Math475a, Fall 2017. Assignment 2. Due September 14

Problem 1: Exponential function. The exponential function may be approximated by Taylor polynomials:

$$e^x \approx P_N(x) = \sum_{n=0}^{N} \frac{x^n}{n!}.$$

The sum may be computed directly term by term using

Algorithm I

```
input: x
s=1.0
t=x
for n=1 to N {
    s=s+t
    t=t*x/(n+1)
}
return s
```

Or using the identity, $P_N(x) = 1 + x\Big(1 + \frac{x}{2}\Big(1 + \frac{x}{3}\Big(1 + \cdots \frac{x}{N-1}\Big(1 + \frac{x}{N}\Big)\cdots\Big)\Big)\Big)$:

Algorithm II

```
input: x
s=1.0
for n=N to 1 {
    s=s*x/n+1
}
return s
```

Use these algorithms to compute e^{50} and e^{-50}. Assume that the built-in function in your environment (likely called "**exp()**") provides the exact value and plot the the error of your numerical approximations, i.e., $|P_N(50) - e^{\pm 50}|$, as the function of N. Values of $N \approx 200$ are probably as far as you need to go. (You should use the logarithmic plot to be able to display the entire range of error values in one figure, i.e., plot the logarithm of the error rather than the error itself.) If your environment uses very high precision arithmetic, you may not notice any significant difference between the algorithms, the error should become more noticeable if you use just single precision (32 bit) arithmetic. Discuss your findings.

Problem 2: Floating Point Numbers and Machine Precision. Start with $x = 1.0$ and keep dividing it by 2 until you get 0.0. If $n+1$ is the number of such divisions, then $1/2^n$ is the smallest number representable in the floating point format in the programming environment of your choice. Similarly, if at the stage $m+1$, $1.0 + x \equiv 1.0$, then $1.0 + 1/2^m$ is the floating point number closest to 1.0.

Now start doubling 1.0 until you get $+$INF (or NaN) at some stage, $N+1$; then $2^N(2 - 1/2^m)$ is the largest floating point number. Obtain m, n, and N. Discuss your findings; deduce the numbers of bits that are used in your environment for the *exponent* and *significand* of the floating point numbers.