Lecture 8 - Floating Point Arithmetic, The IEEE Standard

MIT 18.335J / 6.337J
Introduction to Numerical Methods

Per-Olof Persson (persson@mit.edu)
October 3, 2007

Floating Point Formats

- Scientific notation:
  \[ -1.602 \times 10^{-19} \]
  \[ \text{sign} \quad \text{significand} \quad \text{base} \quad \text{exponent} \]

- Floating point representation
  \[ \pm \left( d_0 + d_1 \beta^{-1} + \ldots + d_{p-1} \beta^{-(p-1)} \right) \beta^e, \quad 0 \leq d_i < \beta \]
  with base \(\beta\) and precision \(p\)

- Exponent range \([e_{\text{min}}, e_{\text{max}}]\)

- Normalized if \(d_0 \neq 0\) (use \(e = e_{\text{min}} - 1\) to represent 0)
Floating Point Numbers

- The gaps between adjacent numbers scale with the size of the numbers
- Relative resolution given by *machine epsilon*, \( \epsilon_{\text{machine}} = .5 \beta^{1-p} \)
- For all \( x \), there exists a floating point \( x' \) such that \( |x - x'| \leq \epsilon_{\text{machine}} |x| \)
- Example: \( \beta = 2, p = 3, \epsilon_{\text{min}} = -1, \epsilon_{\text{max}} = 2 \)

Special Quantities

- \( \pm \infty \) is returned when an operation overflows
- \( x/ \pm \infty = 0 \) for any number \( x \), \( x/0 = \pm \infty \) for any nonzero number \( x \)
- Operations with infinity are defined as limits, e.g.
  \[
  4 - \infty = \lim_{x \to \infty} 4 - x = -\infty
  \]
- NaN (Not a Number) is returned when the an operation has no well-defined finite or infinite result
- Examples: \( \infty - \infty, \infty/\infty, 0/0, \sqrt{-1}, \text{NaN} \odot x \)
Denormalized Numbers

- With normalized significand there is a “gap” between 0 and $\beta^{e_{\min}}$.
- This can result in $x - y = 0$ even though $x \neq y$, and code fragments like
  \[
  \text{if } x \neq y \text{ then } z = 1/(x - y)\]
  might break.
- Solution: Allow non-normalized significand when the exponent is $e_{\min}$.
- This \textit{gradual underflow} guarantees that
  \[
  x = y \iff x - y = 0
  \]

---

IEEE Single Precision

- 1 sign bit, 8 exponent bits, 23 significand bits:

  \[
  \begin{array}{|c|c|c|c|c|}
  \hline
  S & E & M \\
  \hline
  0 & 00000000 & 00000000000000000000000000000000 \\
  \hline
  \end{array}
  \]

- Represented number:
  \[
  (-1)^S \times 1.M \times 2^{E-127}
  \]

- Special cases:

  \[
  \begin{array}{|c|c|c|c|}
  \hline
  & E = 0 & 0 < E < 255 & E = 255 \\
  \hline
  M = 0 & \pm 0 & \text{Powers of 2} & \pm \infty \\
  \hline
  M \neq 0 & \text{Denormalized} & \text{Ordinary numbers} & \text{NaN} \\
  \hline
  \end{array}
  \]
### IEEE Single Precision, Examples

<table>
<thead>
<tr>
<th>S</th>
<th>E</th>
<th>M</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>11111111</td>
<td>00000100000000000000000000000000</td>
<td>NaN</td>
</tr>
<tr>
<td>1</td>
<td>11111111</td>
<td>001000100010010101010101010101010</td>
<td>NaN</td>
</tr>
<tr>
<td>0</td>
<td>11111111</td>
<td>00000000000000000000000000000000</td>
<td>$\infty$</td>
</tr>
<tr>
<td>0</td>
<td>10000000</td>
<td>10100000000000000000000000000000</td>
<td>$+1 \cdot 2^{129-127} \cdot 1.101 = 6.5$</td>
</tr>
<tr>
<td>0</td>
<td>10000000</td>
<td>00000000000000000000000000000000</td>
<td>$+1 \cdot 2^{128-127} \cdot 1.0 = 2$</td>
</tr>
<tr>
<td>0</td>
<td>00000001</td>
<td>00000000000000000000000000000000</td>
<td>$+1 \cdot 2^{1-127} \cdot 1.0 = 2^{-126}$</td>
</tr>
<tr>
<td>0</td>
<td>00000000</td>
<td>10000000000000000000000000000000</td>
<td>$+1 \cdot 2^{-126} \cdot 0.1 = 2^{-127}$</td>
</tr>
<tr>
<td>0</td>
<td>00000000</td>
<td>0000000000000000000000000000000001</td>
<td>$+1 \cdot 2^{-126} \cdot 2^{-23} = 2^{-149}$</td>
</tr>
<tr>
<td>0</td>
<td>00000000</td>
<td>0000000000000000000000000000000000</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>00000000</td>
<td>0000000000000000000000000000000000</td>
<td>$-0$</td>
</tr>
<tr>
<td>1</td>
<td>10000001</td>
<td>1010000000000000000000000000000000</td>
<td>$-1 \cdot 2^{129-127} \cdot 1.101 = -6.5$</td>
</tr>
<tr>
<td>1</td>
<td>11111111</td>
<td>0000000000000000000000000000000000</td>
<td>$-\infty$</td>
</tr>
</tbody>
</table>

### IEEE Floating Point Data Types

<table>
<thead>
<tr>
<th></th>
<th>Single precision</th>
<th>Double precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Significand size ($p$)</td>
<td>24 bits</td>
<td>53 bits</td>
</tr>
<tr>
<td>Exponent size</td>
<td>8 bits</td>
<td>11</td>
</tr>
<tr>
<td>Total size</td>
<td>32 bits</td>
<td>64 bits</td>
</tr>
<tr>
<td>$e_{\text{max}}$</td>
<td>+127</td>
<td>+1023</td>
</tr>
<tr>
<td>$e_{\text{min}}$</td>
<td>-126</td>
<td>-1022</td>
</tr>
<tr>
<td>Smallest normalized</td>
<td>$2^{-126} \approx 10^{-38}$</td>
<td>$2^{-1022} \approx 10^{-308}$</td>
</tr>
<tr>
<td>Largest normalized</td>
<td>$2^{127} \approx 10^{38}$</td>
<td>$2^{1023} \approx 10^{308}$</td>
</tr>
<tr>
<td>$\epsilon_{\text{machine}}$</td>
<td>$2^{-24} \approx 6 \cdots 10^{-8}$</td>
<td>$2^{-53} \approx 10^{-16}$</td>
</tr>
</tbody>
</table>
Floating Point Arithmetic

- Define $\text{fl}(x)$ as the closest floating point approximation to $x$.

- By the definition of $\varepsilon_{\text{machine}}$, we have for the relative error:
  
  $$\text{For all } x \in \mathbb{R}, \text{ there exists } \varepsilon \text{ with } |\varepsilon| \leq \varepsilon_{\text{machine}} \text{ such that } \text{fl}(x) = x(1 + \varepsilon)$$

- The result of an operation $\oplus$ using floating point numbers is $\text{fl}(a \oplus b)$.

- If $\text{fl}(a \oplus b)$ is the nearest floating point number to $a \oplus b$, the arithmetic rounds correctly (IEEE does), which leads to the following property:
  
  $$\text{For all floating point } x, y, \text{ there exists } \varepsilon \text{ with } |\varepsilon| \leq \varepsilon_{\text{machine}} \text{ such that } x \oplus y = (x \ast y)(1 + \varepsilon)$$

- Round to nearest even in the case of ties.