Part 1: Factorial Function

The objective of Part 1 is to write a function that computes the factorial of an integer greater than or equal to 0. Recall that the factorial of a number n is n * (n-1) * (n - 2) * … * 1. The factorial of 0 is defined to be 1. Before taking on this part of the assignment, you may want to review the section on Functional Programming in this course (you can also read that section here).

For this Part you will need to write four different versions of the Factorial function:

1. Factorial_loop: a version that computes the factorial of an integer using looping (such as a for loop)
2. Factorial_reduce: a version that computes the factorial using the reduce() function in the purrr package. Alternatively, you can use the Reduce() function in the base package.
3. Factorial_func: a version that uses recursion to compute the factorial.
4. Factorial_mem: a version that uses memoization to compute the factorial.

After writing your four versions of the Factorial function, use the **microbenchmark** package to time the operation of these functions and provide a summary of their performance. In addition to timing your functions for specific inputs, make sure to show a range of inputs in order to demonstrate the timing of each function for larger inputs.

In order to submit this assignment, please prepare two files:

1. factorial_code.R: an **R script file** that contains the code implementing your classes, methods, and generics for the longitudinal dataset.

2. factorial_output.txt: a **text file** that contains the results of your comparison of the four different implementations.