**Programmierung WS 2002 / 03:**
**Musterlösung zum 4. Übungsblatt**

Prof. Dr. Gert Smolka, Dipl.-Inform. Thorsten Brunklaus

**Aufgabe 4.1: Last** (8)

```
fun last xs = hd (rev xs)
```

Eine effizientere Lösung in bezug auf Laufzeit und Speicherverbrauch (Sie werden bald verstehen, warum) ist:

```
fun last (x::nil) = x
 |  last (_::xr)  = last xr
 |  last nil      = raise Empty
```

**Aufgabe 4.2: Enum** (8)

```
fun enum (m,n) = if m<=n then m::enum(m+1,n) else nil
```

**Aufgabe 4.3: Nth, Take, Drop** ($15 = 3 * 5$)

(a)
```
fun nth(xs,n) = if n<0 orelse null xs
                then raise Subscript
                else if n=0 then hd xs
                     else nth(tl xs, n-1)
```

(b)
```
fun take(xs,n) = if n=0 then nil
                 else if n<0 orelse null xs
                      then raise Subscript
                      else hd xs :: take(tl xs, n-1)
```

(c)
```
fun drop(xs,n) = if n=0 then xs
                 else if n<0 orelse null xs
                      then raise Subscript
                      else drop(tl xs, n-1)
```

**Aufgabe 4.4: Max** (10)

```
fun max xs = if null xs then raise Empty
             else foldl (fn (x,m) => if x<=m then m else x)
                        (hd xs) (tl xs)
```

**Aufgabe 4.5: Member** ($15 = 3 * 5$)

(a)
```
fun member x nil     = false
  | member x (y::yr) = x=y orelse member x yr
```

(b)
```
fun member x = List.exists (fn y => x=y)
```

(c)
```
fun member x = foldl (fn (y,b) => x=y orelse b) false
```

**Aufgabe 4.6: Count** (10)

```
fun count y = foldl (fn (x,n) => if x=y then n+1 else n) 0
```

**Aufgabe 4.7: Dezimaldarstellung** ($10 = 2 * 5$)

(a)
```
fun dec x  = if x<10 then [x] else dec(x div 10) @ [x mod 10]
```

(b)
```
fun int ds = foldl (fn (d,x) => 10*x+d) 0 ds
```

**Aufgabe 4.8: Permutationen** (4)

```
fun perm (xs,ys) = sort xs = sort ys
```

**Aufgabe 4.9: Partition** ($10 = 2 * 5$)

(a)
```
fun partition x nil     = (nil, nil)
  | partition x (y::yr) = let
                            val (us,vs) = partition x yr
                          in
                            if y<x then (y::us, vs)
                            else (us, y::vs)
                          end
```

(b)
```
fun partition x = foldl (fn (y, (us,vs)) =>
                          if y<x then (y::us, vs)
                          else (us, y::vs))
                        (nil,nil)
```

**Aufgabe 4.10: Quicksort** (10)

```
fun qsort nil     = nil
  | qsort (x::xs) = let
                      val (us,vs) = partition x xs
                    in
                      qsort us @ [x] @ qsort vs
                    end
```