Lab 6

# Minifloat arithmetic

Recall from the arithmetic tutorial our minifloat format:

The first bit is the sign bit, the next 4 bits are the exponent (with a bias of 7), and the remaining 3 bits are in the form 1.mmm

Start with the same example from the board in lecture:

X = 0 1001 010

Y = 0 0111 110

Where the leading 1 on the exponent, the 2 and the 0111 are just all the standard, 0111 is the bias (7)

So what is X in base 10? 0b101 = 5

## What is Y in Decimal?

## X\*Y

X = 0 1001 010

Y = 0 0111 110

Lastly we add 7 to the bias (3+7 = 10 = 0b1010), drop all bits at the end.

Though notice that \*more\* correctly we might want to round up that trailing 1

## X+Y

X = 0 1001 010

Y = 0 0111 110  
  
Convert your answer back to base 10 to see how close to correct it is

**Steps: Shift one to the base of the other (there are actually rules about which one you should shift, but for our purposes it doesn’t matter), then add, then shift to a 1.xxx format. Make sure to account for rounding**

With rounding = 0 1001 110

## X-Y

**Steps: Same as addition, only, uh.. subtraction.**

That line might require a bit of explanation

101.000

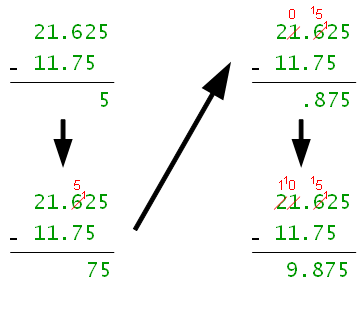
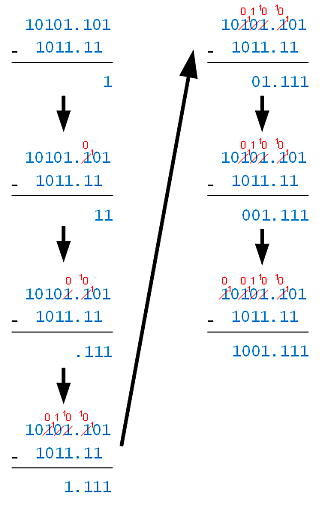
-1.110

To do that you keep borrowing from the left (as needed).

You might need a guide on binary subtraction for this:

<http://www.exploringbinary.com/binary-subtraction/>

I tried to find a nice way to write it up, but it didn’t work as well as their guide

Answer: Binary: 0 1000 101

## X/Y

X = 0 1001 010

Y = 0 0111 110

Steps: (Reverse of multiplication basically) X/Y = (Xf/Yf)\*2(Xe-Ye)

X/Y = 0 1000 011

If we convert that to decimal, we get 2.75. The correct answer is 2.86 (roughly) so we’re pretty close. (can’t do better due to the limitations of 8 bits)

# And 4 More Sample Problems

A= 1 0101 101 B = 0 0011 001

## Binary to Decimal

## A+B

Back to binary 1 0101 010

## A-B

Back to binary = 1 0101 111

## A\*B

Back to binary 1 0001 110  
  
Note with this question, if the exponent on A or B was 2 lower (say -4, -4) then the resulting exponent would be -8, add the bias… and you can’t represent -1 as an exponent so the problem wouldn’t have a valid solution.

## A/B

In binary 1 1001 110