TU Hamburg-Harburg – Institut für Zuverlässiges Rechnen
Prof. Dr. S.M. Rump und Mitarbeiter, Wintersemester 2016/2017

Prozedurale Programmierung – IEEE 754 Erläuterungen

Eine 4-Byte-Gleitkommazahl (`binary32` oder der C-Datentyp `float`) wird nach dem IEEE 754 Standard im Speicher folgendermaßen dargestellt:

sign	exponent			significant			
s	e_1	$\dots$	e_8	m_1	m_2	$\dots$	m_{23}

Gespeichert werden folgende drei Komponenten:

- `sign`: Vorzeichenbit s
- `exponent`: "biased" (um $B = 127$ geshifteter) Exponent

$$e := \sum_{i=1}^{8} 2^{8-i} e_i = 128e_1 + 64e_2 + \cdots + 2e_7 + e_8$$

- fraction (gespeicherte Anteil vom `significant`)

$$f := \sum_{i=1}^{23} 2^{-i} m_i = \frac{m_1}{2} + \frac{m_2}{4} + \frac{m_3}{8} \cdots + \frac{m_{23}}{2^{23}}$$

Die **normalisierte** (Fall 3) Gleitkommazahl x ergibt sich aus den gespeicherten Komponenten:

$$x = \underbrace{(-1)^s}_{sign} \cdot \underbrace{(\underline{1} + f)}_{significant} \cdot \underbrace{2^{e-B}}_{exponent} \tag{1}$$

Die **denormalisierte** (Fall 4) Gleitkommazahl x ergibt sich aus den gespeicherten Komponenten (ohne die **implizite Eins**):

$$x = \underbrace{(-1)^s}_{sign} \cdot \underbrace{(\underline{0} + f)}_{significant} \cdot \underbrace{2^{e-B}}_{exponent} \tag{2}$$

Für die Interpretation der dargestellten Gleitkommazahl x unterscheidet man folgende 5 Fälle:

Fall	exponent	fraction	Resultierende Gleichpunktzahl
1	$e = 255$ $(E = 128)$	$f \neq 0$	$x = nan$ (not a number)
2	$e = 255$ $(E = 128)$	$f = 0$	$x = (-1)^s \cdot inf$ (infinite ∞)
3	$0 < e < 255$ $(-127 < E < 128)$	$f \neq 0$	Gleichung (1)
4	$e = 0$ $(E = -127)$	$f \neq 0$	Gleichung (2)
5	$e = 0$ $(E = -127)$	$f = 0$	$x = 0$