

Übung zur Vorlesung 'Programmieren in C'

Übungsblatt 1

Prof. Thomas Scheffler, Beuth-Hochschule Berlin, FB VII

3. März 2011

Aufgabe 1

Informatiker haben die ärgerliche Angewohnheit normale Worte einer Sprache zu benutzen und ihnen eine ganz eigene Bedeutung zu geben die von ihrer normalen Verwendung abweicht. So haben zum Beispiel die Worte *Aussage (statement)* und *Kommentar (comment)* normalerweise eine sehr ähnliche Bedeutung. In einer Programmiersprache sind sie aber sehr unterschiedlich. Es ist wichtig die Bedeutung der Elemente der Programmiersprache genau zu kennen und sie richtig einzusetzen, anderenfalls können sie keine korrekten Programme schreiben.

Das Glossar am Ende eines jeden Kapitels dient dazu wichtige Begriffe und Phrasen zu rekapitulieren und die besondere Bedeutung dieser Begriffe klar zu machen.

Achten sie darauf, dass ein Begriff den sie aus der Umgangssprache kennen eine ganz eigene Bedeutung haben kann wenn er in der Programmierung verwendet wird.

- Was ist der Unterschied zwischen einer *Aussage (statement)* und einem *Kommentar (comment)* in einer Programmiersprache?
- Was bedeutet es wenn man sagt ein Programm sei *portabel*?
- Was bedeutet es wenn man sagt ein Programm sei *ausführbar (executable)*?

Aufgabe 2

Bevor wir uns eingehender mit der Programmiersprache beschäftigen, ist es wichtig herauszufinden wie wir ein C-Programm auf unserem Computer kompilieren und ausführen lassen können. Die erforderlichen Schritte können je nach verwendeten Betriebssystem und eingesetzten Compiler sehr unterschiedlich sein. Einige Umgebungen stellen auch eigene Beispiele ähnlich unseres Programms aus Abschnitt 1.5 des Scripts zur Verfügung.

- Geben sie das "Hello World"-Programm aus Abschnitt 1.5 des Scripts, in den Computer ein, kompilieren sie es und führen es aus.
- Fügen sie eine zweite `printf()`-Anweisung hinzu, welche eine zweite Nachricht ausgibt. Irgend eine kurze Bemerkung, wie zum Beispiel, "How are you?" Speichern, kompilieren und führen sie das Programm erneut aus.
- Fügen sie eine Kommentarzeile zu ihrem Programm hinzu (wo immer sie wollen) und kompilieren sie das Programm erneut.

Führen Sie das Programm aus. Wie hat sich der Kommentar auf den Ablauf des Programms ausgewirkt?

Diese Übung mag ihnen trivial erscheinen, aber sie ist der Grundstein für all die vielen Programme, die wir in der nächsten Zeit entwickeln werden.

Um mit Vertrauen und Zuversicht die Eigenarten und Besonderheiten einer Programmiersprache zu entdecken ist es erforderlich, dass man Vertrauen in die Programmierumgebung hat. Es ist nämlich zum Teil sehr einfach die Übersicht darüber zu verlieren, welches Programm jetzt gerade bearbeitet, übersetzt und ausgeführt wird. Und es kann leicht vorkommen, dass sie versuchen den Fehler in einem Programm zu finden, während sie versehentlich ein anderes Programm ausführen, oder Änderungen in einem Programm noch gar nicht gespeichert wurden.

Das Hinzufügen und Ändern von Ausgabeanweisungen (`printf()`) ist ein einfacher Weg um herauszufinden, ob das Programm das sie ändern auch das Programm ist, dass sie ausführen.

Aufgabe 3

Es ist eine gute Idee sich mit einer Programmiersprache vertraut zu machen, indem man viele Sachen ausprobiert.

Wir können zum Beispiel in unser Programm ganz bewusst Fehler einbauen und beobachten ob der Compiler diese Fehler findet und wie er sie uns anzeigt. Manchmal wird der Compiler uns genau sagen, was falsch gelaufen ist und wie wir den Fehler beheben können. Manchmal bekommen wir nur eine unverständliche Meldung.

Durch einfaches Ausprobieren können wir uns einen Überblick verschaffen, wann wir dem Compiler trauen können und wann wir selbst herausfinden müssen was falsch gelaufen ist.

Nehmen Sie ein lauffähiges Programm und probieren sie nacheinander die folgenden Veränderung.

Achtung: Verändern Sie immer nur eine Stelle in ihrem Programm und lassen sie es danach ausführen. Machen Sie die Änderung rückgängig, bevor sie die nächste Änderung vornehmen.

- a. Entfernen Sie die schließende, geschweifte Klammer `}`.
- b. Entfernen Sie die öffnende, geschweifte Klammer `{}`.
- c. Entfernen Sie das `int` vor `main`.
- d. Anstelle von `main`, schreiben Sie `mian`.
- e. Entfernen Sie das schließende `*/` von einem Kommentar.
- f. Ersetzen Sie `printf` durch `pintf` im Quelltext des Programms.
- g. Löschen Sie eine der Klammern: `(` oder `)`
- h. Fügen Sie eine weitere Klammer hinzu.
- i. Löschen Sie das Semikolon nach der `return` Anweisung.