University of Innsbruck — Institute of Computer Science
Test 1A — 30 November 2007

Functional Programming | WS 2007/2008 | LVA 703018

Name: **MatrNr:** **StudienKZ:**

This test consists of four exercises. *Explain your answers.* The available points for each item are written in the margin.

[6] 1. Consider the function take : int −> 'a list −> 'a list, defined by:

```
let rec take n xs =
 if n <= 0 then [] else match xs with
   | [] -> []
   | x :: xs -> x :: take (n − 1) xs
;;
```

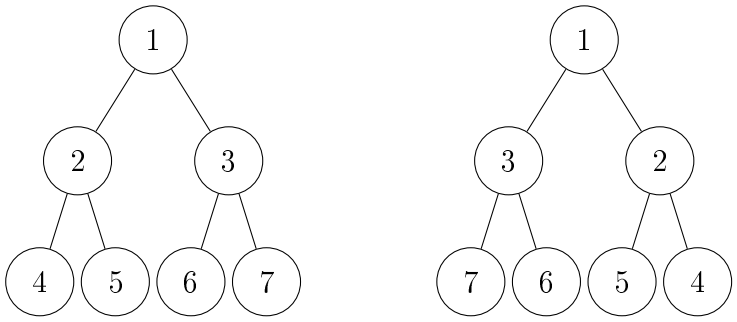
Give at least 5 different intermediate steps of the derivation sequence starting at:

```
take 2 [5; 7; 2; 4]
 →  if 2 <= 0 then [] else 5 :: take (2 − 1) [7; 2; 4]
```

[5] 2. Write a function list_max : 'a list -> 'a that computes the maximum of a list using the function fold1 : ('a -> 'a -> 'a) -> 'a list -> 'a, defined by:

```
let rec fold1 f = function
  | [] -> failwith "fold1: empty list"
  | [x] -> x
  | x :: xs -> f x (fold1 f xs)
;;
```

[8] 3. Consider a binary tree (**type** 'a tree = Empty | Node **of** ('a tree * 'a * 'a tree)). Define a function mirror : 'a tree -> 'a tree that mirrors a tree, i.e., changes the tree on the left into the one on the right.



4. Consider the lambda-term $t = \lambda xy.x\ ((\lambda xz.y)\ z)\ w$.

[2] (a) Compute all subterms of t.

[2] (b) Compute $\mathcal{V}\mathsf{ar}(t)$, $\mathcal{BV}\mathsf{ar}(t)$, and $\mathcal{FV}\mathsf{ar}(t)$.

[2] (c) Is t in normal form?