TECHNISCHE UNIVERSITÄT MÜNCHEN
FAKULTÄT FÜR INFORMATIK

WS 11/12
Einführung in die Informatik 2 **Mini-Test 8**
Prof. Dr. Andrey Rybalchenko, M.Sc. Ruslán Ledesma Garza 20.12.2011

Bearbeitungszeit : 15 min

Name, Vorname und Matrikelnummer **(Bitte leserlich schreiben!)** Gruppe

Aufgabe 8.1 [3 Punkte] **Zähler**

Schreiben Sie eine Prozedur `count : unit → int`, die einen Zähler implementiert, d.h., bei jedem Aufruf die Anzahl der bisherigen count-Aurufe liefert. Die verwendete Referenz sollte von aussen nicht zugreifbar sein.

Lösungsvorschlag 8.1

```
val count =
let val c = ref 0 in
fn () ⇒ c := !c + 1; !c
end
```

Aufgabe 8.2 [3 Punkte] **Length**

Betrachten Sie die folgende Definition.

```
datatype 'a mylist = Nil | Cons of 'a * 'a mylist
```

Schreiben Sie eine Prozedur `length : 'a mylist → int`, die die Anzahl von Cons-Konstruktoren in einem mylist-Wert bestimmt.

Lösungsvorschlag 8.2

```
datatype 'a mylist = Nil | Cons of 'a * 'a mylist
fun length Nil = 0
| length (Cons (x, xs)) = 1 + length xs
```

Aufgabe 8.3 [4 Punkte] **Vertreter**

Betrachten Sie die folgende Definition.

datatype `exp = C of int | V of string | A of exp * exp | M of exp * exp`

Schreiben Sie eine Prozedur `substs : string → exp → exp → exp`, die zu einer String `s`, einem Ausdruck `se` und einem Ausdruck `e` den Ausdruck liefert, den man aus `e` erhält, indem man jedes Vorkommen von `(V s)` durch `se` ersetzt.

Lösungsvorschlag 8.3

```
datatype exp = C of int | V of string | A of exp * exp | M of exp * exp
fun subst v se e =
case e of
C _ ⇒ e
| V v' ⇒ if v = v' then se else V v'
| A (e1, e2) ⇒ A (subst v se e1, subst v se e2)
| M (e1, e2) ⇒ M (subst v se e1, subst v se e2)
```

Aufgabe 8.4 [5 Punkte] **Laub**

Betrachten Sie die folgende Definition.

datatype `binary_tree = Leaf of int | Node of binary_tree * int * binary_tree`

Schreiben Sie eine Prozedur `leaves : binary_tree → int list`, die eine Lister der Blätter eines Baums liefert.

Lösungsvorschlag 8.4

```
datatype binary_tree = Leaf of int | Node of binary_tree * int * binary_tree
fun leaves (Node (l, _, r)) = (leaves l) @ (leaves r)
| leaves (Leaf n) = [n]
```

Feedback Die folgenden Fragen gehören nicht zum Test. Sie beeinflussen Ihre Punkte nicht, sondern dienen uns nur dazu, die Vorlesung einzuschätzen.

a) Wie schwer finden Sie den Stoff der letzten Vorlesungswoche?
☐ leicht ☐ normal ☐ schwierig ☐ sehr schwierig

b) Wie schwer würden Sie diesen Test finden *wenn Sie sich entsprechend vorbereitet haben*?
☐ leicht ☐ normal ☐ schwierig ☐ sehr schwierig

c) Kommentare?