1. 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, 28. Oktober 2002

Aufgaben 1–8 sind per E-Mail an Ihren Übungsgruppenleiter abzugeben. Aufgaben 9–12 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 1.1: Signum (5) Schreiben Sie eine Prozedur *signum: int→int*, die für negative Argumente -1, für positive Argumente 1, und für 0 das Ergebnis 0 liefert.

Aufgabe 1.2: 17. Potenz (8) Schreiben Sie eine Prozedur *hoch17: int→int*, die zu x die Potenz x^{17} berechnet. Dabei sollen möglichst wenig Multiplikationen verwendet werden.

Aufgabe 1.3: Gleitkomma-Potenzen (8) Schreiben Sie eine rekursive Prozedur

```
power : real * int → real
```

die zu einer reellen Zahl x und einer natürlichen Zahl n die Potenz x^n mittels Gleitkommaoperationen berechnet. Welche Zahl liefert `power(3.0,100)`? Handelt es sich dabei wirklich um die Zahl 3^{100}?

Aufgabe 1.4: Fakultäten (10) Schreiben Sie eine rekursive Prozedur

```
fak : int → int
```

die die Fakultätsfunktion berechnet:

$$fak\colon \mathbb{N} \to \mathbb{N}$$
$$fak(n) = 1 \cdot 2 \cdot \cdots \cdot n$$

Dabei soll $fak(0) = 1$ gelten. Für negative Argumente soll die Prozedur `fak` divergieren. Was ist das größte n, für das Sie die Fakultät mit Moscow ML noch berechnen können?

Aufgabe 1.5: Größte gemeinsame Teiler (10) Schreiben Sie eine rekursive Prozedur

```
ggt : int * int → int
```

die zu zwei postiven ganzen Zahlen den größten gemeinsamen Teiler berechnet. Verwenden Sie dabei die folgenden Gleichungen:

$$\begin{array}{ll} ggt(x,x) = x & \text{falls } x > 0 \\ ggt(x,y) = ggt(x-y,y) & \text{falls } x > y > 0 \\ ggt(x,y) = ggt(y,x) & \text{falls } x, y > 0 \end{array}$$

Aufgabe 1.6: Quersumme (10) Schreiben Sie eine rekursive Prozedur

quer : int → int

die die Quersumme einer ganzen Zahl berechnet. Die Quersumme einer Zahl ist die Summe der Dezimalziffern der Zahl. Beispielsweise hat die Zahl -3754 die Quersumme 19. Verwenden Sie ganzzahlige Division (`div`) und ganzzahliger Restbildung (`mod`).

Aufgabe 1.7: Natürliche Quadratwurzel (10) Schreiben Sie eine Prozedur

wurzel : int → int

die zu einer natürlichen Zahl x die größte natürliche Zahl n berechnet mit $n^2 \leq x$. Deklarieren Sie dazu zunächst eine Hilfsprozedur

*wurzel′ : int * int → int*

die zu zwei natürlichen Zahlen n und x die kleinste natürliche Zahl $m \geq n$ berechnet mit $m^2 > x$ (falls ein solches m existiert).

Aufgabe 1.8: Standardstruktur Math (5) Deklarieren Sie Prozeduren des Typs `real` → `real`, die die Funktionen

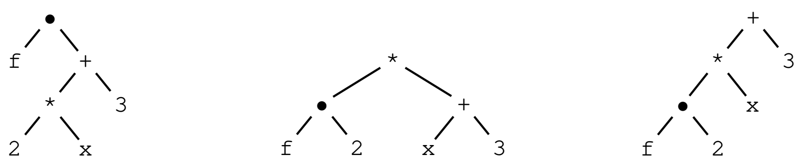
$$g(x) = 2x + 1,4$$
$$h(x) = \sin x + \cos(2\pi x)$$

mit Gleitkommaoperationen berechnen. Verwenden Sie dazu die Standardstruktur `Math`. Testen Sie Ihre Prozeduren, indem Sie einige Ausdrücke mit Moscow ML auswerten.

Aufgabe 1.9: Umgebungen (10) Welche Umgebung berechnet das folgende Programm?

```
fun f (x:bool) = if x then 1 else 0
val x = 5*7
val y = 6*6
fun g (z:int) = f(z<y)<x
val x = g 5
```

Aufgabe 1.10: Klammersparregeln (6) Geben Sie für die folgenden Ausdrücke minimal geklammerte Zeichendarstellungen an:



Aufgabe 1.11: Baumdarstellungen (8) Geben Sie Wort- und Baumdarstellungen für die folgenden 4 Phrasen an:

```
1+2+3+4
1+2 * x-y * 3+4
int * int * int->real
fun potenz(x:int,n:int):int=if n>0then x*potenz(x,n-1)else 1
```

Aufgabe 1.12: Auswertungsprotokolle (10) Gegeben sei die Prozedurdeklaration

```
fun f(n:int, a:int) = if n=0 then a else f(n-1, a*n)
```

Geben Sie das Auswertungsprotokoll für den Ausdruck `f(3,1)` an. Halten Sie sich dabei genau an die im Skript angegebene Beschreibung des Auswertungsprozesses. Sie bekommen dann 18 Auswertungsschritte.