Math 4610 Fundamentals of Computational Mathematics - Lecture 2.

In this lesson, you will be introduced to simple algorithms for determining "machine epsilon" to quantify computer number precision. The output of these algorithms serve as a warning about using computers in the solution of real mathematical problems. In adition to finite precision in the representation of numbers on a computer, most approximation algorithms will require large numbers arithmetic operations to determine an appropriate approximation.
Representation of round off error and the analysis of how errors accumulate in complex computations. One way to proceed is through interval analysis applied to numbers that are approximated in machine arithmetic. A simple example of how to build a shared library will also be discussed. This is an important skill to have in computational mathematics.

Content Items:

- **Building a Repository on Github:** - We will discuss how to create a general repository to start class. The idea is that students should create the math4610 repository in this part of the lecture. > > go there (pdf)
- **Homework repositories on Github:** - We will discuss how to create a homework repository that each student must have to complete homework. The homework repository should be a private repository with you instructor as the only collaborator. Homework will be graded by first cloning the repository and then your constructor can create a branch for grading. This can be repeated by creating a pull request on the repository. > > go there (pdf)
- **Practical Intuition about Machine Precision:** Some simple examples of how computers represent numbers are given that indicate that almost all of the work done by computers is an approximation. Finite representation in a binary number system has significant limitations. Since this is what we have to use, it is important to understand the implications of finite number representation. > > go there (pdf)
- **Code for Determining Machine Precision:** A simple algorithm for determining the precision for number representation will be covered. Two different versions of the code will be presented that determine single and double precision for your computer. Note that these two codes will be used to show students how to build a shared library. > > go there (pdf)
- **Wrap up and Questions:** If there is time and anyone has questions about the lecture, these will be addressed.