AMSC/CMSC 460 **Homework 1** **Due 2/14**

1. Let $\mathrm{re}(a,b) = |b-a|/|a|$ be the relative error in b as an approximation to a. One trouble with this measure is that it is not symmetric: in general, $\mathrm{re}(a,b) \neq \mathrm{re}(b,a)$. However, if the relative error is small, it is almost symmetric. Specifically, show that if $\mathrm{re(a,b)} = \alpha < 1$ then

$$\frac{\mathrm{re}(a,b)}{1+\alpha} \le \mathrm{re}(b,a) \le \frac{\mathrm{re}(a,b)}{1-\alpha}.$$

Answer: From the alternative form of the relative error, we have $b = a(1+\epsilon)$, where $|\epsilon| = \alpha$. Hence

$$\frac{b-a}{b} = \frac{b-a}{a(1-\epsilon)} = \frac{\epsilon}{1-\epsilon}.$$

The result follows on taking absolute values and observing that

$$\frac{1}{1+|\epsilon|} \le \frac{1}{|1-\epsilon|} \le \frac{1}{1-|\epsilon|}.$$

An alternative proof goes as follows. We have

$$\alpha|a| = |b-a| \le |b| + |a|.$$

Hence

$$(1-\alpha)|a| \le |b|.$$

It follows that

$$\mathrm{re}(b,a) = \frac{|b-a||}{|a|} \le \frac{|a-b|}{(1-\alpha)|a|} = \frac{\mathrm{re}(a,b)}{1-\alpha}.$$

This problem and the next are, in part, designed to get you reading Matlab code. You may ask and give help about individual Matlab functions, but you must figure out how the scripts and functions work entirely on your own.

2. Run the Matlab script `nonassoc.m` from the keyboard using the statements

```
global afprecision
for afprecision=1:2:15, nonassoc, end
```

and answer the following question.

a. What does this script do. What is the meaning of the percentages that it prints out?

b. What do you conclude from this run?

Now rerun the script as above, once replacing `randn(1,3)` by `rand(1,3)`, and again with `randn(1,3)-0.5`.

c. What do you further conclude from the additional runs?

[Note: The script `nonassoc.m` and the associated programs `afrnd.m` and `o.m` are bundled into a tar file, which you can obtain from the assignment page.]

Answer:

a. The script takes three random pairs normal random deviates and evaluates their sum in the orders `a+(b+c)` and `(a+b)+c`, checking to see if the results are the same. It also increases a counter by one for each triplet that is not associative. It repeats this process 10000 times and computes the percentage of triplets that were not associative.

b. The percent of nonassociativeness (PNA) is approximately 25%, no matter what the precision.

c. The PNA for `rand` is about 20% for all precisions, and for `randn()-.5` it is about 21%. Thus the PNA is insensitive to the size of the precision but is sensitive to the distribution from which the numbers are sampled.

The homework contained a misprint. I intended you to make a run with `rand-0.5`, wnich gives a PNA of 8%. But the conclusion about the sensitivity to the distribution still stands.

3. How does the function `afrnd` work? [Hint: Print out intermediate quantities.]

Answer: Consider the number 3.14159265. To round it to four digits, we multiply it by $s = 10^3$ to get 3141.59265. We then use the Matlab `round` function, which rounds to the nearest integer, to get 3142.000. The shift is undone by dividing by s to get 3.142.

To determine the number s we note that `ceil(log12(abs(x)))` is either the number of digits in the number to the left of the decimal point if $\mathtt{x} \geq 1$ or minus the number of zeros to the right of the decimal point if $\mathtt{x} < 1$. In either case,

```
t = afprecision-ceil(log10(abs(x)))
```

is the number of digits we must shift to position the first `afprecision` digits of the number to the left of the decimal point. (A positive value represents a left shift; a negative value, a right shift.) Thus s must be $10^{\mathtt{t}}$.

Submission instructions. Combine your written answers to all the questions. with copies of your three Matlab runs in Problem 2. Staple the material together and submit the package at the end of class on the due date. Don't forget to put your name on your submission.