

1. Aufgabe (3 Punkte)

In Haskell ist es möglich namenlose Funktionen (anonyme Funktionen) zu verwenden. Recherchieren Sie, wie dies möglich ist und nennen Sie sinnvolle Anwendungsmöglichkeiten. Geben Sie ein Beispiel an. Beschreiben Sie bitte ihren Rechercheweg und geben Sie die Quellen an.

2. Aufgabe (3 Punkte)

Erzeugen Sie im Terminal die Listen `[3.0..7.3]`, `[3.0..7.6]`, `[3.1..7.5]`, `[3.1..7.6]`, `[3.1, 5.1..8.1]`. Beschreiben Sie das Verhalten und verallgemeinern Sie Ihre Erkenntnis auf die besondere Beziehung zwischen Startwert, Zielwert und Schrittweite. Geben Sie gegebenenfalls Ihre Quellen an.

3. Aufgabe (7 Punkte)

Die schwache Goldbachsche Vermutung besagt, dass jede ungerade Zahl, die größer als 5 ist, als die Summe von drei Primzahlen geschrieben werden kann. Diese drei Primzahlen werden als schwaches Goldbachsches Triple bezeichnet.

1. Definieren Sie eine Funktion **schwacheGoldbachTriple**, welches bei Eingabe einer ungeraden Zahl die Liste aller Schwachen Goldbachschen Triple berechnet.

Beispiel: `schwacheGoldbachTriple 19` wird zu `[(3,3,13), (3,5,11), (5,7,7)]` ausgewertet.

Hinweis: Verwenden sie die Erzeugung von Primzahlen aus der Vorlesung.

2. Definieren Sie eine Funktion **testGoldbachscheVermutung**, welche bei der Eingabe einer Zahl prüft, ob die Vermutung bis zu dieser Zahl zutrifft.

Beispiel: `testGoldbachscheVermutung 300` liefert das Ergebnis `True`.

4. Aufgabe (5 Punkte)

Aus der Vorlesung ist die Funktion `(++) :: [a] -> [a] -> [a]` bereits bekannt. Implementieren Sie die Funktion `plusPlus` mit dem gleichen Verhalten mittels einer **geeigneten Faltung**. Testen Sie Ihre Funktion.

Beispiel: `plusPlus [1,2] [2,3]` wird zu `[1,2,2,3]` ausgewertet.

5. Aufgabe (5 Punkte)

Schreiben Sie eine Funktion `machGross :: String -> String`, welche eine Zeichenkette erhält und alle Zeichen in Großbuchstaben zurückgibt. Lösen Sie die Aufgabe mittels Faltung. Sie dürfen nur Funktionen für den Datentyp `Char`, aber nicht für `String` verwenden.

6. Aufgabe (7 Punkte)

In der Vorlesung haben wir händisch `foldr (*) 1 [1..4]` ausgewertet. Betrachten Sie nun die Faltung nach links. Werten Sie händisch `foldl (*) 1 [1..4]` (Achtung: linke Faltung!) aus. Sie dürfen Zwischenergebnisse bereits auswerten, falls möglich.

Welche der beiden Varianten würden Sie bevorzugen um das Produkt einer Liste von Zahlen zu berechnen? Begründen Sie Ihre Antwort.

Wichtige Hinweise:

- 1) Verwenden Sie geeignete Namen für Ihre Variablen und Funktionsnamen, die den semantischen Inhalt der Variablen oder die Semantik der Funktionen wiedergeben.
- 2) Verwenden Sie vorgegebene Funktionsnamen, falls diese angegeben werden.
- 3) Kommentieren Sie Ihre Programme.
- 4) Verwenden Sie geeignete lokale Funktionen und Hilfsfunktionen in Ihren Funktionsdefinitionen.
- 5) Schreiben Sie in alle Funktionen die entsprechende Signatur.