Assignment 2 - SOLUTION

Objectives:
In this assignment, you will gain familiarity with:

- IEEE floating point representation

Submission:

- When creating your assignment, first include the question itself and its number then include your answer, keeping the questions in its original numerical order.

- **If you hand-write your answers (as opposed to using a computer application to write them):** When putting your assignment together, do not take photos (no .jpg) of your assignment sheets! Scan them instead! Better quality -> easier to read -> easier to mark!

- Submit your assignment electronically on CourSys

Due:

- Thursday, Jan. 30 at 3pm
- Late assignments will receive a grade of 0, but they will be marked in order to provide the student with feedback.

Requirements:

- **Show your work** (as illustrated in lectures).

Marking scheme:
This assignment will be marked as follows:

- Questions 1 and 2 will be marked for correctness.

The amount of marks for each question is indicated as part of the question.
1. [8 marks] Floating point conversion and Rounding.
   
a. Represent the following numbers in IEEE floating point representation (single precision), clearly showing the effect of rounding on the “frac” (mantissa), if rounding occurs, and express your final answer in binary and in hexadecimal form:

   0.001111111₂
   
   * Normalized: \( 0.1111111₁₂ \times 2^{-3} \)
   * \( S = 0 \) (the number is positive)
   * \( M = 1 + \text{frac} \)
   * \( \text{frac} = 1111100\ldots00 \)
   * \( E = \text{exp} - 127 \) (bias = \( 2^{k-1} - 1 = 2^8 - 1 = 127 \))
   * \(-3 + 127 = \text{exp} \)
   * \( \text{exp} = 124₁₀ \rightarrow 01111₁₀₀₂ \)

   ∴ \( 0.001111111₁₂ \rightarrow 0.1111111₀ \times 2^{-3} \)

   No rounding occurred
II. 3.1416015625<sub>10</sub>

2. ii. 3.1416015625<sub>10</sub>

- In binary: 11.0010010001₂
  - Normalized: 1.0010010001₂ x 2<sup>4</sup>
- S = 0
- M = 1 + frac
- frac = 10010010001<sub>10</sub> (pad with 0's)
  - frac = 10010010001000000000000000000000₂
- E = exp - 127
  - 1 + 127 = exp
  - exp = 128<sub>10</sub> → 10000000₂

- 0 10000000 0100 1001 0001 0000 0000 0000
- 0x40491000

No rounding occurred
III. \(-0.9_{10}\)

2. III. \(-0.9_{10}\)

- In binary: \(-0.11100_2\)
- Normalized: \(-1.1100_2 \times 2^{-1}\)
- \(s = 1\) (the number is negative)
- \(M = 1 + \text{frac}\)

\[
\text{Rounding frac:}
\]
\[
\text{frac} = 110011001100110011001100...\overline{1100}...
\]

- \(E = \exp -127\)
- \(-1 + 127 = \exp\)
- \(\exp = 127_{10} \rightarrow 01111110_2\)

\[
10111110 1100 1100 1100 1100 1100 1100
\]

\(0 \times \text{BF} 6666666\)
IV. 1/3_{10} (a third)

2. iv. \( \frac{1}{3} = 0.3 \)

- In binary: 0.01\_2
- Normalized: \( 0.010101\ldots \times 2^{-2} \)
- \( S = 0 \)
- \( M = 1 + \text{frac} \)
- \( \text{frac} = 010101010101010101010101010101\ldots \)
- \( E = \exp -127 \)
- \(-2 + 127 = 125\_e\)
- \( \exp = 125\_e \rightarrow 01111101\_2 \)

\( \therefore \) 0x3EAAAAAAB
b. Convert 0x4AE4C1A from IEEE floating point representation (single precision) to a real number.

\[ \text{0x4AE4C1A} \]

\[ \begin{array}{c}
\text{01001010} \\
\text{11010100} \\
\text{11000011}
\end{array} \]

\[ s \text{ exp} \quad \text{frac} \]

\* Positive number

\* \[ E = \text{exp - bias} = 149 - 127 = 22 \quad \therefore 2^{22} \]

\[ 10010101 \]

\[ 2^7 + 2^4 + 2^2 + 2^0 \]

\[ = 128 + 16 + 4 + 1 \]

\[ = 149 \]

\* \[ M = 1 + .11010100100110000011010 \]

\[ = 1.11010100100110000011010 \approx 1.83044\ldots \]

\[ 0.5 + 0.25 + 0.0625 + 0.015625 + 0.001953125 + \]

\[ 0.000244140625 + 0.000078125 + \]

\[ 0.0000078125 + 0.0000009375 + 0.00000003828125 + \]

\[ 0.0000000000015625 \]

\[ V = (-1)^{22} \cdot 1.83044743844 \]

\[ V = 7677453 \]
c. Round the following binary numbers (rounding position is bolded - $2^4$ position) following the rounding rules of the IEEE floating point representation.

I. $1.001111_2$
II. $1.1001001_2$
III. $1.0111100_2$
IV. $1.0110100_2$

For each of the above rounded binary numbers, indicate what type of rounding you performed and compute the value that is either added to or subtracted from the original number (listed above) as a result of the rounding process. In other words, compute the error introduced by the rounding process.

1. \[1.2421875_{10} \rightarrow 1.0011112\]
   \[-0.0078125_{10} \rightarrow -0.01000002\]
   \[
   \text{the error introduced by the rounding process is}
   \]
   \[
   \begin{array}{c}
   1.0011112 \\
   - 1.01000002
   \end{array}
   \]
   \[
   \begin{array}{c}
   1.2421875_{10} \\
   - 1.25_{10}
   \end{array}
   \]
   \[
   \begin{array}{c}
   -0.0078125_{10} \\
   -0.0078125_{10}
   \end{array}
   \]
   \[
   i.e., 0.0078125_{10} \text{ has been added to the original value } 1.0011112\text{ as part of the rounding process.}
   \]
10. $1.1001001_2$

(half way (half way is $1.1001100_2$))

so we round down and discard the bits to the right of the rounding position.

The error introduced by the rounding process is

\[
1.1001001_2 - 1.1001000_2
\]

\[
0.0000001_2 \rightarrow 7.8125 \times 10^{-3} \text{ or } 0.0078125
\]

i.e., $0.0078125_2$ has been subtracted from the original value $1.1001001_2$ as part of the rounding process.
iii. 1.0111100₂

Is exactly halfway so we round to even number by rounding to closest even number i.e., a number with a zero in rounding position and discard the bits to the right of rounding position.

\[
\begin{align*}
\text{Round position} & : \quad \text{1.1000000₂} \\
\text{Consider the absolute value} & : \quad \text{1.5₁₀} \\
\text{Add} & : \quad -0.03125₁₀ \rightarrow | -0.03125₁₀ | \text{ of error}
\end{align*}
\]

i.e., 0.03125₀ has been added to the original value 1.0111₁₀₀₂ as part of the rounding process.

**Note:** Could 1.0111₁₀₀₂ be rounded to 1.01₁₁₀₂ instead of 1.1₀₀₀₂ ?

No, because the error \((1.0111₁₀₀₂ - 1.01₁₁₀₂)\) is larger \((1.46875₁₀ - 1.375₁₀ = 0.09375₁₀)\) so 1.0₁₁₀₂ is not the closest even number to our original number 1.0111₁₀₀₂.
iv. 1.011\textcolor{red}{0}100_2

is exactly half way so we round to even number by rounding to closest even number i.e. a number with a zero in rounding position. Since the rounding position is already a zero, we do not transform it and simply discard the bits to the right of rounding position.

\[ : 1.0110_2 \]

the error introduced by the rounding process is 1.0110100_2
\[
\begin{align*}
-1.0110000_2 \\
&= 0.0000100_2 \\
&= 0.03125_{10}
\end{align*}
\]

i.e., 0.03125_{10} has been subtracted from the original value 1.0110100_2 as part of the rounding process.
2. [12 marks] Creating hypothetical smaller floating-point representations based on the IEEE floating point format allows us to investigate this encoding scheme more easily, since the numbers are easier to manipulate and compute.

Below is a table listing several real numbers represented as 6-bit floating-point numbers \((w = 6)\). The format of these 6-bit floating-point numbers is as follows: 1 bit is used to express for the sign, 3 bits are used to express “exp” \((k = 3)\) and 2 bits are used to represent “frac” \((n = 2)\), in the following order: sign  exp  frac.

Complete the table (the same way as in Figure 2.35 in our textbook) then answer the questions below the table.

Tip: Have a look at Figure 2.35 in our textbook, which illustrates a similar table for a hypothetical 8-bit floating-point format. This will give you an idea of how to complete the table. Also, Figure 2.34 displays the complete range of these 6-bit floating point numbers as well as their values between -1.0 and 1.0. This diagram may be helpful when you are checking your work.

<table>
<thead>
<tr>
<th>Description</th>
<th>Bit representation</th>
<th>exp</th>
<th>E</th>
<th>2^E</th>
<th>frac</th>
<th>M</th>
<th>M 2^E</th>
<th>V</th>
<th>Decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>zero</td>
<td>0000 00</td>
<td>0</td>
<td>-2</td>
<td>1/4</td>
<td>0/4</td>
<td>0/4</td>
<td>0/16</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>Smallest positive denormalized</td>
<td>0000 01</td>
<td>0</td>
<td>-2</td>
<td>1/4</td>
<td>¾</td>
<td>¾</td>
<td>1/16</td>
<td>1/16</td>
<td>0.0625</td>
</tr>
<tr>
<td></td>
<td>0000 10</td>
<td>0</td>
<td>-2</td>
<td>1/4</td>
<td>2/4 = ½</td>
<td>2/4 = ½</td>
<td>2/16</td>
<td>2/16</td>
<td>0.125</td>
</tr>
<tr>
<td>Largest positive denormalized</td>
<td>0000 11</td>
<td>0</td>
<td>-2</td>
<td>1/4</td>
<td>¾</td>
<td>¾</td>
<td>3/16</td>
<td>3/16</td>
<td>0.1875</td>
</tr>
<tr>
<td>Smallest positive normalized</td>
<td>0001 00</td>
<td>1</td>
<td>-2</td>
<td>1/4</td>
<td>0/4</td>
<td>4/4 = 1</td>
<td>4/16</td>
<td>4/16</td>
<td>0.25</td>
</tr>
<tr>
<td></td>
<td>0001 01</td>
<td>1</td>
<td>-2</td>
<td>1/4</td>
<td>¾</td>
<td>5/4</td>
<td>5/16</td>
<td>5/16</td>
<td>0.3125</td>
</tr>
<tr>
<td></td>
<td>0001 10</td>
<td>1</td>
<td>-2</td>
<td>1/4</td>
<td>2/4 = ½</td>
<td>6/4</td>
<td>6/16</td>
<td>6/16</td>
<td>0.375</td>
</tr>
<tr>
<td></td>
<td>0001 11</td>
<td>1</td>
<td>-2</td>
<td>1/4</td>
<td>¾</td>
<td>7/4</td>
<td>7/16</td>
<td>7/16</td>
<td>0.4375</td>
</tr>
</tbody>
</table>

Notice the smooth transition from 3/16 to 4/16
<table>
<thead>
<tr>
<th>Decimal</th>
<th>Binary</th>
<th>Description</th>
<th>Fraction</th>
<th>Fractional Part</th>
<th>Sum</th>
<th>Sum of Fractions</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 010 00</td>
<td>2</td>
<td>-1</td>
<td>1/2</td>
<td>0/4</td>
<td>4/4 = 1</td>
<td>4/8</td>
</tr>
<tr>
<td>0 010 01</td>
<td>2</td>
<td>-1</td>
<td>1/2</td>
<td>¼</td>
<td>5/4</td>
<td>5/8</td>
</tr>
<tr>
<td>0 010 10</td>
<td>2</td>
<td>-1</td>
<td>1/2</td>
<td>2/4 = ½</td>
<td>6/4</td>
<td>6/8</td>
</tr>
<tr>
<td>0 010 11</td>
<td>2</td>
<td>-1</td>
<td>1/2</td>
<td>¾</td>
<td>7/4</td>
<td>7/8</td>
</tr>
<tr>
<td>One</td>
<td>0 011 00</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0/4</td>
<td>4/4 = 1</td>
</tr>
<tr>
<td>0 011 01</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>¼</td>
<td>5/4</td>
<td>5/4</td>
</tr>
<tr>
<td>0 011 10</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2/4 = ½</td>
<td>6/4</td>
<td>6/4</td>
</tr>
<tr>
<td>0 011 11</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>¾</td>
<td>7/4</td>
<td>7/4</td>
</tr>
<tr>
<td>0 100 00</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>0/4</td>
<td>4/4 = 1</td>
<td>8/4</td>
</tr>
<tr>
<td>0 100 01</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>¼</td>
<td>5/4</td>
<td>10/4</td>
</tr>
<tr>
<td>0 100 10</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>2/4 = ½</td>
<td>6/4</td>
<td>12/4</td>
</tr>
<tr>
<td>0 100 11</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>¾</td>
<td>7/4</td>
<td>14/4</td>
</tr>
<tr>
<td>0 101 00</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>0/4</td>
<td>4/4 = 1</td>
<td>16/4</td>
</tr>
<tr>
<td>0 101 01</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>¼</td>
<td>5/4</td>
<td>20/4</td>
</tr>
<tr>
<td>0 101 10</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>2/4 = ½</td>
<td>6/4</td>
<td>24/4</td>
</tr>
<tr>
<td>0 101 11</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>¾</td>
<td>7/4</td>
<td>28/4</td>
</tr>
<tr>
<td></td>
<td>0 110 00</td>
<td>6</td>
<td>3</td>
<td>8</td>
<td>0/4</td>
<td>4/4 = 1</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>-----</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td>0 110 01</td>
<td>6</td>
<td>3</td>
<td>8</td>
<td>¼</td>
<td>5/4</td>
</tr>
<tr>
<td></td>
<td>0 110 10</td>
<td>6</td>
<td>3</td>
<td>8</td>
<td>2/4 = ½</td>
<td>6/4</td>
</tr>
<tr>
<td>Largest positive</td>
<td>0 110 11</td>
<td>6</td>
<td>3</td>
<td>8</td>
<td>¾</td>
<td>7/4</td>
</tr>
<tr>
<td>normalized</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ Infinity</td>
<td>0 111 00</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>NaN</td>
<td></td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

a. What is the value of the bias?

\[ \text{bias} = 2^{k-1}-1 \text{ and since } k=3 \text{ then } \text{bias} = 2^{3-1} = 2^2 - 1 = 4 - 1 = 3 \]

b. Consider two adjacent denormalized numbers. How far apart are they? Expressed this difference (“delta”) as a decimal number.

\[ \Delta_d = \frac{1}{16} = 0.0625 \]

c. Consider two adjacent normalized numbers ...
   a. with the \textbf{exp} field set to 001. How far apart are they?
   b. with the \textbf{exp} field set to 010. How far apart are they?
   c. with the \textbf{exp} field set to 011. How far apart are they?

Expressed these differences (“delta”) as decimal numbers.
d. Without doing any calculations, can you guess how far apart are two adjacent normalized numbers ...

a. with the exp field set to 100?

b. with the exp field set to 101?

c. with the exp field set to 110?

\[
\begin{align*}
\Delta_{100} & \to \frac{8}{16} = 0.5 \\
\Delta_{101} & \to \frac{16}{16} = 1 \\
\Delta_{110} & \to \frac{22}{16} = 2
\end{align*}
\]

\[M \cdot 2^E\] increases by a power of 2.

\[
\{\text{range of real numbers} \to [-14.0, 14.0] \text{ not considering } \pm\infty, \text{NaN} \}
\]

(e) What is the “range” (not contiguous) of real numbers that can be represented using this 6-bit floating-point representation?

(f) What is the range of the normalized exponent $E$ (found in the equation $v = (-1)^s M 2^E$) which can be represented by this 6-bit floating-point representation?
g. Give an example of a real number that cannot be represented using this 6-bit floating-point representation, but is within the “range” of representable values.

11.0 cannot be represented but it is within the range

h. Give an example of a real number that would overflow if we were trying to represent it using this 6-bit floating-point representation. The best way to answer this question is to convert this real number into a 6-bit IEEE floating-point representation and clearly indicate why it would overflow.

16.0_{10} \rightarrow \begin{align*}
\text{in binary:} & \quad 10000 \\
\text{normalized:} & \quad 1.0000 \times 2^4 \\
\text{sign:} & \quad s = 0 \\
\text{Exponent:} & \quad E = \text{exp} - \text{bias} = 4 + 3 = 7 \implies \text{exp} = 7 \rightarrow 111 \\
\text{However, exp 111 is outside exp's range and therefore would overflow.}
\end{align*}
i. How close is the value of the “frac” of the largest normalized number to 1? In other words, what is $\epsilon$ (epsilon)? Expressed it as a decimal number.

Answer:

The value of the “frac” of the largest normalized number is $0.11 \rightarrow \frac{3}{4} = 0.75_{10}$

How close is the value of the “frac” of the largest normalized number to 1 $\rightarrow \frac{1}{4} = 0.25_{10}$

So, $\epsilon$ (epsilon) is $\frac{1}{4} = 0.25_{10}$