2. Übungsblatt

Ausgabe: 24.06.08
Abgabe: 25.07.08

Dieses Übungsblatt beschäftigt sich mit der statischen Analyse der Sprache C. Lernziele hierbei sind die verschiedenen Typkonversionen in C (explizit und implizit), und die Feinheiten der Typüberprüfung in C.

4 *Konversionen hier und da*

Ausgehend von der in Aufgabe 2 (Übungsblatt 1) entwickelten Repräsentation von Typen entwickeln Sie zwei Funktionen, welche zum einen *integer promotion* (C-Standard, §6.3.1.1(2)) und zum anderen die *usual arithmetic conversions* (C-Standard, §6.3.1.8) implementieren. Die erste Funktion ist einstellig, die zweite zweistellig.

Hinweis: Sie werden Hilfsfunktionen wie beispielsweise für den *conversion rank* benötigen.

5 *Wie war noch der Typ?*

Entwickeln Sie einen Datentypen `symtab` für Symboltabellen, welcher es erlaubt, Bezeichner (Symbole) mit ihrer Typinformation abzulegen, nach einem Symbol zu suchen, und ein Symbol wieder zu löschen.

6 *Personalwechsel, bitte den Typ vorzeigen.*

Unter Nutzung der beiden letzten Aufgaben und des vorherigen Übungsblattes (bzw. dessen Lösung) entwickeln Sie jetzt eine Funktion, welche innerhalb eines Kontextes (gegeben durch eine Symboltabelle), der Bezeichnern (Variablen) Typen zuorden

- zum einen Deklarationen parsiert, und diese in die Symboltabelle einträgt,
- und zum anderen Ausdrücke parsiert, und die Typkorrektheit dieser Ausdrücke nach dem C-Standard prüft.

Hinweise: Auch hier können Sie annehmen, dass alle Präprozessor-Anweisunge aufgelöst worden sind, und brauchen Typdefininitionen (`typedef`) nicht zu behandeln. Funktionsaufrufe können behandelt werden, als wenn kein Prototyp vorliegt. Sie brauchen keine Anweisungen zu behandeln, aber Fragmente der Sprache wie diese, d.h. Deklaration gefolgt von einem Ausdruck:

```
int x;
char *y:

*(&x+ 5)- *y;
```

7 *Prototypisch Funktional (Zusatzaufgabe)*

Erweitern Sie Symboltabelle und Typüberprüfung so, dass zum einen Funktionsprototypen geparst und in die Symboltabelle eingetragen werden, und zum anderen beim Funktionsaufruf auf Typkorrektheit überprüft werden. Ellipsen (`...`) brauchen nicht behandelt zu werden.