

Informatik A, WS 2016/17 — 6. Übungsblatt

Abgabe bis Freitag, 25. November 2016, 12:00 Uhr, in die Fächer der Tutor/inn/en

32. Resolutionskalkül, 0 Punkte

- (a) Wie viele verschiedene mögliche Klauseln kann man mit n Variablen bilden?
- (b) Wie viele Klauseln können bei Anwendung des Resolutionskalküls maximal entstehen, wenn es n Variablen gibt und alle Klauseln die Länge 2 haben?
- (c) Sei nun $n \geq 3$. Geben Sie Beispiele für eine Menge von Klauseln der Länge 3, für die der Resolutionskalkül eine Klausel erzeugt, die alle n Variablen enthält.

33. Ausdrücke mit Variablen, Programmieraufgabe, 11 Punkte

- (a) Erweitern Sie die Funktion `rechneAus` aus der Vorlesung um einen zweiten Parameter vom Typ `[(String,Integer)]`, der eine Liste von Variablennamen mit zugehörigen Werten enthält. Beispiel:

```
rechneAus ((3+Var "c")*(4+Var "b")) [("a",5),("b",-5),("c",3)]=-6
```

Variablen, die nicht in der Liste vorkommen, sollen als 0 betrachtet werden.

- (b) Schreiben Sie dafür zum Bestimmen der Wertes einer Variablen eine polymorphe Hilfsfunktion `schaueNach :: Eq a => a -> [(a,b)] -> Maybe b`. Beispiele:
`schaueNach "c" [("a",5),("b",-5),("c",3)] = Just 3`
`schaueNach "d" [("a",5),("b",-5),("c",3)] = Nothing`

34. Textverarbeitung, Programmieraufgabe, 10 Punkte

Schreiben Sie eine Funktion `ersetze :: String -> String -> String -> String`, sodass `ersetze alt neu s` das erste Vorkommen von `alt` in der Zeichenkette `s` durch `neu` ersetzt. Beispiel: `ersetze "ich" "du" "sicherlich" = "sduerlich"`. Wenn `alt` nicht in `s` vorkommt, soll das Ergebnis `s` sein.

35. Uhrzeit, Programmieraufgabe, 0 Punkte

Machen Sie den Typ `data Uhrzeit = Zeit Int Int` aus der Vorlesung zu einem Exemplar der Typklasse `Show` mit einer eigenem Funktion `show`, die die Uhrzeit in der Form `"x:y Uhr"` ausgibt. Die Minuten sollen immer zweistellig erscheinen.

36. Klammern bei arithmetischen Ausdrücken, 0 Punkte

- (a) Die Funktion `drucke` aus der Vorlesung geht beim Anzeigen eines arithmetischen Ausdrucks auf Nummer sicher, indem sie um jeden Ausdruck eine Klammer setzt. Gibt es trotzdem die Möglichkeit, den Benutzer zu täuschen und einen Ausdruck zu schreiben, der nicht seine echte Struktur wiedergibt?
- (b) Schreiben Sie eine modifizierte Funktion `druckeKnapp :: Ausdruck -> String`, die alle überflüssigen Klammern weglässt, z.B. `1*2-3+4*5/(6*7-8)-9`

37. Vollkommene Zahlen, Listendurchlauf, Programmieraufgabe, 9 Punkte

- (a) Schreiben Sie eine Funktion `teiler`, die die Liste der positiven Teiler einer Zahl in irgendeiner Reihenfolge bestimmt. Beispiel: `teiler 45 = [45,3,15,1,9,5]`.
- (b) Bestimmen Sie jede Zahl n zwischen 1 und 1000, bei der die Summe ihrer von n verschiedenen Teiler größer ist als die Zahl n selbst.
(Hinweis: Die Funktion `sum` berechnet die Summe der Elemente einer Liste)
- (c) Bestimmen Sie alle *vollkommenen Zahlen* zwischen 1 und 1000: die Zahlen n , die gleich der Summe ihrer von n verschiedenen Teiler sind.