Allgemeine Hinweise: Verwenden Sie für Ihre Lösungen keinen Taschenrechner und geben Sie die einzelnen Lösungsschritte an, sodass ersichtlich ist, wie Sie zu Ihren Lösungen gekommen sind. Sie selbst tun sich dann bei der Präsentation Ihrer Lösung umso leichter!
Bei den Tests dürfen keine elektronischen Hilfsmittel (Taschenrechner, Handy etc.) verwendet werden – nutzen Sie die Übung als Vorbereitung und verzichten Sie bereits jetzt darauf!

Aufgabe 1: Zahlenumwandlungen

Gegeben sind die folgenden Dezimalzahlen:

$$\mathrm{A} = (455.5625)_{10}$$
$$\mathrm{B} = (1162.51875)_{10}$$

Wandeln Sie die Zahlen A und B direkt in die folgenden Zahlensysteme um:

- Binärsystem
- Oktalsystem
- Hexadezimalsystem

Berechnen Sie Vor- und Nachkommateil getrennt. Rechnen Sie jeweils auf 5 Nachkommastellen genau, der Rest wird abgeschnitten. Geben Sie Ihre Berechnungen an!

Aufgabe 2: Zahlenumwandlungen

a) Wandeln Sie die Hexadezimalzahl $(3A5F1C.21A)_{16}$ in eine Binärzahl um.

b) Wandeln Sie die Oktalzahl $(27414.41237)_8$ in eine Hexadezimalzahl um.

c) Wandeln Sie die Binärzahl $(10100011011101.00101111001)_2$ in eine Oktalzahl um.

Aufgabe 3: Rechnen im Binärsystem

Es sind die folgenden Binärzahlen gegeben:

$$A = (111001110)_2$$
$$B = (11111001)_2$$
$$C = (11101)_2$$
$$D = (100)_2$$

Führen Sie mit diesen beiden Zahlen die folgenden Berechnungen binär(!) durch.
Geben Sie Ihre Berechnungen an!

a) Addition: $A + B$

b) Subtraktion: $A - B$

c) Division: A/C (Divisionsrest gegebenenfalls angeben!)

d) Multiplikation: $A * D$

Aufgabe 4: Zahlendarstellungen

Es sind folgende Zahlen gegeben:

$$\begin{aligned} A &= (10F)_{16} \\ B &= (-738)_{10} \\ C &= 0 \end{aligned}$$

Wandeln Sie die Zahlen A, B und C ins Binärsystem um und geben Sie anschließend alle entsprechenden 16 Bit langen Maschinenwörter in den folgenden Darstellungen an:

a) Darstellung durch Vorzeichen und Betrag

b) Einerkomplementdarstellung

c) Zweierkomplementdarstellung

d) Exzessdarstellung (Exzess = 2^{15})

Aufgabe 5: Rechnen in unterschiedlichen Zahlendarstellungen

Führen Sie mit den in den Teilaufgaben a) bis c) vorgegebenen Zahlen Z_1 und Z_2 jeweils die folgenden Rechenoperationen durch:

- $-Z_1$ (arithmetische Negation)
- $Z_1 + Z_2$ (Addition von Z_1 und Z_2)

Die binäre Codierung der Zahlen Z_1 und Z_2 entspricht bereits der im jeweiligen Unterpunkt angegebenen Darstellung. Die Länge des Maschinenwortes ergibt sich aus der Anzahl der vorgegebenen Binärziffern. Rechnen Sie binär und bleiben Sie, wenn möglich, in der jeweils angegebenen Zahlendarstellung. Falls Sie die Zahlendarstellung einmal verlassen müssen, geben Sie eine Begründung dafür an.

a) Zweierkomplementdarstellung: $Z_1 = (110001)_2$, $Z_2 = (001011)_2$

b) Darstellung durch Vorzeichen und Betrag: $Z_1 = (1010011)_2$, $Z_2 = (0010101)_2$

c) Exzessdarstellung mit Exzess = $(100)_2$: $Z_1 = (00001)_2$, $Z_2 = (00110)_2$

Aufgabe 6: Darstellung von Gleitpunktzahlen

Gegeben sind die folgenden Dezimalzahlen:

$$\text{A} = (511.1)_{10}$$
$$\text{B} = (\text{-}97.7578125)_{10}$$

a) Stellen Sie A und B im *Single Precision*-Format nach IEEE 754 für Gleitpunkt-Zahlensysteme dar (vgl. *Informatik Grundlagen*, 5. Auflage, S.130 ff).

   Hinweis: Es gelten die Vorschriften *implizites erstes Bit* und *optimale Rundung* in Form der Kombination *round to nearest/round away from zero*.

b) Stellen Sie A und B im folgenden 16 Bit-Gleitpunktformat dar:

$$\mathbb{F}(2, 11, -14, 15, true)$$

   Mit Ausnahme der kleineren Formatbreite ist dieses Gleitpunktformat ident zum IEEE 754 *single precision* Format aufgebaut. Verwenden Sie – wie bei IEEE 754 auch – Guard- und Round-Digit sowie das Sticky-Bit zur Vermeidung von numerischen Ungenauigkeiten.

Aufgabe 7: Darstellung von Gleitpunktzahlen

Gegeben ist das 16 Bit-Gleitpunktformat von Teilaufgabe 6 b).

a) Berechnen Sie den Wert der größten (x_{max}) und der kleinsten (x_{min}) in diesem Format darstellbaren positiven Zahl ($x_{max} < \infty$ und $x_{min} \neq 0$).

b) Geben Sie die Codierung der größten positiven Gleitpunktzahl ($\neq \infty$) an.

c) Geben Sie die Codierungen der kleinsten normalisierten und denormalisierten positiven Gleitpunktzahlen ($\neq 0$) an.

Kleinste positive normalisierte Zahl:

Kleinste positive denormalisierte Zahl:

d) Geben Sie die Darstellungen der Werte 0 (Null) und $-\infty$ an.

Null:

$-\infty$:

Aufgabe 8: Arithmetik auf Gleitpunktzahlen

Gegeben sind die Zahlen A, B und C im 16 Bit-Gleitpunktformat von Teilaufgabe 6 b):

$$\begin{aligned} \mathrm{A} &= 0\ 10011\ 1101010000 \\ \mathrm{B} &= 1\ 10001\ 1010100000 \\ \mathrm{C} &= 1\ 10011\ 1110110000 \end{aligned}$$

Führen Sie die nachfolgenden Berechnungen durch. Verwenden Sie Guard- und Round-Digit sowie das Sticky-Bit zur Vermeidung von numerischen Ungenauigkeiten (vgl. *Informatik Grundlagen*, 5. Auflage, S.145-157). Vergessen Sie nicht auf das implizite erste Bit!

a) $A + C$

b) $A * B$

c) $\dfrac{A}{C}$

Aufgabe 9: Gleitpunktrechnung

Gegeben ist das folgende Gleitpunktformat:

$$\begin{aligned} b &= 10 \\ p &= 5 \\ e_{min} &= -48 \\ e_{max} &= +49 \\ Exzess &= 49 \\ denorm &= false \end{aligned}$$

Berechnen Sie in diesem Gleitpunktformat $A + B$ und $A + C$ für die folgenden Zahlen:

$$A = 226.38 \qquad B = 0.125 \qquad C = 0.125003$$

Runden Sie ggf. mittels optimaler Rundung durch die Kombination *round to nearest/round away from zero*. Benutzen Sie dafür einen Additions-/Subtraktionsalgorithmus analog zu jenem von IEEE 754 (siehe *Informatik Grundlagen*, 5. Auflage, S.148). Stellen Sie für ausgewählte Rechenschritte Vorzeichen, Mantisse und Exponent inklusive der Werte von Guard-Digit, Round-Digit und Sticky-Bit dar.

Hinweis: Es wird in dieser Aufgabe nicht erwartet, dass Sie mit einer vollständig codierten Gleitpunktdarstellung rechnen. Sie dürfen stattdessen eine zur internen Codierung analoge Darstellung der Form $X.XXX \times 10^{YY}$ verwenden.

Aufgabe 10: Festpunkt-/Gleitpunktformat

Gegeben ist ein binäres 16 Bit-Festpunktformat mit Vorzeichenbit, 7 Vorkommastellen und 8 Nachkommastellen.

a) Geben Sie die Schrittweite und den dezimalen Wert der größten und der kleinsten darstellbaren positiven Zahl ($\neq 0$) an.
Hinweis: Es genügt, wenn Sie als Lösung einen aus Zweierpotenzen bestehenden Ausdruck angeben.

b) Geben Sie ein Gleitpunktformat $\mathbb{F}(2, p, e_{min}, e_{max}, false)$ mit möglichst kleiner Wortbreite an, das alle mit dem gegebenen Festpunktformat darstellbaren Zahlen ebenfalls darstellen kann.

c) Mit wievielen Bits könnte dieses minimale Format theoretisch implementiert werden? Ergeben sich in diesem Format Einschränkungen hinsichtlich der Codierung von Sonderwerten (0, ∞, ...)?