

Name:

Matrikelnummer:

Tutor:

	Aufgabe 1	Aufgabe 2	Aufgabe 3	Aufgabe 4	Gesamt
Punkte	/8	/8	/7	/9	/32

Die Lösung der Aufgaben sollte immer auf dem entsprechenden Aufgabenzettel, dessen Rückseite oder auf einem Zusatzblatt (Namen nicht vergessen!) zu finden sein. Im diesem Fall bitte auf dem Aufgabenblatt einen Hinweis auf das Zusatzblatt.

Es sind keine Hilfsmittel erlaubt!

Aufgabe 1:**Logik**

2 + 4 + 2 Punkte

Wir bezeichnen mit dem Symbol \downarrow die Boolesche Operation NOR,

d.h. $x \downarrow y \equiv \neg(x \vee y)$.

- Zeigen Sie, dass die Operation \downarrow nicht assoziativ ist.
- Bestimmen Sie die kanonische KNF für die Formel $(x_1 \downarrow x_2) \downarrow (x_2 \downarrow x_3)$.
- Vereinfachen Sie diese KNF, falls das möglich ist.

Name:

Aufgabe 2:**Listen und Induktion**

4 + 4 Punkte

Die Funktionen `ping`, `pong` :: [Int] -> Int sind wie folgt definiert:

`ping [] = 0` -- (1)

`ping (x:xs) = (pong xs) - x` -- (2)

`pong [] = 0` -- (3)

`pong (x:xs) = (ping xs) + x` -- (4)

Achtung: Bitte genau hinsehen, `ping` ruft `pong` und `pong` ruft `ping` auf!

a) Beweisen Sie mit vollständiger Induktion bezüglich der Länge einer Liste `xs`, dass `ping xs` den gleichen Wert wie `-(pong xs)` berechnet. Die Zeilennummerierung in der Definition kann zur Begründung der Schritte herangezogen werden.

b) Wie ist die Funktion `foldr1` definiert? Bitte Signatur und Definition angeben. Beweisen Sie mit vollständiger Induktion bezüglich der Listenlänge, dass für nichtleere Listen `xs` der Aufruf `foldr1 (-) xs` den gleichen Wert berechnet wie `pong xs`. Sie können und sollten die Aussage aus Teil a) als Voraussetzung nutzen.

Name:

Aufgabe 3: Klassen und Instanzen 4 + 3 Punkte

Durch folgende Datentyp `Figur` kann man ein Rechteck durch seine zwei Seitenlängen konstruieren, aber es gibt auch einen Konstruktor für den Spezialfall eines Quadrats bei dem natürlich nur eine Seitenlänge angegeben wird.

```
data Figur = Rechteck Float Float | Quadrat Float
```

- a) Implementieren Sie `Figur` als Instanz der Klasse `Eq`, wobei zwei Figuren als gleich betrachtet werden, wenn man sie (auch mit Drehungen) zur Deckung bringen kann.
 - b) Implementieren Sie `Figur` als Instanz der Klasse `Ord`, wobei die Fläche als primäres Vergleichskriterium herangezogen wird, bei gleicher Fläche entscheidet der Umfang.
- Hinweis: Definieren Sie Funktionen `flaeche, umfang :: Figur -> Float`.

Name:

Aufgabe 4: Präfixcodes und Huffman-Algorithmus 4 + 1 + 4 Punkte

a) Für das Alphabet der 26 Kleinbuchstaben a, b, c, \dots, x, y, z sollen binäre Präfixcodes mit bestimmten Längenbedingungen gefunden werden. Entscheiden und begründen Sie unter welcher der folgenden 4 Bedingungen ein solcher Präfixcode existiert:

- a1) Alle ganzen Zahlen von 1 bis 26 treten als Codewortlänge auf.
- a2) Alle Codewörter haben die Länge 5.
- a3) Genau die Hälfte der Codewörter hat die Länge 4 und die andere Hälfte der Codewörter hat die Länge 6.
- a4) Zwölf Codewörter haben die Länge 4, zwei Codewörter haben die Länge 5 und zwölf Codewörter haben die Länge 6.

b) Bei welchen der positiven Antworten aus Teil a) kann der Präfixcode nicht optimal sein (d.h. für keine Wahrscheinlichkeitsverteilung)?

c) Für das Alphabet $A = \{a, b, c, d, e\}$ ist die folgende Wahrscheinlichkeitsverteilung vorgegeben:

$$p(a) = 0.25, p(b) = p(c) = 0.1, p(d) = 0.15 \text{ und } p(e) = 0.4.$$

Bestimmen Sie einen optimalen Code für (A, p) .