

Grundlagen der Betriebssysteme

Tim Luchterhand, Paul Nykiel (Gruppe 017)

7. Mai 2018

1 Festkomma Darstellung

(a)

$$7.75 = 4 + 2 + 1 + 0.5 + 0.25 = 00111110_2$$

(b)

$$2.71 \approx 2 + 0.5 + 0.25 = 00010110_2$$

$$\text{Fehler: } |2.71 - 2.75| = 0.04$$

(c)

$$5.375 = 4 + 1 + 0.25 + 0.125 = 00101011_2$$

(d)

$$9.12 \approx 8 + 1 + 0.125 = 01001001_2$$

$$\text{Fehler : } |9.12 - 9.125| = 0.005$$

2 Gleitkomma Darstellung

Vorzeichen in **rot**, Exponent in **grün**, Mantisse in **blau**.

(a)

$$17.75 = (-1)^0 \cdot (1 + 0.109375) \cdot 2^{131-127} = \text{0 10000011 000111000000000000000000}$$

(b)

$$3.625 = (-1)^0 \cdot (1 + 0,8125) \cdot 2^{128-127} = \text{0 10000000 110100000000000000000000}$$

3 Gleitkomma Interpretationen

(a) Addition:

$$2^{128-127} \cdot 1.0010111 + 2^{128-127} \cdot 1.1001011 = 2^{129-127} \cdot 1.0110001$$

Multiplikation:

$$2^{128-127} \cdot 1.0010111 \cdot 2^{128-127} \cdot 1.1001011 = 2^{129-127} \cdot 1.11011110111101$$

Division:

$$2^{128-127} \cdot 1.0010111 / 2^{128-127} \cdot 1.1001011 = 2^{127-127} \cdot 1.000000011$$

(b) Addition:

$$\begin{aligned} -1 \cdot 2^{131-127} \cdot 1.1011 + 2^{127-127} \cdot 1.000011 & \quad \text{Auf größeren Exponenten normalisieren} \\ -1 \cdot 2^{131-127} \cdot 1.1011 + 2^{131-127} \cdot 0.0001000011 & \quad = \\ -1 \cdot 2^{131-127} \cdot 1.11000001 & \end{aligned}$$

Multiplikation:

$$\begin{aligned} -1 \cdot 2^{131-127} \cdot 1.1011 \cdot 2^{127-127} \cdot 1.000011 & = \\ -1 \cdot 2^{131-127} \cdot 1.110001001 & \end{aligned}$$

Division:

$$\begin{aligned} -1 \cdot 2^{131-127} \cdot 1.1011 / 2^{127-127} \cdot 1.000011 & = \\ -1 \cdot 2^{131-127} \cdot 1.100111001 & \end{aligned}$$

(c) Addition:

$$\begin{aligned} -1 \cdot 2^{128-127} \cdot 1.1010011 + -1 \cdot 2^{133-127} \cdot 1.0101011 & \quad \text{Auf größeren Exponenten normalisieren} \\ -1 \cdot 2^{133-27} \cdot 0.000011010011 + -1 \cdot 2^{133-127} \cdot 1.0101011 & \quad = \\ -1 \cdot 2^{133-27} \cdot 1.01100011 & \end{aligned}$$

Multiplikation

$$\begin{aligned} -1 \cdot 2^{128-127} \cdot 1.1010011 \cdot -1 \cdot 2^{133-127} \cdot 1.0101011 & = \\ 2^{135-127} \cdot 1.0001101 & \end{aligned}$$

Division:

$$\begin{aligned} -1 \cdot 2^{128-127} \cdot 1.1010011 / -1 \cdot 2^{133-127} \cdot 1.0101011 & = \\ 2^{122} \cdot 1,001111 & \end{aligned}$$

4 Bitinterpretation

(a) Vorzeichen in **rot**, Exponent in **grün**, Mantisse in **blau**.

$$\begin{aligned} & 4496A000_{16} \\ = & 0100\ 0100\ 1001\ 0110\ 1010\ 0000\ 0000\ 0000 \\ = & \text{0 10001001 001011010100000000000000} \\ = & (-1)^0 \cdot \left(1 + \frac{1}{8} + \frac{1}{32} + \frac{1}{64} + \frac{1}{256} + \frac{1}{1024}\right) \cdot 2^{137-127} \\ = & 1205_{10} \end{aligned}$$

(b) Erste Zahl:

$$\begin{aligned} & 4496_{16} \\ = & 4 \cdot 16^3 + 4 \cdot 16^2 + 9 \cdot 16^1 + 6 \cdot 16^0 \\ = & 17558_{10} \end{aligned}$$

Zweite Zahl (wir betrachten die Zahl als vorzeichenlose Zahl):

$$\begin{aligned} & A000_{16} \\ = & 10 \cdot 16^3 + 0 \cdot 16^2 + 0 \cdot 16^1 + 0 \cdot 16^0 \\ = & 40960_{10} \end{aligned}$$

5 UTF8 Darstellung

Startsequenz in **grün**, Bytestartsequenz in **rot**, Daten in **blau**.

(a)

$$U+202E = \text{1110 0010 1000 0000 1010 1110}$$

(b)

$$\text{1111 0000 1001 1111 1001 1000 1000 1000} = U+1F608$$