

Algorithmen und Programmierung I

WS 2004 / 2005

Übung 12 (25 Punkte)

Ausgabe: Mo. 24.1.2005

Abgabe: Mi. 2.2.2005, 10.00

Verspätet abgegebene Lösungen werden nicht akzeptiert.

Die Abgabe erfolgt in Zweiergruppen. Die Gruppe darf nur mit ausdrücklicher Genehmigung des Tutors gewechselt werden.

Abgegeben werden müssen

- alle Lösungen einmal ausgedruckt. **Handschriftliche Lösungen werden nicht akzeptiert.**
- Programme, Testdaten, Testergebnisse in elektronischer Form in Absprache mit den Tutorinnen und Tutoren.

Aufgabe 12.1 (5 Punkte)

Reduzieren Sie die folgenden Lambda-Ausdrücke. Manche Klammern sind nicht notwendig, andere, die nicht notwendig erscheinen, aber schon. Das Lambda-Zeichen ist hier ein `\` ("backslash").

- $((\lambda x. \lambda y. (y \ x)) (\lambda p. \lambda q. p)) (\lambda i. i)$
- $(((((\lambda x. \lambda y. \lambda z. (x \ y)) z)) (\lambda f. \lambda a. (f \ a))) (\lambda i. i)) (\lambda j. j))$
- $(\lambda h. (((\lambda a. \lambda f. (f \ a)) h) h) (\lambda f. (f \ f)))$
- $((((\lambda f. \lambda g. \lambda x. (f \ (g \ x))) (\lambda s. (s \ s))) (\lambda a. \lambda b. b)) (\lambda x. \lambda y. x))$
- $((\lambda p. ((\lambda q. (p \ q)) ((\lambda x. x) (\lambda a. \lambda b. a)))) (\lambda k. k))$

Aufgabe 12.2 (12 Punkte)

a) Gesucht ist eine Haskell-Funktion `isSyntax :: String -> Bool`, die feststellt, ob die Eingabezeichenkette ein syntaktisch korrekter λ -Ausdruck ist. λ wird durch das Zeichen `\` dargestellt. Namen sind Zeichen aus einem Buchstaben (a, b, ..x,y,z), Jeder Ausdruck kann geklammert oder ungeklammert sein. Es ist sinnvoll, die Eingabe zunächst zu normieren, z.B. Entfernen führender und mehrfacher Leerzeichen.

b) Schreiben Sie eine Funktion, die für einen syntaktisch korrekten λ -Ausdruck die freien bzw. die gebundenen Variablen bestimmt: `freeBoundVar :: String -> [[String]]`

Aufgabe 12.3 (8 Punkte)

a) Definieren Sie die Funktionen "kleiner als" und "größer als" für zwei Zahlen.

b) Die Wahrheitswerte True und False definiert man $\lambda x \ y. x$ (Projektion auf die erste Komponente) bzw. $\lambda x \ y. y$. Gehen Sie von den Definitionen für and, or und not im λ -Kalkül aus (siehe Tutorial von R.Rojas) und definieren Sie die Funktionen XOR und wenden sie die auf alle vier möglichen Argumentpaare (True, True; ..., False, False) an.