Aufgabe 6-2:

a)

```
(define f (lambda (x) (local [(define y 3)](* x y))))
(f 3)
```

CONST, APP

→ (local [(define y 3)](* 3 y))

LOCAL
→ (define y_0 3)
    (* 3 y_0)
PROG, PRIM
→ (* 3 y_0)
CONST
→ (* 3 3)
PRIM
→ 9

b)

```
(define f (lambda (x) (local [(define y 5)]
                                    (+ x (local [(define x 3)] (+ x y))))))
(f 7)
```

CONST, APP
→ (local [(define y 5)]
                (+ 7 (local [(define x 3)] (+ x y))))
LOCAL
→ (define y_0 5)
    (+ 7 (local [(define x 3)] (+ x y_0)))
PROG
→ (+ 7 (local [(define x 3)] (+ x y_0)))
LOCAL
→ (define x_0 3)
    (+ 7 (+ x_0 y_0))
PROG
→ (+ 7 (+ x_0 y_0))
CONST
→ (+ 7 (+ 3 5))
PRIM
→ 15