7. Übungsblatt zu Programmierung WS 2002 / 03

Prof. Dr. Gert Smolka und Dipl.-Inform. Thorsten Brunklaus
`www.ps.uni-sb.de/courses/prog-ws02/`

Abgabe: Montag, 9. Dezember 2002

Die Aufgaben sind auf Papier im Briefkasten Nr. 15 im Durchgang zwischen den Gebäuden 36 und 45 abzugeben. Ihre Abgaben können nur berücksichtigt werden, wenn Sie jeweils Ihren Namen und die Nummer Ihrer Übungsgruppe angeben.

Aufgabe 7.1: Iteratives Tabulate (5) Schreiben Sie eine Prozedur

tabulate : (int → 'a) → int → 'a list

die nur iterative Rekursion benutzt und für $n \in \mathbb{N}$ die folgende Gleichung erfüllt:

`tabulate` $f\ n\ =\ [f\ 0\ , \ldots ,\ f\ n]$

Verwenden Sie eine Hilfsprozedur `tabulate'` mit einem Akkumulatorargument.

Aufgabe 7.2: Reversierendes Map $(10 = 4 + 6)$ Schreiben Sie eine Prozedur

rmap : ('a → 'b) → 'a list → 'b list

die nur iterative Rekursion benutzt und die folgende Gleichung erfüllt:

`rmap` $f\ [x_1, \ldots, x_n]\ =\ [f\ x_n\ , \ldots ,\ f\ x_1]$

(a) Schreiben Sie `rmap` mit `foldl`.

(b) Schreiben Sie `rmap` mithilfe einer iterativen Hilfsprozedur.

Aufgabe 7.3: Iteratives Berechnen von balancierten Binärbäumen $(18 = 3 * 6)$ Seien Bäume gemäß der folgenden Typdeklaration dargestellt:

```
datatype 'a tree = T of 'a * ('a tree list)
```

(a) Schreiben Sie eine Prozedur

tree : int → unit tree

die für $n \in \mathbb{N}$ einen balancierten Binärbaum der Tiefe n mit exponentieller Laufzeit berechnet.

(b) Schreiben Sie eine Prozedur

tree : int → unit tree

die für $n \in \mathbb{N}$ einen balancierten Binärbaum der Tiefe n mit linearer Laufzeit berechnet. Dabei soll nur iterative Rekursion verwendet werden. Verwenden Sie eine Hilfsprozedur `tree'` mit einem Akkumulatorargument.

(c) Schreiben Sie eine Prozedur

```
tree : int -> int tree
```

die für $n \in \mathbb{N}$ einen balancierten Binärbaum der Tiefe n berechnet, dessen Knoten mit der Tiefe des jeweils erreichbaren Teilbaums markiert sind. Beispielsweise soll

```
tree 2 = T(2, [ T(1,[T(0,[]),T(0,[])]),
                T(1,[T(0,[]),T(0,[])]) ])
```

gelten. Dabei soll nur iterative Rekursion verwendet werden und die Laufzeit soll linear sein. Verwenden Sie eine Hilfsprozedur `tree'` mit zwei Akkumulatorargumenten.

Aufgabe 7.4: Rekursionsbäume und Laufzeit $(10 = 2 + 1 + 2 + 5)$ Sei die folgende Prozedur gegeben:

```
fun p n = if n< 4 then () else (p(n-1) ; p(n div 2))
```

(a) Zeichnen Sie den Rekursionsbaum für den Aufruf `p 8`.

(b) Geben Sie die Laufzeit des Aufrufs `p 8` an.

(c) Schreiben Sie eine Prozedur `phi:` `int` $\to$ `int`, die die Laufzeit von `p` berechnet.

(d) Sei p die der obigen Prozedur entsprechende abstrakte Prozedur mit der Typzusicherung $\mathbb{N} \to \{\langle\rangle\}$. Geben Sie die Rekursionsrelation von p an. Terminiert p? Warum?

Aufgabe 7.5: Rekursionsbäume und Laufzeit $(13 = 2 + 1 + 4 + 2 + 4)$ Sei die folgende Prozedur gegeben:

```
fun f(m,n) = if m=0 then n
             else if n=0 then m
                  else if m<n then f(m-1,n+1)
                       else f(m+1,n-1)
```

(a) Zeichnen Sie den Rekursionsbaum für `f(3,45)`.

(b) Ist `f` iterativ?

(c) Geben Sie die exakte Laufzeit von `f` für Argumente $(x, y) \in \mathbb{N}^2$ an.

(d) Geben Sie eine Prozedur `g` an, die für alle Argumente $(x, y) \in \mathbb{N}^2$ dasselbe Ergebnis wie `f` berechnet, aber mit konstanter Laufzeit.

(e) Sei f die der obigen Prozedur entsprechende abstrakte Prozedur mit der Typzusicherung $\mathbb{N}^2 \to \mathbb{N}$. Geben Sie die Rekursionsrelation von f an. Geben Sie eine Funktion an, die die Rekursionsrelation von f in $NO(\mathbb{N})^{\succ}$ einbettet.

Aufgabe 7.6: Größter gemeinsamer Teiler ($23 = 1 + 2 + 2 + 4 + 6 + 6 + 2$) Sei die folgende Prozedur gegeben:

$$gcd: \ \mathbb{N}^+ \times \mathbb{N}^+ \to \mathbb{N}^+$$
$$gcd(x, y) = \text{if } x = y \text{ then } x \text{ else if } x < y \text{ then } gcd(x, y - x) \text{ else } gcd(x - y, y)$$

(a) Ist *gcd* iterativ?

(b) Zeichnen Sie den Rekursionsbaum für den Aufruf `gcd(216,60)`.

(c) Geben Sie die Rekursionsrelation von *gcd* an.

(d) Geben Sie eine Funktion an, die die Rekursionsrelation in $NO(\mathbb{N})^{\succ}$ einbettet. Haben Sie damit die Terminierung von *gcd* bewiesen?

(e) Beweisen Sie, dass *gcd* für alle zulässigen Argumente (x, y) den größten gemeinsamen Teiler von x und y berechnet.

(f) Sei ϕ die Laufzeit von `gcd`. Beweisen Sie durch Induktion:

$$\forall (x, y) \in \mathbb{N}^+ \times \mathbb{N}^+ : \ \phi(x, y) \leq \max\{x, y\}$$

Geben Sie die Grundmenge und die Induktionsrelation ihres Beweises an.

(g) Geben Sie für alle $x \in \mathbb{N}^+$ ein $y \in \mathbb{N}^+$ an, sodass $\phi(x, y) = \max\{x, y\}$.

Aufgabe 7.7: Laufzeitsynthese ($21 = 7 * 2 + 7 * 1$)

(a) Schreiben Sie möglichst einfache Prozeduren `int` $\to$ `unit`, die für $n \in \mathbb{N}$ die folgenden asymptotischen Laufzeiten haben:

$\Theta(\log n)$	`fun tlogn n =` $\cdots$
$\Theta(n)$	`fun tn n =` $\cdots$
$\Theta(n \log n)$	`fun tnlogn n =` $\cdots$
$\Theta(n^2)$	`fun tn2 n =` $\cdots$
$\Theta(n^3)$	`fun tn3 n =` $\cdots$
$\Theta(2^n)$	`fun t2n n =` $\cdots$
$\Theta(3^n)$	`fun t3n n =` $\cdots$

Sie können Hilfsprozeduren verwenden. Die folgende Fakten sind hilfreich:

(i) $\sum_{k=1}^{n} k^2 = \frac{1}{6} n(n+1)(2n+1)$ für $n > 0$.

(ii) Balancierte Binärbäume der Tiefe n haben die Größe $2^{n+1} - 1$.

(iii) Balancierte Ternärbäume der Tiefe n haben die Größe $\frac{1}{2}(3^{n+1} - 1)$.

(b) Schreiben Sie ihre Prozeduren zu Prozeduren des Typs `int` $\to$ `int` um, die die jeweils geforderte asymptotische Laufzeit haben und zusätzlich zu jedem Argument die exakte Laufzeit liefern.