

Programmierübungen

Wintersemester 2007/2008

0. Übungsblatt

10. Oktober 2007

Organisatorische Hinweise

Dieses Übungsblatt dient als Einstieg in die Ada-Programmierung. Die Bearbeitung ist freiwillig. Es erfolgt keine Abgabe. Sollten Sie bei der Bearbeitung auf Probleme stoßen, fragen Sie nach! Arbeiten Sie in kleinen Gruppen und tauschen Sie Erfahrungen aus!

Die Gruppenübungen beginnen in KW 45 ab 5.11.2007.

Die Übungsblätter werden jeweils auf der Webseite zur Vorlesung zum Download bereitgestellt. Abgabe erfolgt durch das EClaus-System.

<http://www.iste.uni-stuttgart.de/ps/Lehre/WS0708/inf-prokurs>

Aufgabe 0.1: Einarbeitung

(0 Punkte)

Machen Sie sich mit ihrer Arbeitsumgebung vertraut. Testen Sie verschiedene Texteditoren oder Entwicklungsumgebungen und wählen Sie eine geeignete aus. Legen Sie jeweils die Datei „hello_world.adb“ an und schreiben Sie dieses Ada-Programm:

```
with Ada.Text_IO;
```

```
procedure Hello_World
is
begin
  Ada.Text_IO.Put_Line ("Hello World.");
end Hello_World;
```

Folgende Liste von Werkzeugen kann als Start dienen (nicht alle werden gleichzeitig auf Ihrem System verfügbar sein):

- scite (Linux + Windows)
- AdaGIDE <http://adagide.martincarlisle.com/> (nur für MS Windows)
- emacs
- vi
- GNAT Programming Studio (GPS), <https://libre2.adacore.com/gps/> (derzeit nicht im Grundstudiumspool installiert)
- Texteditor Ihrer Wahl
- Notepad (nur für MS Windows; *Notlösung, suchen Sie ein anderes Werkzeug*)

Übersetzen Sie das Programm mit dem Befehl

```
$ gnatmake hello_world
```

Prüfen Sie ob Ihre Entwicklungsumgebung dafür eine Tastenkombination anbietet.

Aufgabe 0.2: Einarbeitung

(0 Punkte)

a) Schreiben Sie ein Ada Programm „Calculate“, mit folgender Funktionalität:

- Das Programm fordert den Benutzer auf, nacheinander zwei ganze Zahlen a, b einzugeben. Das Programm speichert diese Zahlen in zwei Variablen des Typs Integer.
- Das Programm gibt die Ergebnisse der Berechnungen aller vier Grundrechenarten aus: $a+b, a-b, a*b, \frac{a}{b}$. Für die Darstellung von $\frac{a}{b}$ soll das Programm den ganzzahligen Anteil der Rechnung (mit dem Ada-Operator A / B) und den Divisionsrest (mit dem Ada-Operator $A \text{ rem } B$) getrennt ausgeben.

Beispiel:

```
Gib die erste Zahl ein! a = 15
Gib die zweite Zahl ein! b = 5
```

```
a + b = 20
a - b = 10
a * b = 75
a / b = 3 Rest 0
```

Hinweis: Halten Sie sich genau an das hier angegebene Ausgabeformat! Bei künftigen Abgaben wird die Ausgabe Ihres Programms automatisiert auf Korrektheit überprüft. Formatierungsfehler in der Ausgabe führen dann zu Punktabzug.

- b) Testen Sie was geschieht, falls $b = 0$ eingegeben wird. Erweitern Sie Ihr Programm (mit einem If-Statement), so dass in diesem Fall eine Fehlermeldung „Kann nicht dividieren!“ ausgegeben wird.
- c) Testen Sie Ihr Programm mit den Zahlen 2_000_000_000 und 1_000_000_000. Versuchen Sie die Ergebnisse zu erklären. Übersetzen Sie Ihr Programm erneut, aber diesmal mit dem Schalter `-gnato`. Falls `gnatmake` eine Meldung ausgibt, dass Ihr Programm bereits „up to date“ sei, so müssen Sie zusätzlich den Schalter `-f` angeben:

```
$ gnatmake -f -gnato calculate
```

Testen Sie erneut. Hat sich das Verhalten verändert? Falls ja, welches Verhalten erscheint Ihnen sinnvoller?

Die Entwicklungsumgebung AdaGIDE verwendet den Schalter `-gnato` als Standard, das lässt sich jedoch ausschalten. Finden Sie heraus wie.