

Peter Julius M. Estacio
Grouped with Nino de Mesa and Osh Ong
ENGG 27.01
August 21, 2025

First Progress Report on Project 1: Smallest and Largest Numbers

This project focuses on a program that accurately calculates for the largest and smallest positive number that can be represented using the `double` data type in C++. Specifically, the program is limited to basic arithmetic operations, the constants 1, 2, and 0, and the `isinf()` function from the `std` library.

Basis for the program's accuracy is the IEEE 754 standard for floating-point numbers, as described in the linked Wikipedia article [1]. Specifically, the `double` is represented with 64 bits; 1 bit for the number's sign, 11 bits for the number's exponent, and 52 bits for the number's significand. By multiplying the number's sign, significand, and exponent, the number can accurately be represented. Therefore, the significand can reach up to 2^{52} while the exponent can reach up to 2^{1023} due to the exponent biasing of the `double` [1].

With this in mind, major progress has been achieved for the program itself. The `computeLargestNumber()` and `computeSmallestNumber()` functions have been initially programmed with initial results. The smallest number was found by iteratively dividing by 2 until just before the number equalled 0, as the smallest `double` is an exponent of 2 [1]. The largest number was found by iteratively multiplying the exponent by 2 just before it hit `inf`. Then, the significand was divided by 2 until just before it hit 0, then subtracted from 2.

While results have been verified, further progress includes more rigorous ways of verifying results as well as neater written code. Shown below are the results displayed by the program in the terminal:

```
The largest positive number stored in a double is: 1.79769e+308
The smallest positive number stored in a double is: 4.94066e-324
```

Sources:

- [1] Contributors to Wikimedia projects. “Double-precision floating-point format - Wikipedia.” Wikipedia, the free encyclopedia. Accessed: Aug. 21, 2025. [Online]. Available: https://en.wikipedia.org/wiki/Double-precision_floating-point_format