Colin Harfst

9/17/16

floatingpoint.pdf

I was given the following assignments.

1- to convert -17.65625 to a little-endian binary string and then into a little-endian hex string
2- to convert 0x00809f40 from little-endian hex into a floating point number

I will proceed with the second assignment first as it is slightly easier. First, I converted the given hex into the little-endian binary string 0000 0000 1000 0000 1001 1111 0100 0000. Then, I converted the binary string into its big-endian form 0100 0000 1001 1111 1000 0000 0000 0000. The first bit tells us that our "sign" is positive as it is a 0. The next 8 bits tell us our "exponent". Since 1000 0001 is 129 in base ten and $129 - 127 = 2$, 2 must be our exponent. The next 23 bits provide our "mantissa". 0011 1111 0000 0000 0000 000 has the third, fourth, fifth, sixth, seventh, and eighth most significant positions held with a 1. That means our mantissa is found by seeing that $\frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} + \frac{1}{128} + \frac{1}{256} = 0.24609375$. This gives us a final value as a floating point of $+1.24609375 * 2^2 = 4.984375$.

For the first assignment, we must think bit by bit. Since our value is negative, our first bit must be a 1. Note that $1 \leq \frac{17.65625}{2^4} < 2$, so the next 8 bits will represent the exponent $4 + 127 = 131$. In binary 131 is 1000 0011. The final 23 bits are carefully determined by figuring out that $.65625 = \frac{1}{16} + \frac{1}{32} + \frac{1}{128} + \frac{1}{512}$ so the fourth, fifth, seventh, and ninth bit of this set must be set to a 1. That is, 0001 1010 1000 0000 0000 000. So our overall big-endian binary string is 1100 0001 1000 1101 0100 0000 0000 0000. Which is 0xc18d4000. So the little-endian hex string is 0x00408dc1.