CS 594 - Spring 2003
Understanding Parallel Architectures: From Theory to Practice
Assignment 3
Due: February 19, 2003

The accuracy of a floating point system can be characterized by a quantity variously known as the *unit roundoff, machine precision, machine epsilon,* or *macheps.* Its value, which we denote by ϵ_{mach}, depends on the particular rounding rules used. With rounding by chopping,

$$\epsilon_{mach} = \beta^{1-p},$$

(where β is the base and p is the number of digits in the mantissa) whereas with rounding to nearest

$$\epsilon_{mach} = \tfrac{1}{2}\beta^{1-p}.$$

The unit roundoff is important because it determines the maximum possible relative error in representing a nonzero real number x in a floating point system.
A characterization of the unit roundoff that you may sometimes see is that it is the smallest number ϵ such that

$$fl(1 + \epsilon) > 1.$$

Here $fl(x)$ is the floating point approximation to x.

Part 1:
Compute the machine precision on your computer. In addition compute the overflow and underflow threshold. Explain how you did the computation and your results.

Part 2:
Explain why an alternating infinite series, such as

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots$$

for $x < 0$, is difficult to evaluate accurately in floating point arithmetic.

Part 3:
What happens when you evaluate the infinite series?

$$\sum_{n=1}^{\infty} \frac{1}{n}$$

Explain why summing the series in floating point arithmetic yields a finite sum.