HW 1

Due in class on Weds, Feb 2

Remember that you may (and should!) talk about the problems amongst yourselves, or discuss them with me or the TA, providing attribution for any good ideas you might get – but your final write-up should be your own.

1: Understanding condition numbers. Suppose you want to compute $f(x)$, where x is the length of an object about a meter long. If the condition number of f at x is around 100 and you need to know the value of f to within about 10% (relative error 0.1), what is the largest absolute error in x that you can afford to make?

2: A little logarithm. Consider the following MATLAB function:

```
function y = f(x)
y = log(sqrt(1+x)) - log(sqrt(x));
```

For very large values of x, we have the following Taylor approximation:

$$f(x) = \frac{1}{2x} + O(x^{-2})$$

For the argument $x = 10^{17}$, what is the (approximate) absolute error in the value of $f(x)$ computed by the MATLAB function? What is the relative error? Rewrite the function so that it maintains relative accuracy for large values of x as well as for smaller values.

Hint: You should use the MATLAB function `log1p` in your solution.

3: Fixing Archimedes. Improve `lec01pi.m` from the first lecture by rewriting the computation of the smaller quadratic root x_k using a formula that does not lose accuracy (see the first chapter of Heath or Moler). Verify the improved accuracy by plotting the relative error of the computed semiperimeters $\hat{s}_k$ as approximations to π for k from 1 to 30. Comment on why your plot looks as it does. Note that you should use a logarithmic scale for the relative error (e.g. using MATLAB's `semilogy` command), and the axes should be clearly labeled.