



Aufgabe 8-1 (ABGABE!)

- a) Schreiben Sie eine Funktion `groupByResult`, die alle Elemente einer Liste nach dem Ergebnis einer gegebenen Funktion gruppiert:
- ```
groupByResult (**2) [-1.0, 2.0, 1.0, 3.0, -3.0] =
 [[-1.0, 1.0], [2.0], [-3.0, 3.0]]
```
- b) Definieren Sie folgende Funktion, indem Sie `foldl` oder `foldr` verwenden: `unwordsWith`, Gegenstück zu `words`. Erzeugt aus einer Liste von Wörtern einen String, der die Wörter mit einem gegebenen Separator zusammenfügt. Z.B.:
- ```
unwordsWith '+' $ words "Haskell Serie 8" = "Haskell+Serie+8"
```

Aufg. 8.1 bitte bis 26.11. 23:59 Uhr in Moodle hochladen! Max. 2 Punkte!

Dateiname: Serie8.hs, als erste Zeile fügen Sie bitte ein: `module Serie8 where`

Aufgabe 8-2 (Prädikate)

- a) Schreiben Sie eine Funktion
- ```
largestMatchingElement :: Enum a => a -> (a -> Bool) -> a
```
- welche eine Zahl `a` und eine Prädikat erhält und die größte Zahl kleiner oder gleich `a` zurückgibt, die das Prädikat erfüllt. Geben Sie außerdem 3 Funktionsaufrufe mit den jeweils übergebenen Werten und erhaltenen Ergebnissen an.
- b) Nutzen Sie `takeWhile`, um zwei Varianten von Ausdrücken zu schreiben, welche die Summe der geraden Quadratzahlen zurückgeben, die kleiner als 100.000 sind.
- Benutzen Sie für den ersten Ausdruck `filter` und/oder `map` und für den zweiten
  - List-Comprehension.

### Aufgabe 8-3 (Funktionen höherer Ordnung)

Definieren Sie folgende Funktionen, indem Sie `foldl` oder `foldr` verwenden.

- a) `myAll` überprüft ob alle Elemente einer Liste ein gegebenes Prädikat erfüllen.
- b) `unwordsWithSeperator`, Gegenstück zu `words`. Erzeugt aus einer Liste von Wörtern einen String, der die Wörter mit einem gegebenen Separator zusammenfügt. Z.B.:
- ```
unwordsWithSeperator '+' $ words "Haskell Serie 7" = "Haskell+Serie+7"
```



Aufgabe 8-4 (IO-Monade)

Schreiben Sie einen einfachen Chatbot. Dieser soll fragen, wie es dem Benutzer geht und danach folgendermaßen vorgehen:

1. Liest die Antwort des Benutzers ein
2. Hängt ein Fragezeichen an das letzte Wort der Benutzerantwort an und gibt es aus
3. Liest Antwort des Benutzers ein
4. Fragt: „Warum?“
5. Geht zurück zu Punkt 1.

Ein möglicher Gesprächsablauf könnte folgendermaßen aussehen:

Bot: Wie geht es dir?

User: Mir geht es gut

Bot: gut?

User: Ja

Bot: Warum?

User: Meine Oma hat mir eine Mütze gehäkelt

Bot: gehäkelt?

User: Ja, weil sie mich so gern hat

Bot: Warum?