PR ÜBERSETZERBAU 621.401 – 621.402 **SS 2007**

Institut für Informationstechnologie (ITEC) **Taschwer**

Übungsblatt 10

Termin: 2007/06/12

Ü 10.1 Peephole-Optimierung

Führen Sie für die unten angegebene Drei-Adress-Codesequenz folgende Optimierungen durch:

- Eliminierung unerreichbaren Codes;
- Eliminierung unnötiger Sprünge;
- algebraische Vereinfachungen.

Woran erkennt ein Compiler diese Optimierungsmöglichkeiten, und wie führt er die Transformationen durch?

```
L1: if i <= 0 goto L4
    a := a + b
    b := b * 4
    if 0 = 0 goto L2
    i := i + 0
    goto L3
L2: i := i / 2
    goto L3
L3: goto L1
L4: return a
```

Ü 10.2 Compiler-Bootstrapping

a) Angenommen, Sie haben einen C-Compiler, der in C geschrieben ist, so erweitert, dass er Maschinencode für eine neue Zielarchitektur X erzeugt. Ihnen liegt ein C-Compiler vor, der auf Ihrer Entwicklerplattform E ausführbar ist und E-Maschinencode erzeugt. Wie erhalten Sie daraus einen C-Compiler, der auf X ausführbar ist und X-Maschinencode erzeugt?

b) Angenommen, der aus a) resultierende C-Compiler erzeuge *nicht optimierten* X-Maschinencode. Wie können Sie aus dem C-Source-Code für einen optimierenden C++-Compiler, der X-Maschinencode erzeugt, einen *optimierten*, auf X ausführbaren C++-Compiler erhalten?