

Small step Semantik

Übung 4, Aufgabe 1 d

```
let f = λx:int. 0
in let f = λx:int. if x = 0 then 1 else f (x - 1)
   in f 3
```

steht für

```
let f = λx:int. 0
in let f = λx:int. if = x 0 then 1 else f (- x 1)
   in f 3
```

→ let f = λx:int. if = x 0 then 1 else (λx:int. 0) (- x 1)
 in f 3

mit Regel (LET-EXEC)

→ (λx:int. if = x 0 then 1 else (λx:int. 0) (- x 1)) 3

mit Regel (LET-EXEC)

→ if = 3 0 then 1 else (λx:int. 0) (- 3 1)

mit Regel (BETA-V)

→ if false then 1 else (λx:int. 0) (- 3 1)

mit Regel (BOP)

→ (λx:int. 0) (- 3 1)

mit Regel (COND-FALSE)

→ (λx:int. 0) 2

mit Regel (BOP)

→ 0

mit Regel (BETA-V)