

1) *Lösung.* Eine mögliche Implementierung wäre wie folgt:

```
module Main where

data BF = Zero | One | Add BF BF | Mul BF BF | Not BF
  deriving (Show, Eq)

evaluate :: BF → BF
evaluate Zero = Zero
evaluate One = One
evaluate (Add x y) | (evaluate x == Zero) && (evaluate y == Zero) = Zero
                  | otherwise = One
evaluate (Mul x y) | (evaluate x == One) && (evaluate y == One) = One
                  | otherwise = Zero
evaluate (Not x) | evaluate x == Zero = One
                 | otherwise = Zero

a, b, c :: BF
a =
  (Not
    (
      ((Zero 'Add' ((One 'Mul' Zero) 'Mul' Not (One))) 'Mul' Zero)
      'Add' (((One 'Add' Zero) 'Mul' (Not One)) 'Add' Zero)
    )
  )
  'Mul' ((Not One) 'Add' One)
b = Not
  (
    (Not (Not ((Not Zero) 'Add' One)) 'Mul' (Zero 'Mul' (Not One)))
    'Add' ((Not ((Not Zero) 'Add' One)) 'Mul' (Not (Zero 'Mul' (Not One))))
  )
c =
  ((One 'Add' (Not One)) 'Mul' (Zero 'Add' (Not ((Not Zero) 'Mul' One))))
  'Add' (
    (Not ((Zero 'Mul' (Not One)) 'Add' ((Not Zero) 'Mul' One)))
    'Mul' (Not (Zero 'Mul' One))
  )

main :: IO ()
main = mapM_
  (λ(s, f) →
    putStrLn
      $ s
      ++ " = "
      ++ (show f)
      ++ "\nEvaluate "
      ++ s
      ++ ":\n")
```

```

    ++ (show $ evaluate f)
    ++ "λnλn"
  )
[("a", a), ("b", b), ("c", c)]

```

Den oben angegebenen Source finden Sie auch auf der LVA Homepage. □

- 3) *Lösung.* a) Die Grammatik G_1 ist kontextsensitiv und beschreibt die kontextfreie Sprache $\{ab\} \cup \{a^n cb^n | n > 0\}$. Die kontextfreie Grammatik G'_1 beschreibt die selbe Sprache.

$$\begin{aligned}
 S &\rightarrow ab \mid aCb \\
 C &\rightarrow aCb \mid c
 \end{aligned}$$

- b) Die Grammatik G_2 ist kontextfrei und beschreibt die reguläre Sprache $\{\text{int } x = 42, \text{int } y = 0, \text{bool } \text{iscf} = \text{false}, \dots\}$, angelehnt an Variablendeklarationen in der Programmiersprache C. Die vier Ableitungen

$$\begin{aligned}
 S &\rightarrow D_i \mid D_b \\
 D_i &\rightarrow \text{int } I = V_i \\
 D_b &\rightarrow \text{bool } I = V_b
 \end{aligned}$$

sind nicht rechtslinear. Die Grammatik kann aber einfach angepasst werden.

$$\begin{aligned}
 S &\rightarrow \text{int } I_i \mid \text{bool } I_b \\
 I_i &\rightarrow a = V_i \mid \dots \mid z = V_i \mid aI_i \mid \dots \mid zI_i \\
 I_b &\rightarrow a = V_b \mid \dots \mid z = V_b \mid aI_b \mid \dots \mid zI_b \\
 &\dots
 \end{aligned}$$

- c) Die Grammatik G_3 ist kontextfrei und beschreibt die kontextfreie Sprache der Booleschen Ausdrücke ohne Variablen. □