

Übungsserie 4b

Programmierung mit Common-Lisp

Bitte beachten Sie den Abgabezeitpunkt in Moodle. Serien per E-Mail oder verspätete Abgaben können nicht gewertet werden. Die Textaufgaben bitte per PDF mit den Programmieraufgaben zusammen in ein Zip-Archiv bündeln.

Aufgabe 1 (5 Punkte) Implementieren Sie in einer Datei `binom.cl` eine Funktion zur Berechnung der Fakultät einer natürlichen Zahl n , sowie eine darauf aufbauende Funktion zur Bestimmung des Binomialkoeffizienten zweier natürlicher Zahlen a und b .

Testen Sie beide Methoden mit jeweils mindestens zwei Beispielen. Für welche a arbeitet Ihr Programm, bei gültigem b , noch korrekt? Was können Sie entsprechend über die Rechengenauigkeit sagen und vergleichen Sie dies insbesondere mit dem Wertebereich von vorzeichenlosen Ganzzahlen in 64Bit. Geben Sie bitte Ihre Antwort in Kommentarform ab.

Aufgabe 2 (8 Punkte) Die Cäsar-Verschlüsselung ist eine einfache Verschiebechiffre. Alle Buchstaben eines Eingabetextes werden dabei eindeutig auf einen Geheimbuchstaben abgebildet. Das Geheimalphabet entsteht durch die zyklische Verschiebung der geordneten Buchstaben des Eingabealphabetes um eine feste Anzahl an Stellen. Diese Anzahl der Verschiebungen stellt somit den Schlüssel zum Kodieren und Dekodieren von Eingaben dar.

Schreiben Sie ein Programm `caesar.cl`, welches die Cäsar-Chiffrierung durchführt und dabei Zeichen für Zeichen von der Standardeingabe liest und chiffriert auf der Standardausgabe wieder ausgibt. Dabei soll das Programm alle ASCII-Buchstaben [A-Z] sowie [a-z] mit einem festen Schlüssel verschieben. Alle Großbuchstaben werden wieder auf Großbuchstaben abgebildet und alle Kleinbuchstaben wieder auf Kleinbuchstaben verschoben. Zeichen, welche keine Buchstaben ([A-Za-z]) sind, bleiben hierbei intakt und werden nicht verschoben. Wenn eine Verschiebung über das Ende des Alphabets gehen würde, wird sie am Anfang fortgesetzt (zyklisch). Übergeben Sie der Funktion den Schlüssel optional. Andernfalls nutzen Sie 13 als voreingestellten Schlüssel. Testen Sie auch andere Schlüssel: 0, 26, oder 1.

Beispiel: Für den Verschiebeschlüssel 13 und der Zeichenkette: "Hello World!" ergibt sich demnach: "Uryyb Jbeyq!". Der Verschiebeschlüssel ist im Programmcode festgelegt. Testen Sie Ihr Programm: `echo "Hello World." | clisp caesar.cl`

Aufgabe 3 (6 Punkte) Schreiben Sie zunächst ein Programm `quer.cl`, welches die Quersumme einer Zahl n berechnet. Dabei sei n als Integer gegeben, nicht als Liste von Ziffern.

Eine Verallgemeinerung sind gewichtete Quersummen, bei denen die Ziffern erst mit den Werten einer periodischen Zahlenfolge multipliziert und diese Ergebnisse aufsummiert werden. Dabei wird mit der niedrigstwertigen Ziffer begonnen. Implementieren Sie dies ebenfalls. Hierbei sei die Folge:

1, 3, 2, -1, -3, -2, 1, 3, 2, -1, -3, -2, ...

Dann ist die gewichtete Quersumme für bspw. der Zahl 1234567:

$$7 \cdot 1 + 6 \cdot 3 + 5 \cdot 2 + 4 \cdot (-1) + 3 \cdot (-3) + 2 \cdot (-2) + 1 \cdot 1 = 7 + 18 + 10 - 4 - 9 - 4 + 1 = 19$$

Erweitern Sie Ihr Programm so, dass es auch die oben gezeigte gewichtete Quersumme einer Zahl berechnen kann. Testen Sie ihre Funktionen und geben Sie Beispiele mit an.