
Physiker sind von Aufgabe 1 befreit, sollten aber den Stoff beherrschen!

Aufgabe 1:**2-Komplement**

(2 + 3 + 2 Punkte)

Wenden Sie die Konstruktion des 2-Komplements auf 8-Bit-Darstellungen $a_7a_6 \dots a_1a_0$ von Zahlen an, d.h. a_7 ist das Vorzeichenbit und $a_6 \dots a_1a_0$ codiert einen Wert zwischen 0 und 127. So codiert der String 01001101 die Zahl $z_1 = 77$.

- a) Bestimmen Sie die Codierungen für die negativen Zahlen $z_2 = -55$ und $z_3 = -100$.
- b) Führen Sie die Additionen $z_1 + z_2$, $z_1 + z_3$ und $z_2 + z_3$ in den Binärdarstellungen aus und überprüfen Sie die Korrektheit der Ergebnisse.
- c) Stellen Sie eine Vermutung auf, in welchen Situationen ein Überlauf nach der Addition der führenden Bits (also der Vorzeichenbits) auftritt, obwohl das Ergebnis im Darstellungsbereich liegt und versuchen Sie, dafür eine Begründung zu geben.

Aufgabe 2:**Datentyp Char**

(2 + 3 + 3 Punkte)

a) Eine bei Laien beliebte, für geheime Nachrichten keinesfalls zu empfehlende, Methode zur Verschlüsselung von Texten basiert auf der Vertauschung von Buchstaben. Die folgende Verschlüsselung soll schon Cäsar verwendet haben: Man wählt sich eine beliebige, geheime Zahl k und ersetzt jeden Kleinbuchstaben des Alphabets durch den k -ten Nachfolger im Alphabet, wobei durch Zyklenschluss 'a' als Nachfolger von 'z' festgelegt wird. Implementieren Sie die Funktion `caesar :: Int -> Char -> Char`, die bei Eingabe einer Zahl k und eines Kleinbuchstabens, den k -ten zyklischen Nachfolger bestimmt und an allen anderen Symbolen aus `Char` keine Änderungen vornimmt.

b) Implementieren Sie eine Methode `reflectLetter :: Char -> Char`, die wie folgt auf allen Groß- und Kleinbuchstaben operiert: $a \mapsto Z$, $b \mapsto Y$, ..., $z \mapsto A$, $A \mapsto z$, ..., $Z \mapsto a$. Für alle anderen Zeichen soll die Fehlermeldung "out of range" ausgegeben werden.

c) Wir führen die folgende Ordnungsrelation auf den (Groß- und Klein-) Buchstaben des Alphabets ein: $a < A < b < B < c < C < d < \dots$. Implementieren Sie eine Funktion `newOrder :: Char -> Char -> Bool`, die für zwei Buchstaben `True` ausgibt, wenn beide gleich sind oder wenn der erste nach obiger Ordnung kleiner als der zweite ist. Andernfalls (auch wenn einer der Eingabewerte kein Buchstabe ist) soll `False` ausgegeben werden.

Aufgabe 3:**Euklidischer Algorithmus**

(2 + 3 + 2 Punkte)

Aus der Vorlesung kennen wir den Fakt, dass für zwei beliebige ganze Zahlen a und b ganze Zahlen s und t existieren, so dass $ggT(a, b) = s \cdot a + t \cdot b$ erfüllt ist, und eine Methode zur Berechnung von s und t .

- a) Wenden Sie diese Methode an, um die Koeffizienten s und t für die Zahlen $a = 130$

und $b = 46$ zu bestimmen.

b) Seien $a, b, c \in \mathbb{Z}$ Zahlen mit der Eigenschaft, dass es keine ganze Zahl $d > 1$ gibt, die alle drei Zahlen teilt. Zeigen Sie, dass dann ganze Zahlen s, t, u existieren, so dass $1 = sa + tb + uc$, und erläutern Sie, wie man diese Werte bestimmen kann.

c) Wenden Sie Ihre in b) beschriebene Methode zur Berechnung von s, t und u für $a = 12, b = 20$ und $c = 75$ an.