



Programmieren in MOSTflexiPL

Vorlesung im Wintersemester 2025/2026
 Prof. Dr. habil. Christian Heinlein

4. Aufgabenblatt (1. Dezember 2025)

Aufgabe 8: Generische Operatoren

Teilaufgabe 8.a)

Definieren Sie die logischen Operatoren aus Aufgabe 1 generisch, sodass sie auf Werte x und ggf. y mit beliebigen (auch verschiedenen) Typen X und Y angewandt werden können:

- Die Negation / x liefert true , wenn x nil ist, sonst false (d. h. der Resultattyp ist weiterhin `bool`).
- Die Konjunktion $x / \wedge y$ liefert y , wenn x nicht nil ist, sonst nil mit Typ Y (d. h. der Resultattyp ist Y).
- Die Disjunktion $x / \vee y$ liefert y , wenn x nil ist, sonst einen neuen synthetischen Wert des Typs Y (d. h. der Resultattyp ist ebenfalls Y).

Definieren Sie zusätzlich eine Spezialisierung der Disjunktion für Werte x und y mit dem gleichen Typ XY , die x liefert, wenn es nicht nil ist, und sonst y (d. h. der Resultattyp ist XY).

Aufgrund der in §3.9.2 der Vorlesungsfolien erläuterten Regeln, wird diese Spezialisierung bei einer Verwendung der Disjunktion gegenüber dem allgemeineren Operator bevorzugt, wenn beide Operanden den gleichen Typ besitzen.

Definieren Sie zusätzlich zu den bereits in Aufgabe 1 definierten Bindungseigenschaften:

- Beide Varianten der Disjunktion besitzen die gleichen Bindungseigenschaften.
- Im linken Operanden der Disjunktion und Konjunktion sind alle Operatoren ausgeschlossen, die im linken Operanden der vordefinierten Zuweisung $\bullet = ! \bullet$ ausgeschlossen sind (das schließt die Zuweisung selbst mit ein).
- Im rechten Operanden der Disjunktion und Konjunktion sowie im Operanden der Negation sind alle Operatoren ausgeschlossen, die im rechten Operanden der vordefinierten Zuweisung ausgeschlossen sind, sowie die Zuweisung selbst (die in ihrem eigenen rechten Operanden nicht ausgeschlossen ist).
- Disjunktion und Konjunktion sind überall ausgeschlossen, wo der vordefinierte Gleichheitstest $\bullet == \bullet$ ausgeschlossen ist.
- Die Negation ist im linken Operanden des vordefinierten Gleichheitstests ausgeschlossen. Außerdem ist sie überall ausgeschlossen, wo der vordefinierte Operator `char` \bullet (Umwandlung von `int` nach `char`) ausgeschlossen ist.

Diese Ausschlüsse bewirken zusammen, dass sich die drei logischen Operatoren in der Vorranghierarchie von §2.13.1 zwischen der Zuweisung und dem Gleichheitstest befinden.

Teilaufgabe 8.b)

Definieren Sie, ähnlich zu Aufgabe 7, einen Operator, der verkettete Gleich-/Ungleichheitstests mit zwei oder mehr Operanden eines beliebigen Typs erlaubt, zum Beispiel:

```
b1 = b2 =/ b3      $$ mit b1, b2, b3 vom Typ bool  
c1 =/ c2 = c3 = c4    $$ mit c1, c2, c3, c4 vom Typ char
```

Wie der Operator aus Aufgabe 7, soll auch dieser Operator die gleichen Bindungseigenschaften wie der vordefinierte Gleichheitstest `==` haben.

Teilaufgabe 8.c)

Definieren Sie die Operatoren zur Vertauschung von Variablenwerten aus Aufgabe 3 sowie zur Rotation von Variablenwerten aus Aufgabe 6 generisch, sodass sie jeweils auf Variablen mit einem beliebigen Inhaltstyp angewandt werden können.

Teilaufgabe 8.d)

Definieren Sie einen generischen Postfix-Operator `•?`, sodass `x?` für einen Wert `x` eines beliebigen Typs `X` jeweils eine neue Variable mit Typ `X?` und Anfangswert `x` liefert.

Definieren Sie für diesen Operator folgende Ausschlüsse:

- Die vordefinierte Konstantendeklaration ist im Operanden dieses Operators ausgeschlossen, sodass `v := 1?` eine Variable `v` mit Inhaltstyp `int` und Anfangswert `1` definiert (und nicht etwa eine Konstante `v` mit Typ `int` und Wert `1` innerhalb des Operanden des Operators `•?`).
- Der Operator selbst ist im Operanden des vordefinierten Schleifenoperators `?*•` sowie im Operanden der vordefinierten Variablenabfrage `?•` jeweils ausgeschlossen.

Weitere Ausschlüsse sind nicht notwendig, weil viele andere Ausdrücke wie z. B. `1 + 2 ?` oder `x =! y ?` bereits aufgrund der Typen der beteiligten Teilausdrücke eindeutig sind.