Name: Ashley Nguyen - apn2my
PreLab04 - floatingpoint.pdf
2/14/16

1. *Floating Point → Little Endian Binary (expressed in Hexadecimal)*

My Magic Number = -17.5625

<u>Sign</u> = 1_b (negative)

<u>8-bit Exponent</u> = $17.5625/2^4 = 1.09765625$

$= 4 + 127 = 131 \Rightarrow 1000\ 0011_b$

<u>23-bit Mantissa</u> = 1.09765625

1.09765625 – 1 = 0.09765625

$$0.09765625 = \frac{25}{256}$$

$$\frac{25}{256} - \frac{\mathbf{16}}{\mathbf{256}} = \frac{9}{256}$$

$$\frac{9}{256} - \frac{\mathbf{8}}{\mathbf{256}} = \frac{1}{256}$$

$$\frac{1}{256} - \frac{\mathbf{1}}{\mathbf{256}} = 0$$

$$\frac{25}{256} = \frac{\mathbf{16}}{\mathbf{256}} + \frac{\mathbf{8}}{\mathbf{256}} + \frac{\mathbf{1}}{\mathbf{256}}$$

$$= \left(\frac{1}{2}\right)^4 + \left(\frac{1}{2}\right)^5 + \left(\frac{1}{2}\right)^8$$

Insert 1's at bit locations based on exponents:

⇨ $000\ 1100\ 1000\ 0000\ 0000\ 0000_b$

<u>Big Endian Binary (Sign + Exp. + Mantissa)</u> = $1100\ 0001\ 1000\ 1100\ 1000\ 0000\ 0000_b$

<u>Big Endian Hexadecimal:</u> 0xC18C8000

<u>Little Endian Hexadecimal:</u> 0x00808CC1 or 0x808CC1

2. Hexadecimal → Base 10 Real Number

My Magic Number = 0x00809f40

Convert to Big Endian → 0x409f8000

Expand to Binary form → 0100 0000 1001 1111 1000 0000 0000 0000

Converting to floating point number:
<u>Sign</u> = 0

<u>8-bit exponent</u> = 1000001
Exponent = 10000001 – 127 = 129 (converted to base 10) – 127 = **2**

<u>23-bit mantissa</u> = 001 1111 1000 0000 0000 0000

Summing up ½ to the power of the '1' bit position:

$$= \left(\frac{1}{2}\right)^3 + \left(\frac{1}{2}\right)^4 + \left(\frac{1}{2}\right)^5 + \left(\frac{1}{2}\right)^6 + \left(\frac{1}{2}\right)^7 + \left(\frac{1}{2}\right)^8$$

= 0.24609375

Adding 1:
= 1 + 0. 24609375
= 1.24609375

Multiply by exponent:
= 1. 24609375 * 2^2

Final Answer in floating point number:
= 4.984375$_d$