different types of equations and graphs.

## References

- Dr. Steven Gubkin. Mathematics Professor at CSU.
- "Makers of MATLAB and Simulink." *MathWorks*, [www.mathworks.com/](http://www.mathworks.com/).

## Acknowledgments

- Many thanks given to Dr. Steven Gubkin for helping guide the research and always giving a helping hand when walls were hit in the research.