

1. Übungsblatt

Ausgabe: 25.10.16

Abgabe: 04.11.16

1.1 Vokalitäten

7 Punkte

Zwei Zeichenketten sind gleich, wenn sie die gleichen Zeichen in der gleichen Reihenfolge enthalten — soweit, so einfach. In dieser Aufgabe betrachten wir ähnliche Zeichenketten: zwei Zeichenketten seien *vokalgleich*, wenn sie die gleichen Vokale in derselben Reihenfolge enthalten; alle anderen Zeichen sind irrelevant, genauso wie Groß/Kleinschreibung. So sind beispielsweise die Zeichenketten "Apfel" und "Haskell" vokalgleich, aber "Banane" und "Apfelmus" nicht.

Implementieren Sie eine Funktion

```
vowelEq :: String → String → Bool
```

welche prüft, ob die zwei Argumente vokalgleich sind.

1.2 Psst, geheim!

13 Punkte

In diesen Zeiten kann man einfach nicht wachsam genug sein. Das finden auch diverse Nachrichtendienste, und lesen deshalb vorsichtshalber unsere elektronische Post mit. Deshalb wollen wir unsere Nachrichten verschlüsseln.

Die hier betrachtete Verschlüsselungstechnik ist nicht vollständig sicher, erfreut sich dafür aber in den Kindergärten der Republik großer Beliebtheit. Sie funktioniert sie ganz einfach: nach jedem Vokal wird der Buchstabe 'p' angehängt, und der Vokal wiederholt. Aus "Apfelmus" wird somit "Apapfepelmupus".

Implementieren Sie zwei Funktionen

```
encrypt :: String → String
```

```
decrypt :: String → String
```

die eine Zeichenkette auf diese Art und Weise ver- und wieder entschlüsseln. Die Entschlüsselung nicht verschlüsselter Worte sollte im Normalfall das Wort nicht ändern (z.B. `decrypt "Rhabarber"`), außer wenn das Wort zufällig ein 'p' zwischen zwei gleichen Vokalen enthält (z.B. `decrypt "Topologie"`).

Hinweis: Folgende vordefinierte Funktionen können hilfreich sein:

```
null :: String → Bool
```

```
head :: String → Char
```

```
tail :: String → String
```

```
(:) :: Char → String → String
```

```
toLower :: Char → Char
```

Die letzte davon muss mittels einer magischen `import Data.Char` Inkantation am Anfang der Datei importiert werden.

? *Verständnisfragen*

Auf diesem und den folgenden Übungsblättern finden sich Verständnisfragen zur Vorlesung. Diese sind nicht Bestandteil der Abgabe, können aber im Fachgespräch thematisiert werden. Wenn Sie das Gefühl haben, diese Fragen nicht sicher beantworten zu können, wenden Sie sich gerne an Ihren Tutor oder an den Dozenten.

1. Was bedeutet Striktheit, und welche in Haskell definierbaren Funktionen haben diese Eigenschaft?
2. Was ist ein algebraischer Datentyp, und was ist ein Konstruktor?
3. Was sind die drei Eigenschaften, welche die Konstruktoren eines algebraischen Datentyps auszeichnen, was ermöglichen sie und warum?