CSCI 320, Computer Architecture, Spring 2013

IEEE 754 Examples

1. Give the IEEE 754 single precision floating point representation of the decimal number -91.6875.

   Start by converting 91.6875 into normalized binary format.

   First convert the integer part to binary

$$\begin{array}{rcrcl} 91 \div 2 & = & 45 & R & 1 \\ 45 \div 2 & = & 22 & R & 1 \\ 22 \div 2 & = & 11 & R & 0 \\ 11 \div 2 & = & 5 & R & 1 \\ 5 \div 2 & = & 2 & R & 1 \\ 2 \div 2 & = & 1 & R & 0 \\ 1 \div 2 & = & 0 & R & 1 \end{array}$$

   $(91)_{10} = (1011011)_2$

   Next convert the fractional part to binary

$$\begin{array}{rcl} .6875 \times 2 & = & 1.375 \\ .375 \times 2 & = & 0.75 \\ .75 \times 2 & = & 1.5 \\ .5 \times 2 & = & 1.0 \end{array}$$

   $(0.6875)_{10} = (0.1011)_2$

   So $(-91.6875)_{10} = (-1011011.1011)_2$
   In normalized form this is -1.0110111011×2^6

   Now we can construct the IEEE 754 representation:
   Since the number is negative, the first bit will be 1.
   The exponent is 6, which in excess 127 format is $127 + 6 = (133)_{10} = (10000101)_2$.
   The mantissa is taken directly from the fractional portion of the normalized format.

   Putting this all together, we have

```
1 10000101 01101110110000000000000
```

   Converting this to hex we have

```
1100 0010 1011 0111 0110 0000 0000 0000
  C    2    B    7    6    0    0    0
```

   So the IEEE 754 single precision floating point representation of the decimal number -91.6875 is `C2B76000`.

2. Give the decimal number represented by the IEEE 754 single precision floating point representation `3CC80000`.

   First convert to binary.

   ```
     3    C    C    8    0    0    0    0
   0011 1100 1100 1000 0000 0000 0000 0000
   ```

   Then group the bits into the parts of the IEEE format.

   ```
   0 01111001 10010000000000000000000
   ```

   The sign bit is zero, so the number is positive.

   The exponent is $(01111001)_2 = (121)_{10}$
   The exponent is stored in excess 127, so the exponent is $121 - 127 = -6$.

   Adding the hidden bit to the mantissa we have 1.1001

   Putting the pieces together we have 1.1001×2^{-6}.

   Now convert to decimal:
   $1.1001 \times 2^{-6} = 0.0000011001$
   Multiplying by 2^{10} we get
   $(11001)_2 = (25)_{10}$
   Now divide by 2^{10}
   $\frac{25}{2^{10}} = 0.0244140625$

   So the decimal number represented by the IEEE 754 single precision floating point representation `3CC80000` is 0.0244140625.

3. Give the IEEE 754 single precision floating point representation of the decimal number 6.1.

   Start by converting 6.1 into normalized binary format.

   First convert the integer part to binary. We can do this in our head. $(6)_{10} = (110)_2$

   Next convert the fractional part to binary

$$
\begin{aligned}
.1 \times 2 &= 0.2 \\
.2 \times 2 &= 0.4 \\
.4 \times 2 &= 0.8 \\
.8 \times 2 &= 1.6 \\
.6 \times 2 &= 1.2 \\
.2 \times 2 &= 0.4 \\
.4 \times 2 &= 0.8 \\
.8 \times 2 &= 1.6 \\
.6 \times 2 &= 1.2 \\
.2 \times 2 &= 0.4 \\
.4 \times 2 &= 0.8 \\
.8 \times 2 &= 1.6 \\
.6 \times 2 &= 1.2 \\
.2 \times 2 &= 0.4 \\
.4 \times 2 &= 0.8
\end{aligned}
$$

   This is a repeating fraction and not an exact representation. We'll use extra digits for now and limit them when forming the IEEE 754 representation.

   $(0.1)_{10} \approx (0.000110011001100110011001100110011001100)_2$.

   So $(6.1)_{10} \approx (110.000110011001100110011001100110011001100)_2$

   In normalized form this is $1.10000110011001100110011001100110011001100 \times 2^2$

   Now we can construct the IEEE 754 representation:

   Since the number is positive, the first bit will be 0.

   The exponent is 2, which in excess 127 format is $127 + 2 = (129)_{10} = (10000001)_2$.

   For the mantissa we use the normalized form. We don't use the leading (hidden) one, but use only the first 23 bits of the fractional part. So the mantissa is 10000110011001100110011.

   Putting this all together, we have

```
0 10000001 10000110011001100110011
```

   Converting to hex we get

```
0100 0000 1100 0011 0011 0011 0011 0011
  4    0    C    3    3    3    3    3
```

   So the IEEE 754 single precision floating point representation of the decimal number 6.1 is `40C33333`.

4. Give the decimal number represented by the IEEE 754 single precision floating point representation `FF000001`.

   First convert to binary.

```
  F    F    0    0    0    0    0    1
1111 1111 0000 0000 0000 0000 0000 0001
```

   Then group the bits into the parts of the IEEE format.

```
1 11111110 00000000000000000000001
```

   The sign bit is one, so the number is negative.

   The exponent is $(11111110)_2 = (254)_{10}$
   The exponent is stored in excess 127, so the exponent is $274 - 127 = 127$.

   Adding the hidden bit to the mantissa we have 1.00000000000000000000001

   Putting the pieces together we have $1.00000000000000000000001 \times 2^{127}$.

   Now convert to decimal:

$$
\begin{aligned}
1.00000000000000000000001 \times 2^{127} &= 100000000000000000000001.0 \times 2^{127-23} && \textit{Move radix point} \\
&= 100000000000000000000001.0 \times 2^{104} \\
&= \left(2^{23} + 1\right) \times 2^{104} && \textit{Convert integer part} \\
&= 2^{23+104} + 2^{104} && \textit{Distribute } 2^{104} \\
&= 2^{127} + 2^{104} \\
&= 170141183460469231731687303715884105728 \\
&\quad + 20282409603651670423947251286016 \\
&= 170141203742878835383357727663135391744 \\
&= 1.70141203742878835383357727663135391744 \times 10^{38}
\end{aligned}
$$

   So the decimal number represented by the IEEE 754 single precision floating point representation `FF000001` is $1.70141203742878835383357727663135391744 \times 10^{38}$.