CPSC 2311 Lab 8  
Charlie Hartsell  
[ckharts@clemson.edu](mailto:ckharts@clemson.edu)

For the purposes of this lab, repeating patterns will be underlined, as I cannot figure out how to place a line on top of text in word.

Convert 64.48 to binary. (Decimal -> IEEE 754 single precision binary)

1. Left hand side = 100 0000 (no work for this because we’ve already done lots of conversions from decimal to binary, so I’ve already proven that I understand how to do this part.)
2. Right hand side:  
   0.48\*2 = 0.96  
   0.96\*2 = 1.92  
   0.92\*2 = 1.84  
   0.84\*2 = 1.68  
   0.68\*2 = 1.36  
   0.36\*2 = 0.72  
   0.72\*2 = 1.44  
   0.44\*2 = 0.88  
   0.88\*2 = 1.76  
   0.76\*2 = 1.52  
   0.52\*2 = 1.04  
   0.04\*2 = 0.08  
   0.08\*2 = 0.16  
   0.16\*2 = 0.32  
   0.32\*2 = 0.64  
   0.64\*2 = 1.28  
   0.28\*2 = 0.56  
   0.56\*2 = 1.12  
   0.12\*2 = 0.24  
   0.24\*2 = 0.48  
   0.48\*2 = 0.96  
   At this point I found a repeating pattern.  
   Reading down, right hand side = 011110101110000101000
3. Full binary number:  
   1000000.011110101110000100000  
   After moving the decimal:  
   1.000000011110101110000100000 \* 2^6
4. Sign = 0  
   Exponent = 6 + 127 = 133 = 1000 0101  
   Mantissa = 000000011110101110000100000
5. Putting this together, we get the full number in IEE-754 single precision binary:  
   01000010100000001111010111000010

Convert 01000010100000001111010111000010 back to decimal. (Binary -> decimal)

1. Sign = 0 (positive number)  
   Exponent = 10000101 = 133 – 127 = 6  
   Mantissa = 00000001111010111000010
2. This comes out to:  
   1.0000000111101011100001 (binary) \* 2^6 (decimal)  
   Which equals: 1000000.0111101011100001
3. Convert 1000000.0111101011100001 to decimal:
   1. LH side = 64 (no work here again as explained earlier)
   2. RH side = 2^-2 +2^-3 +2^-4 +2^-5 +2^-7 +2^-9 +2^-10 + 2^-11 +2^-16  
       = 0.47999572753
   3. Combine them: 64 + 0.47999572753 = 64.47999572753

Convert -195.56 to binary. (Decimal -> IEEE 754 single precision binary)

1. LH side = 1100 0011 (not writing out my work here for previously stated reasons)
2. RH side:  
   0.56 \* 2 = 1.12  
   0.12 \* 2 = 0.24  
   0.24 \* 2 = 0.48  
   0.48 \* 2 = 0.96  
   0.96 \* 2 = 1.92  
   0.92 \* 2 = 1.84  
   0.84 \* 2 = 1.68  
   0.68 \* 2 = 1.36  
   0.36 \* 2 = 0.72  
   0.72 \* 2 = 1.44  
   0.44 \* 2 = 0.88  
   0.88 \* 2 = 1.76  
   0.76 \* 2 = 1.52  
   0.52 \* 2 = 1.04  
   0.04 \* 2 = 1.08  
   0.08 \* 2 = 0.16  
   0.16 \* 2 = 0.32  
   0.32 \* 2 = 0.64  
   0.64 \* 2 = 1.28  
   0.28 \* 2 = 0.56  
   At this point I found a repeating pattern.  
   Reading down, RH side = 1000 1111 0101 1100 0010
3. Putting it together: 11000011.10001111010111000010
4. After moving the decimal: 1.100001110001111010111000010 \* 2^7
5. Sign = 1 (negative number)  
   Exponent = 7 + 127 = 134 = 1000 0110  
   Mantissa = 100001110001111010111000010
6. Putting it into IEEE 754 single precision binary format with the correct number of bits:  
   11000001010000111000111101011100

Convert 11000011010000111000111101011100 back to decimal. (IEEE 754 single precision binary -> decimal)

1. Sign = 1 (negative number)  
   Exponent = 10000110 = 134 – 127 = 7  
   Mantissa = 100001110001111010111000010
2. Putting it together:  
   1. 100001110001111010111000010 \* 2^7
3. Moving the decimal 7 places to the right:  
   11000011.10001111010111000010
4. LH side comes out to 195 (not writing out my work here for previously stated reasons)
5. RH side:  
   2^-1 + 2^-6 + 2^-7 + 2^-8 + 2^-12 + 2^-13 + 2^-14 + 2^-15 + 2^-17 + 2^-19 + 2^-20 + 2^-21 = 0.52781248092
6. Combine the two sides: 195 + 0.52781248092 = 195.52781248092
7. After including the sign and rounding: -195.53

The reason why our number is slightly off is that we had to cut some bits of the mantissa off to fit within the 23-bit limit. 195.56 cannot be exactly represented in a finite number of bits due to its repeating nature in binary.