# MATH-151 Lab 3: Functions and Recursion

**Due:** Monday, September 18, 2023, 10:00am

---

Please perform the following tasks using Matlab, submitting all relevant code. You are welcomed to work with other students, however each student must submit their own unique code.

## Task 1: Return of the Factorials

We looked at the factorial function previously to practice using loops in Lab 1. The same property to compute the factorial using a loop also makes it very well-suited to be used as a recursive function! As a reminder, the property of interest is below

$$n! = n \times (n-1)!, \text{ with base case } 0! = 1$$

a) Using this property, create a recursive function `recursive_factorial(n)` that computes the factorial for positive integer `n`. Using this function, what is 16! ?

## Task 2: What's Your Sine?

Anyone who has taken a class with me should be familiar that I believe the Taylor series is one of the most important concepts in mathematics. In this task I will try to drive that concept home by showing you how it is used to approximate the Sine function. Reminder, the Taylor series for Sine using $N$ terms is given as

$$\sin(x) \approx \sum_{n=1}^{N} \frac{(-1)^{n-1} x^{2n-1}}{(2n-1)!}$$

a) Guess how many terms of the Taylor series Matlab uses for the `sin(x)` function.

b) Create a function `taylor_sine(x,N)` that accepts a vector `x` and an integer `N` and outputs a vector containing estimates of the Sine of each element of `x` using `N` terms of the Taylor series. Using `x = -pi:0.1:pi`, plot the outputs of `sin(x)` and `taylor_sine(x,3)` as functions of `x` on the same axis. How well do they agree with each other? *(Note: I would prefer if you use your factorial function from Task 1, but you may also use Matlab's built-in* `factorial` *function.)*

c) Now try using `taylor_sine(x,5)`, how does that look?

*It can be helpful to test your functions with small, known examples first!*