

## Programmierung 1 (Wintersemester 2012/13)

---

### Erklärung 8

(Rekursive Abstraktionen)

---

**Hinweis:** Dieses Blatt enthält eine zusätzliche Erklärung erstellt von den Tutoren. Für die Richtigkeit besteht daher keine Gewähr.

*Die Erklärung sowie ihr Thema sind für die Klausur weder relevant noch irrelevant.*

### Rekursive Prozeduren als Erweiterung von `elab` und `eval`

Zunächst die Erweiterung der Datentypen:

```
datatype value = IV of int | Proc of id*exp*value env | RProc of id*id*exp*value env
```

```
datatype exp = ...  
| Rabs of id*id*ty*ty*exp  
(*Prozedurname, Argument, Argumenttyp, Ergebnistyp, Prozedurrumpf*)
```

Dann die Erweiterung von `elab`:

```
fun elab ...  
| elab f (Rabs(g,x,t,t',e)) =  
Arrow (t, elab update (update f g (Arrow(t,t'))) x t) e)
```

Und schließlich die Erweiterung von `eval`:

```
fun eval ...  
| eval f (Rabs(g,x,t,t',e)) = RProc (g,x,e,f)  
| eval f (App(e1,e2)) = case (eval f e1, eval f e2) of  
  (Proc(x,e,f'),v) => ... (*wie vorher*)  
  | (RProc(g,x,e,f'),v) => let  
    val f'' = update (update f' g (Rproc(g,x,e,f')))) x v  
    in eval f'' e end  
| _ => raise Error
```