Written Homework #1

Due by the end of class lecture on Thursday, September 8th. Please write your answers clearly and show all work. You will be required to have a printed out version of code as well as submit a file containing code for all of the homework problems to be submitted on myWPI.

The preferable programming language is MATLAB, although assignments in other languages are accepted. You can use MATLAB for free at a computer lab on campus or you can Remote Desktop to log into windows.wpi.edu. Instructions for connecting to the Terminal server can be found at

`http://wiki.wpi.edu/ITS/Terminal_Server`

If you want to use the linux compute machine, you can SSH into Rambo.wpi.edu, or fourbanger.wpi.edu. For example, I type:

```
ssh -X sdolson@Rambo.wpi.edu
```

and after entering my password, I just type matlab and matlab pops up. You could also use GNU Octave which is free and pretty much the same as MATLAB. For info on GNU Octave, go to

`https://www.gnu.org/software/octave/`

1. Prove that any number 2^{-N}, where N is a positive integer, can be represented as a decimal number that has N digits, that is $2^{-N} = 0.d_1 d_2 d_3 \cdots d_N$

2. Derive a bound on the truncation error in the approximation

$$\frac{1}{1-x} \approx \sum_{j=0}^{10} x^j$$

   that is valid for $-1/2 \leq x \leq 1/2$.

3. In a floating point number system, there is an important number called machine epsilon (a.k.a. unit roundoff, unit rounding error, etc.). This is defined to be the smallest floating point number which, when added to one, results in a floating point number greater than one. Write a short MATLAB code to determine machine epsilon. Later, compare this answer to typing in "eps" at the command prompt.

   Note: Please hand in m-file and hand write the answer from your MATLAB code and the answer for the built in machine epsilon eps.

4. The harmonic series $\sum_{n=1}^{\infty} \frac{1}{n}$ diverges. Write a program that calculates the sums S_1 and S_2:

$$S_1 = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{1835419} + \frac{1}{1835420}$$

$$S_2 = \frac{1}{1835420} + \frac{1}{1835419} + \cdots + \frac{1}{3} + \frac{1}{2} + 1$$

What is $|S_1 - S_2|$? Why do you get different answers for S_1 and S_2? Which of the two is a more exact approximation?
Hint: You can start a for loop with: `for i=1:183540` for S_1 and `for i=1835420:-1:1` for S_2, which will count down by one each time through the loop. At the top of your program, put in the line `format long`.
Note: Please hand in the m-file and type or hand write in the program the value of S_1, S_2, $|S_1 - S_2|$, and your explanation.

5. From a calculus course, you learned that a derivative can be approximated by:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

As you might guess, there is a balance between trying to minimize round off error as well as other types of errors when approximating derivatives (more on this in another class). For the following function:

$$f(x) = x^2 \ln(x)$$

We want to numerically approximate the derivative of the function at the point $x = 2$, i.e. we want to determine $f'(2)$ numerically using the simple first order forward difference scheme,

$$f'(x) = \frac{f(x+h) - f(x)}{h}$$

   (a) Write a code to find the optimal h (the h for which error is smallest) by numerical experimentation.
   (b) Create a graph that shows how error decreases and then starts to increase as h continues to decrease.