

Aufbau

長大後

Signed

↑

0	00000000	00000000000000000000000000000000
---	----------	----------------------------------

$$\pm \text{Exp. (8)}$$

Mantisse (23)

$$255-127 =$$

128

$$\begin{array}{r} 255 \\ - 127 \\ \hline 128 \end{array}$$

2183

$$SignBit \times Mantisse \times 2^{Exponent - Bias}$$
$$(\underbrace{777 \dots 7}_{237}) \times 2^{183} \quad (1)$$

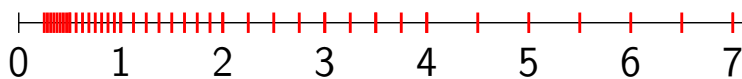
float: 127

-127

- ▶ Feste Anzahl an signifikanten Stellen
- ▶ Größerer Wertebereich als Fixkomma Zahlen
- ▶ Konsequenz: Höhere Genauigkeit bei kleinen Zahlen

任 + 0 = 任

x-2



70 0 m -

23-1

Aufbau

2^{20} 2^{32} 100^{+2}_{-2} 2^{2-2}

- ▶ Normalisiert: $1 \leq \text{Mantisse} < 2$ 反正都是1.几
- ▶ Führende Eins nicht abgespeichert 2就不有了
- ▶ Exponent(gespeichert) = Exponent(real) + Bias 为了负数
 - ▶ Bias statt Zweierkomplement
 - ▶ Lexikografischer Vergleich statt Subtraktion und Vergleich mit 0
 - ▶ In der Theorie weniger Operationen

$2^{31} 2$

$10110000011001111000000000000000$

指数 131 1966050 x 2.4

Datentypen float/double

$$2^{131-122} = 2^9$$

	Größe	Dezimalziffern	Abs. Min.	Abs. Max.
float (Single Prec.)	32 Bit	≈ 7	$\approx 1.18 \cdot 10^{-38}$	$\approx 3.4 \cdot 10^{38}$
<div><div>110001010</div><div>1101101010101010101010011</div></div> <div>± Exp. (8)Mantisse (23)</div>				
double (Double Prec.)	64 Bit	≈ 15	$\approx 10^{-308}$	$\approx 10^{308}$
<div><div>111110001010</div><div>11011010101010101010100111101101010101010011111111</div></div> <div>± Exp. (11)Mantisse (52)</div>				

1.1024

$$10011111 \times 2^{-6} \times 2^4$$

Quiz: Zahlen zuordnen 1

$(1.007777)_{10} \times 2^4$

26

= 133

Welchen Wert hat

0 10000100 010100000000000000000000 ?

$133 - 127 = 6$

↑

7. 7

[1.0101]

010000102

1.3125

$1.3125_{10} \times 2^{132}$

$= (10101)_2 \times 2^{-4}$

$= 21 \times 2^{-4}$

$127 + 6$

$= 133$



42.0₁₀

1.100100×2^6



0x42280000

$21 \times 2^{-4} \times 2^{132-127}$

$= 21 \times 2$

1.000000

1.000000

1.000000

Quiz: Zahlen zuordnen 2

Welchen Wert hat

1	01111111	00000000000000000000
---	----------	----------------------

☒ -1.0_{10}

☐ -127.0_{10}

☐ 0

Handwritten notes:

- $\log = \dots = 42$
- $100(000\dots)$
- $100(000\dots)$
- Mantisse:
- Mantisse x
- $-1.0^{127-127}$
- 真值: 0
- $100100\dots$

□ 康乃尔 实验

Welchen Wert hat

1	01111111	000000000000000000000000
---	----------	--------------------------


 ?

↓
Sign

Mantissa : [1.0]

Mantissa $\times 2^{\text{指数}}$

☒
$$-1.0_{10}$$

$$-127.0_{10}$$
$$-1.0^{127-127}$$


0

真相：口

110010016)00

Addition und Subtraktion

Handwritten notes in red ink:

~~Index~~
P
100 =
1.10.0100 = ... (b)
 $\times 2^{P-127}$
133
6

- ▶ Kleineren Wert auf selben Exponenten bringen wie großen Wert (denormalisieren)
- ▶ Mantissen addieren bzw. subtrahieren
- ▶ Mantisse entsprechend der Genauigkeit runden
- ▶ Ergebnis normalisieren

Subtraktion – Beispiel

$$\begin{array}{|c|c|c|} \hline + & 2^1 & 1.0000 \\ \hline \end{array} - \begin{array}{|c|c|c|} \hline + & 2^0 & 1.5000 \\ \hline \end{array}$$

① Gleicher Exponent 变

$$\begin{array}{|c|c|c|} \hline + & 2^1 & 1.0000 \\ \hline \end{array} - \begin{array}{|c|c|c|} \hline + & 2^1 & 0.7500 \\ \hline \end{array}$$

② Mantrisse subtrahieren 算

$$= \begin{array}{|c|c|c|} \hline + & 2^1 & 0.2500 \\ \hline \end{array}$$

③ Normalisieren 变

$$= \begin{array}{|c|c|c|} \hline + & 2^{-1} & 1.0000 \\ \hline \end{array}$$



在 1.2 间

Multiplikation und Division

- ▶ Exponenten addieren bzw. subtrahieren
- ▶ Mantissen multiplizieren bzw. dividieren (Führende 1 beachten)
- ▶ Mantisse entsprechend der Genauigkeit runden
- ▶ Ergebnis normalisieren

Probleme bei Genauigkeit

- ▶ Rundung: Ergebnis muss wieder FP-Darstellung gespeichert werden
 - ▶ Verschiedene Rundungsmodi, Standard: *round to nearest, ties to even*
- ▶ Absorption: Addition/Sub. von sehr großer und sehr kleiner Zahl
 - ▶ Keine Veränderung der großen Zahl wg. Rundung
 - ▶ Beispiel: $1000000.00f + 0.01f = 1000000.00f$
Handwritten notes: A red circle around the first 0 in 1000000.00f, a red wavy line under the first 0, and a black circle around 0.01f. To the right, handwritten text "12B 45" and "7B 5".
- ▶ Auslöschung: Subtraktion großer ähnlicher Zahlen
 - ▶ Subtraktion verstärkt Rundungsfehler
 - ▶ Beispiel: $1000000.1f - 1000000.0f = 0.125f \neq .1f$
 - ▶ Grund: $1000000.1f$ tatsächlich dargestellt als 1000000.125

Assoziativität und Distributivität

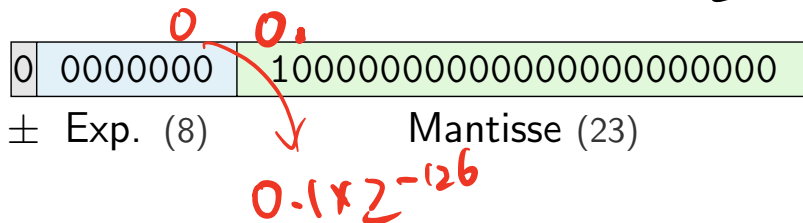
- ▶ Sowohl Addition und Multiplikation
- ▶ Nicht assoziativ
 - ▶ $(x + y) + z \neq x + (y + z)$
 - ▶ $(x \times y) \times z \neq x \times (y \times z)$
- ▶ Nicht distributiv
 - ▶ $x(y + z) \neq (xy) + (xz)$
- ▶ Achtung: -ffast-math (-Ofast) in GCC ignoriert diese zwecks Geschwindigkeit
- ▶ Weiterführend: What Every Computer Scientist Should Know About Floating-Point Arithmetic
https://www.itu.dk/~sestoft/bachelor/IEEE754_article.pdf

Denormale Zahlen / Subnormale Zahlen



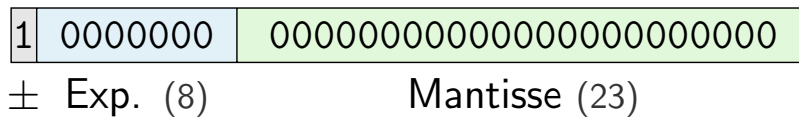
- ▶ Zahlen deren Exponent kleiner ist, als eine normalisierte Darstellung zulassen würde
- ▶ Beispiel, single precision, normalisiert: $1.0_2 \times 2^{-127}$ ~~x~~
- ▶ Denormalisiert: $0.1_2 \times 2^{-126}$
- ▶ Exponent hat speziellen Wert: alle Bits 0

最小是
值-126 !



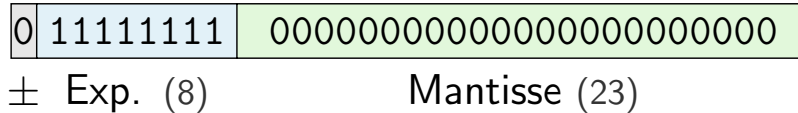
Null mit Vorzeichen

- ▶ Null: Exponent und Mantisse alle Bits 0
- ▶ Sign-Bit kann gesetzt sein $\rightarrow +/ - 0$ möglich
- ▶ Üblicherweise: $x + 0 = x$
- ▶ Sonderfall: $x = -0 \rightarrow -0 + 0 = +0$
- ▶ $-0 \neq +0$



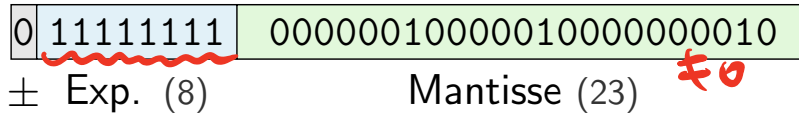
Unendlich / Infinity / ∞

- ▶ Alle Bits in Exponent = 1
- ▶ Alle Bits in Mantisse = 0
- ▶ → je nach Sign-Bit: $+/ -$ Unendlich
- ▶ z.B. Ergebnis bei $x/0$



Not a Number / NaN

- ▶ Alle Bits in Exponent = 1
- ▶ Mantisse $\neq 0$
- ▶ \rightarrow Not a Number
- ▶ z.B. Ergebnis bei $0/0$ und Unendlich - Unendlich
- ▶ $x \circ \text{NaN} = \text{NaN}$
- ▶ für jeden NaN Wert: $\text{NaN}_1 == \text{NaN}_2 \rightarrow \text{false}$



Quiz: Sonderfälle 1

Welches Ergebnis hat `NaN == NaN`?

☐

true

☒

false

☐

Segmentation Fault

Quiz: Sonderfälle 2

Welches Ergebnis hat $\text{NaN} \neq \text{Infinity}$?

☒

true

☐

false

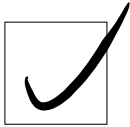
☒

1

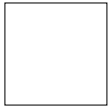
$!(\text{NaN} == \text{NaN})$
 $= !(0)$
 $= 1$

Quiz: Sonderfälle 3

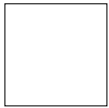
Welches Ergebnis hat $-\text{Infinity} < \text{Infinity}$?



true



false



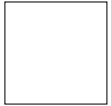
Arithmetic Exception: Invalid Operation

Quiz: Sonderfälle 4

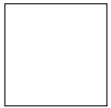
Welches Ergebnis hat $10 \neq \text{NaN}$?



true



false



Infinity

Quiz: Sonderfälle 5

Welches Ergebnis hat $5.0 / 0.0$?

☐

-Infinity

☐

NaN

☒

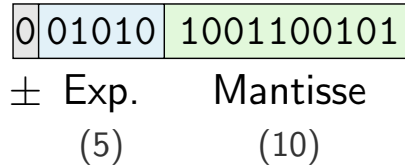
Infinity

☐

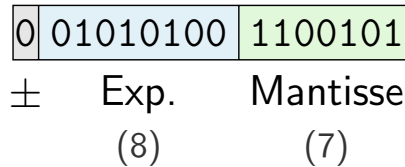
Arithmetic exception: Division by Zero

Weitere Floating Point Formate

- ▶ 16 Bit half precision / half



- ▶ Brain Floating Point / bfloat



- ▶ Extended Formate