CSC338: Tutorial 2

1. Show that in a floating point number system \( F(\beta, p, L, U) \), the largest floating point number is \((1-\beta^p)\beta^U+1\).

2. Consider, again, the IEEE SP floating point system \( F(\beta = 2, p = 24, L = -126, U = 127) \). We discussed in class that 23 bits are used to store the mantissa, 8 bits for the exponent, and 1 bit for the sign. Suppose we were to use 22 bits to store the mantissa and 9 bits to store the exponent. Would our new system have more or fewer normalized floating point numbers?

3. Consider the floating point number system \( F(\beta = 10, p = 4, L = -8, U = 8) \), and the following constants:

\[
\begin{align*}
a &= 5.659 \times 10^4 \\
b &= 5.629 \times 10^4 \\
c &= 9.337 \times 10^2 \\
d &= 7.529 \times 10^{-1}
\end{align*}
\]

What are the values of:
1. \( a + b \)
2. \( a - b \)
3. \( c/d \)
4. \( b \times d \)
5. \( (a + c) - (b + d) \)

Is there a way to rearrange the last computation, to make the result more accurate?

4. You are having trouble computing \( x^2 - y^2 \) with enough precision in floating point arithmetic. What can you do?