Übung 1 *Zahlendarstellung*
In der Vorlesung wurde die allgemeine Darstellung von Fließkommazahlen als $\mathbb{F}(\beta, r, s)$ vorgestellt. Dabei ist β die Basis, r die Anzahl der Stellen der Mantisse und s die Anzahl der Stellen des Exponenten.

a) Gegeben sei $x_0 = (0.5731 \times 10^5)_8 \in \mathbb{F}(8, 5, 1)$ in der oktalen Darstellung. Wie lautet diese Zahl in der normierten Fließkommadarstellung $\mathbb{F}(10, 5, 1)$ zur Basis 10?

b) Gegeben sei die reelle Zahl $x_1 = 0.3 \in \mathbb{R}$ in der Dezimaldarstellung. Bilden Sie x_1 in die normierten Fließkommadarstellung $\mathbb{F}(2, 11, 2)$ zur Basis 2 ab und danach wieder auf $\mathbb{F}(10, r, 1)$ ab. Für welche r bekommt man wieder 0.3?

c) Berechnen Sie $\max\limits_{x_2, x_3} |x_2 - x_3|$ in der Dezimaldarstellung, wobei $x_2 \in \mathbb{F}(4, 6, 2)$ und $x_3 \in \mathbb{F}(3, 7, 1)$.

d) Sei $x_4 \in \mathbb{F}(\beta, r, s)$ und es existieren $x_5, x_6 \in \mathbb{F}(\beta, r, s)$, der linke, resp. der rechte direkte Nachbarn von x_4 in $\mathbb{F}$. Dann sind beide Nachbarn genau gleich weit von x_4 entfernt, d.h. es gilt

$$|x_4 - x_5| = |x_4 - x_6|.$$

Beweisen Sie diese Behauptung oder widerlegen Sie sie durch ein Gegenbeispiel.

Sie dürfen einen Taschenrechner oder Computer für die Rechnungen benutzen. Achten Sie darauf wenn nötig richtig zu Runden (natürliche Rundung).

(2+3+3+2 Punkte)

Übung 2 *Richtig runden*
Zwei gängige Verfahren zum Runden von Zahlen sind das Aufrunden (natürliche Rundung) und die gerade Rundung. Wenn x eine auf r Stellen zu rundende Zahl ist und $\text{left}(x) = \max\{y \in \mathbb{F} \mid y \leq x\}$ sowie $\text{right}(x) = \min\{y \in \mathbb{F} \mid y \geq x\}$ dann gilt beim Aufrunden:

$$rd(x) = \begin{cases} \text{left}(x) & \text{falls} \quad 0 \leq m_{r+1} < \beta/2 \\ \text{right}(x) & \text{falls} \quad \beta/2 \leq m_{r+1} < \beta \end{cases}$$

Beim geraden Runden ist dagegen:

$$rd(x) = \begin{cases} \text{left}(x) & \text{falls} \quad (|x - \text{left}(x)| < |x - \text{right}(x)|) \vee \\ & \qquad\quad (|x - \text{left}(x)| = |x - \text{right}(x)| \wedge m_r \text{ gerade}) \\ \text{right}(x) & \text{sonst} \end{cases}$$

Dabei ist m_i jeweils die i-te Nachkommastelle von x.
Berechnen Sie die Folge von Fließkommazahlen

$$\begin{aligned} x_0 &:= x \\ x_n &:= (x_{n-1} \ominus y) \oplus y \end{aligned}$$

mit $x = 2.46$ und $y = -0.755$. Dabei seien x, x_i und y Fließkommazahlen in der Darstellung $\mathbb{F}(10, 3, 1)$ und die Fließkommaoperationen $\oplus, \ominus$ stellen exakte Arithmetik mit Runden dar

$$x \oplus y = rd(x + y), \quad x \ominus y = rd(x - y).$$

Welche Ergebnisse erhält man für die ersten 10 Folgenglieder mit natürlicher Rundung bzw. mit gerader Rundung?

(5 Punkte)

Übung 3 *Numerische Nulladdition (Praktische Übung)*
In Gleitkommaarithmetik können Gleichungen mehr Lösungen haben als mit exakter Arithmetik. Beispielsweise hat die Gleichung $(1+x) = 1$ für $x \in \mathbb{R}$ und exakter Arithmetik nur die Lösung $x = 0$. In Gleitkommaarithmetik hat diese Gleichung in der Regel neben $x = 0$ noch weitere Lösungen, nämlich alle Zahlen, die zu klein sind um bei der Summe noch einen Effekt zu erzeugen.

In dieser Aufgabe sollen Sie sich mit den Standard-Fließkommatypen in C++ vertraut machen. Schreiben Sie ein Programm, welches eine Fließkommazahl `x` vom Typ `float` bzw. `double` über die Standardeingabe (`std::cin`) einliest. Berechnen sie dann

$$x = x + 1.$$

a) Wie klein muss `x` gewählt werden, damit die Ausgabe (via `std::cout`) wieder exakt 1 zurückgibt? Hinweis: Denken Sie daran die Ausgabegenauigkeit von `std::cout` entsprechend einzustellen. Verwenden Sie dafür `std::setw` und `std::setprecision` (suchen Sie ggf. im Internet nach der korrekten Verwendung).

b) Ist dies die kleinste positive Zahl, die der Typ `float` bzw. `double` darstellen kann?

(5 Punkte)