Computer Science 206 – Scientific Computing (Wayne Hayes)

Assignment #1: Machine Representation of real numbers

Write a program, in the language of your choice, to determine the following values on the openlab machines: the number of bits in the mantissa (aka the "significand"), the machine epsilon, and the smallest and largest possible represented values. **Your answers must be correct to all digits for full marks.** Do it for the following floating point types: single precision, double precision, Intel extended precision (if you're using Python, I believe the data types are called float32, float64, and longdouble/float128, respectively—but I'm not a Python guru so you'll need to verify this yourself). **Bonus:** if your language supports it, also do this for quadruple precision. You may assume the machine uses base-2 (finding the base without looking inside the representation is hard), but you may *not* use any library routines that automatically compute these values for you. You are also not allowed to look "inside" the floating-point number at the bits. The floating-point representation is to be treated as a "black box" that you know nothing about and cannot see inside of. Use only floating-point operations (add, subtract, multiply, divide, and comparisons). Present the results in a table that has as many lines as there are types, and order the lines from least to most accurate representation. Your output table must have ***exactly 7 columns, with no header line, separated by TABS (not spaces, because the name can include spaces)***: (1) language (repeated for each row), (2) name of datatype, (3) size of the datatype in bytes, (4) number of mantissa bits determined by your program, (5) machine epsilon determined by your program, and finally (6) smallest and (7) largest representable positive numbers. All the latter values should be printed using the normal output routine (eg, 10^{-7} is printed 1e-7 or 1E-07 or something very similar) as many figures of precision as make sense. If you are using C or C++ and using "printf" to output, then note that "long double" must be output as "%llf". There is a script on openlab called /home/cs206p/bin/a1syntax that will check that your output satisfies the I/O spec.

If you fail to satisfy the output spec as specified (and checked) in the above script, we reserve the right to apply a penalty. Software specs are important, even in numerical computing!