

Grundlagen der Programmiersprachen und Compilerbau

Wintersemester 2009/2010

1. Übungsblatt 5/6. November 2009

Organisatorische Hinweise

Die Übungen finden 14-täglich statt. Alle Informationen sind auf der Webseite verfügbar.
http://www.iste.uni-stuttgart.de/ps/Lehre/WS0910/V_Grundlagen.

Aufgabe 1.1

Vergleichen Sie die Programmausführung durch einen Interpreter mit den Vorgängen bei Verwendung eines Übersetzers.

- Welche Vor- und Nachteile haben die beiden Ansätze ?
- Nennen Sie Beispiele für Programmiersprachen, die typischerweise durch Interpreter bzw. Übersetzer oder durch Mischformen der beiden Ansätze implementiert werden.
- Besteht ein Zusammenhang zwischen charakteristischen Eigenschaften einer Sprache und der meistens verwendeten Art der Implementierung ?

Aufgabe 1.2

Das folgende Ada-Programm enthält einige Fehler. Eigentlich sollte es die ersten zehn Fibonacci-Zahlen ausgeben. Finden Sie alle Fehler im Programm. Geben Sie genau an, an welcher Programmstelle der Übersetzer bzw. die Laufzeitumgebung den Fehler melden würde.

```
with Ada.Text_IO;

procedure Fehler
is
  type My_Array is array (1 .. 10) of Integer;
  Fib : My_Array := (1 .. 2 => 1);
  I   : Integer := Fib'First + 1;
begin
  while I <= Fib'Last do
    Fib (I) := Fib (I-1) + Fib (I-2);
    I := I + '1';
  end loop;
```

```

    for J in Fib'Range loop
      Ada.Text_IO.Put_Line
        (Integer'Image (J) & ". fibonacci is " & Integer'Image (Fib(J)));
    end loop;
end Fehler

```

Aufgabe 1.3

Gegeben sei ein Compiler für die Sprache Oberon, der Code für PCs erzeugt und auf PCs ausführbar ist. Der Compiler ist in Oberon geschrieben; der Quelltext ist vorhanden.

Dieser Compiler soll auf eine Workstation übertragen werden, d.h. er soll dort ausführbar sein und Code generieren, der auf der Workstation läuft. Zwischen beiden Rechnern können sowohl Text- als auch Binärdateien übertragen werden. Anschliessend soll zusätzlich eine auf die speziellen Erfordernisse des SSE4-Befehlssatzes der Workstation zugeschnittene Codeoptimierung in den Compiler eingebaut werden, von welcher der Compiler auch selber profitieren soll. Erläutern Sie die hierzu erforderlichen Schritte anhand von T-Diagrammen.

Aufgabe 1.4

Sie sind für die Compilerentwicklung in einer Firma verantwortlich. Vor Jahren wurde für einen Kunden ein Compiler für die Spezialsprache JCN entwickelt, der auf und für Rechner des Typs „Dino“ (D) implementiert ist. Der Compiler ist in einer Untermenge von JCN geschrieben, die zielmaschinenabhängigen Komponenten sind in Hinblick auf Ersetzbarkeit geschrieben. Weltweit sind keine anderen JCN-Compiler erhältlich. Weil Dino-Rechner vom Hersteller neuerdings nicht mehr gewartet werden, will der Kunde einen ähnlichen JCN-Compiler auf und für Rechner des Typs „Tachyonrush“ (T).

1. Die Firmenleitung hat Ihnen volle Verantwortung für die Entwicklung des JCN-Compilers für die Tachyonrush Systeme übertragen. Beschreiben Sie Ihre technische Vorgehensweise graphisch mit T-Diagrammen und kurzer, präziser Charakterisierung der wesentlichen Arbeitsschritte und Arbeitsprodukte.
2. Bei dieser Gelegenheit hätte der Kunde auch gerne einige kleine Änderungen in der übersetzten Programmiersprache, nun JCN09 genannt. Leider ist JCN09 nicht voll kompatibel mit JCN, so dass zu erwarten ist, dass die Quelltexte des Compilers minimal angepasst werden müssen, bevor sie mit einem JCN09-Compiler übersetzt werden können. Nach Auslieferung des JCN09-Compilers soll die Wartung von JCN-Compilern eingestellt werden, d.h., der JCN09-Compiler soll mit sich selbst übersetzbar sein. Modifizieren oder ergänzen Sie Ihre obige Vorgehensweise entsprechend, um einen JCN09-Compiler auf und für Tachyonrush Systeme zu produzieren.