

MATH 417: Numerical Analysis

Lecturers: Prof. Wolfgang Bangerth, Prof. Guido Kanschat
bangerth@math.tamu.edu,
kanschat@math.tamu.edu

Teaching Assistants: Seungil Kim, Yan Li
sgkim@math.tamu.edu,
yli@math.tamu.edu

Lab assignment for the week 08/28–09/01/2006

All problems are discussed in the lab on Wednesday/Thursday. You do not need to hand anything in.

Problem 1 (Continuous vs. discrete). Functions $f(x)$ are usually defined over an entire domain $x \in I = (a, b) \subset \mathbb{R}$ and – if interesting – take values in an image $f(I) \subset \mathbb{R}$. Both domain and image are sets with infinitely many elements. On the other hand, computers can only represent numbers using a finite number of bits, most often as 32-bit (`float`, or `REAL*4`) or 64-bit (`double`, or `REAL*8`) IEEE floating point numbers, which store numbers in the form $\pm m \cdot 2^e$, where $0 \leq m < 1$ is the mantissa

$$m = b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \dots + b_M 2^{-M} \quad (1)$$

and e is the exponent and has the form

$$e = \pm(u_1 2^1 + u_2 2^2 + u_3 2^3 + \dots + u_E 2^E). \quad (2)$$

The coefficients b_i, e_i are single-bit numbers, i.e. either 0 or 1. In the binary system, floating point numbers can therefore be written as $\pm 0.b_1 b_2 b_3 \dots \times 2^{\pm u_1 u_2 u_3 \dots u_E}$. The total number of bits needed for the representation are M bits for the mantissa, $E + 1$ bits for the exponent, and 2 bits for the two signs.

Obviously, not all elements of I and $f(I)$ can be represented. Write a short program to find

- an approximation to the smallest and largest positive numbers that can be represented in float and double precision;
- the smallest float and double floating point number you can add to 1 such that the result is different from 1.
- In exact arithmetic, the system of linear equations

$$\begin{aligned} x_1 + x_2 &= 2, \\ x_1 + 10^{20} x_2 &= 1 + 10^{20} \end{aligned}$$

has the solution $x_1 = x_2 = 1$. Are there corresponding floating point numbers for x_1, x_2 that when plugged into the left hand side of the equations yields the exact values on the right hand side? If so, which? If not, is this a problem?

Problem 2 (Floating point vs real numbers). Let ε be the smallest floating point number in double precision such that in computer arithmetic $1 + \varepsilon \neq 1$ (you determined ε in Problem 1b). What are the floating point values of $(1 + \frac{\varepsilon}{2}) - 1$, $1 + (\frac{\varepsilon}{2} - 1)$, and $(1 - 1) + \frac{\varepsilon}{2}$? In what important way do exact and floating point arithmetic therefore differ?