Numerical Practical Training, UKNum
WS 2023/2024 (Block Course Feb. 19 - Mar. 1st, 2024)
Exercise 1 (Feb. 19th)Prof. Dr. Hubert Klahr
Numerical Representation of Numbers
Return by 9:15 a.m. Feb. 20th
as `.pdf` by Mail to: muley@mpia.de

- Make yourself acquainted with your computer desktop (Unix environment). Use the Unix commands `ls`, `df`, `ps`, test the use of an editor of your choice to write small programs or texts (e.g. `vi, emacs, joe, nano,` ...).
- Check how you can produce plots, e.g. using the `gnuplot` program, `matplotlib` or any other software of your choice.

Your code should be in a programming language of your choice (support can only be offered for Python, Fortran or C, C++).
Ensure readable and organized code:

- using naming conventions for variables;
- placing whitespaces, indentations and tabs within code;
- adding comments throughout to aid in interpretation.

Assignment for the Afternoon / Homework

- **Exercise 1, 6 points**: Round-off Errors
  Convert the decimal number $(-0.004831)_{10}$ into a binary format used for the hypothetical ten-bit word presented in the lecture. Compute the true error and the relative true error (absolute values) made by the ten-bit representation of $(-0.004831)_{10}$. (No programming necessary.)
- **Exercise 2, 6 points**: Truncation Errors
  Calculate the value of $e^{1.5}$ using the Taylor series of e^x. Increase the number of terms used in the Taylor series until the relative approximate error (absolute value) is less than 0.1 %. Document the results in a table, the code in a printout. Do this for at least two different machine precisions, e.g. Python: `numpy.float32, numpy.float64, numpy.single, numpy.double` or C++: `float a; double d;` for comparison.
- **Exercise 3, 8 points**: Machine ε
  Solve the quadratic equation $x^2 + x + c = 0$ directly using the quadrature $x_1 = (-1 + \sqrt{1-4c})/2$, for $0 \leq c \leq 1/4$. Prepare a computer program, which outputs x_1 as a function of c. What is the smallest c which produces a correct solution for $x1 \neq 0$? Hint $c_{init} = 0.25$ then $c_{new} \leftarrow c_{old} \times 0.5$. Does $\times 0.9$ make a difference? Relate this to the machine ε for single precision. How can you obtain a more reliable result even numerically for small c by rewriting the quadrature expression? Please print this for two different machine precisions as well.