EP501: Numerical Methods for Engineers and Scientists

Fall 2016, J. B. Snively

ERAU Daytona Beach

Homework Assignment Project #1, Due: Tuesday, 9/20.

For this assignment... Understand the most basic principles (and limitations) of floating point arithmetic and numerical linear algebra.

Submission Instructions: Submit in class, on paper, your work done by hand and printouts of the results with discussion ("Publish", in as few pages as possible).
Email your .m file to **snivelyj@erau.edu**, with the subject "EP501: HW1 Last Name, First Name".

1) For a hypothetical 8-bit floating point number system, with E=3 and M=4, **explain** (in a sentence or two) and **obtain** each of the following:

- **a.** The range of exponents *e* before a bias is applied.
- **b.** An appropriate bias *b*.
- **c.** The range of exponents *e* after a bias is applied.
- **d.** The range of positive values for the fraction *f*.
- **e.** The effective machine precision *eps*.
- **f.** The smallest (nonzero) and largest normal number (binary form and decimal).
- **g.** The smallest (nonzero) and largest denormal numbers (binary form and decimal).
- **h.** The binary representation of the number closest to the day of the month on which you were born divided by 2.

2) Demonstrate your comfort with matrix multiplication:

- **a.** Define random 3x3 and 3x2 matrices **A** and **B**, where each element is a number between 1 and 10. Multiply them by hand.

*** Note, the following should be coded and commented in Matlab, so that results and all discussion can be "Published" to an output file — this will simplify your reporting!*

- **b.** Code in Matlab the algorithm described in the text, page 25, using 3 nested loops to multiply any *m* x *n* matrix **A** by an *n* x *p* matrix **B**, using your matrices above.
- **c.** Code (in same file) the algorithm using 2 nested loops, leveraging " : ".
- **d.** Code (in same file) the algorithm using 1 loop, also leveraging ":".
- **e.** Demonstrate consistency with the " * " command. (The easy part.)
- **f.** Using the Matlab "cputime" to time each method (3 loop, 2 loop, 1 loop, *) when using *n* x *n* matrices **A** and **B** that are defined by the "magic" command (type "help magic" to understand what this is), where *n* is large enough that the overall sequence will take a few seconds (hundreds of elements).
  *** Note: Do not output the full contents of the magic matrix or results for this part, but do show the results of timing for each loop approach.*
- **g.** Write a few sentences about what you observe and, if you can, try to explain any surprising results.

3) Do the following textbook problems from Chapter 1: 13, 21, 29, showing all steps. Before starting, first write the system in matrix form. Then, for each solution approach, time yourself from start to finish. Explain your findings in a few sentences, comparing to predicted estimates of computational cost for these algorithms.