

Grundlagen der Programmierung 2

Sommersemester 2018

Aufgabenblatt Nr. 2

Abgabe: Mittwoch 25. April 2018 **vor!** der Vorlesung

Aufgabe 1 (20 Punkte)

Die Funktionen f_1 bis f_5 seien in Haskell definiert als:

```
f1 x = f2 (x*13) - f4 (f1 x)
f2 x = if (f1 (x-1) < 30) then f4 (x*x) else f4 (x+7)
f3 x = if x /= 42 then f4 (x*x*x) else f5 (x+x)
f4 x = if x > 180 then 33+x else f4 170
f5 x = f3 (if not (x > 1) then x+77 else 23-x)
```

- a) Für welche Funktionen f_i, f_j ($1 \leq i, j \leq 5$) gilt:
Die Funktion f_i *referenziert* Funktion f_j *direkt*? (4 Punkte)
- b) Für welche Funktionen f_i, f_j ($1 \leq i, j \leq 5$) gilt:
Die Funktion f_i *referenziert* die Funktion f_j ? (4 Punkte)
- c) Welche der Funktionen f_i mit $1 \leq i \leq 5$ sind *direkt rekursiv*? (4 Punkte)
- d) Welche der Funktionen f_i mit $1 \leq i \leq 5$ sind *rekursiv*? (4 Punkte)
- e) Welche Paare (f_i, g_j) mit $1 \leq i < j \leq 5$ sind *verschränkt rekursiv*? (4 Punkte)

Aufgabe 2 (20 Punkte)

- a) Die Funktionen g_1, g_2, g_3 und g_4 seien in Haskell definiert als:

```
g1 a b c = if a-b <= c-b then a+b+c else g1 (2*a) (b/2) c
g2 a b c = if b > c-a then 3+(g2 (a+2) (b-3) (c*4)) else 1337
g3 a b c = if a+b-c == 1 then (g3 c b a)-(g3 (g3 (c-b) (b-a) (a-c)) (a-33) b)
           else 0
g4 a b c = if a == 0 then (g4 (a+b) (b+c) (a+c))*(g4 c c c) else 31415
```

Vervollständigen Sie die folgende Tabelle: (10 Punkte)

	g_1	g_2	g_3	g_4
... ist iterativ	ja / nein	ja / nein	ja / nein	ja / nein
... ist endrekursiv	ja / nein	ja / nein	ja / nein	ja / nein
... ist linear rekursiv	ja / nein	ja / nein	ja / nein	ja / nein
... ist Baum-rekursiv	ja / nein	ja / nein	ja / nein	ja / nein
... ist geschachtelt Baum-rekursiv	ja / nein	ja / nein	ja / nein	ja / nein

- b) Gegeben sei die Funktion $h\ x = \text{if } x < 10 \text{ then } 0 \text{ else } 2*x + h\ (x-1)$, welche Ganzzahlen entgegennimmt. Geben sie eine **endrekursive** Funktion h' an, die für alle Ganzzahlen stets das gleiche Ergebnis wie h liefert. Eine rekursive Hilfsfunktion könnte hierbei nützlich sein. (10 Punkte)

Aufgabe 3 (60 Punkte)

Die Funktion f sei definiert als

```
f a b c = if c >= 21
          then (if c < 27 then a*c else f a (f a a (c+100)) (c+9))
          else f a (f a a (c+100)) (c+9)
```

Geben Sie für alle Teilaufgaben jeweils sämtliche Reduktionsschritte sowie die jeweils verwendete Regel als Buchstabe **D**, **A** oder **I** an (**D** = Definitionseinsetzung, **A**=Arithmetische Auswertung, **I**=if-Auswertung).

- a) Werten Sie $f\ 1\ 2\ 3$ in *normaler Reihenfolge* aus. (15 Punkte)
- b) Geben Sie die ersten 12 Schritte der Auswertung von $f\ 1\ 2\ 3$ in *applikativer Reihenfolge* an. Wie setzt sich die Berechnung danach weiter fort? Begründen Sie Ihre Antwort, die Fortsetzung der Rechnung ist dafür nicht erforderlich. (20 Punkte)
- c) Werten Sie den Ausdruck $f\ (1+1)\ 2\ 17$ in *verzögerter Reihenfolge* aus. (20 Punkte)
- d) Der innere **if-then-else**-Ausdruck von f ist redundant. Geben Sie eine abgewandelte Funktion g an, die komplett identische Ergebnisse liefert wie f – das heißt für jeweils Normalordnung, applikative Auswertung und verzögerte Auswertung soll stets das gleiche Ergebnis rauskommen – allerdings mit nur einem **if-then-else**-Ausdruck auskommt. (5 Punkte)