

## Übungsblatt 1: Programmieren in C (WS 2018/19)

Das erste Übungsblatt soll Sie mit grundlegenden Funktionalitäten vertraut machen, die Sie für die Bearbeitung aller weiteren Übungsblätter benötigen.

Beachten Sie bitte folgende Punkte, bevor Sie mit der Bearbeitung des Übungsblatts beginnen:

1. Sofern noch nicht geschehen, melden Sie sich bitte bis zum 27. Oktober 2018 im STATS an. Alle Details dazu sind auf der Homepage zur Vorlesung unter <https://softech.cs.uni-kl.de/homepage/de/teaching/WS18/pinc/> zu finden.
2. Zur Bearbeitung der Aufgaben an den Rechnern im Terminalraum, beantragen Sie bitte einen Account beim SCI (Gebäude 48, Erdgeschoss). Alternativ können Sie auch Ihren Laptop zu den Präsenzterminen mitbringen und die Aufgaben dort bearbeiten. Nähere Informationen zur Installation der benötigten Software finden Sie in den Folien zur Vorlesung.

**Zur Beantwortung von Fragen und Hilfe bei Problemen stehen wir Ihnen in der ersten Woche am Mittwoch und Donnerstag 17:00-18:30 sowie Freitag 15:30-17:00 im Terminalraum von Gebäude 32 (32-410) zur Verfügung. Nehmen Sie diese Gelegenheit wahr, um sich mit Ihrer Arbeitsumgebung und den Werkzeugen vertraut zu machen!**

## Hinweise zu den Übungen und Plagiaten

Programmieren lässt sich nur durch praktische Anwendung erlernen. Daher legen wir großen Wert darauf, dass Sie die Übungen selbst bearbeiten. Insbesondere werden Plagiate an der Universität nicht akzeptiert.

- Die Übungen werden in Teams von je 2 Studierenden bearbeitet. Wir erwarten, dass jeder im Team an der Lösung der Aufgaben mitarbeitet und dass Lösungen im Team diskutiert werden, so dass jeder im Team alle Aufgaben erklären kann.
- Einzureichende Abgaben sind Montags vor 12:00 Uhr mittags über das Exclaim System abzugeben.
- Sie können sich im Exclaim System unter <https://softech.cs.uni-kl.de/exclaim> mit Ihrem Stats-Account einloggen und Dateien zu den einzelnen Übungen hochladen.
- Laden Sie nur Dateien in den folgenden Formaten hoch: Textdatei (UTF-8) für Programmaufgaben sowie PDF, JPG oder PNG für andere Aufgaben. Verwenden Sie auf keinen Fall Formate, welche spezielle Programme benötigen (**kein** Microsoft Word, **kein** Open-/LibreOffice o.ä., **kein** Pages). Sie können Grafiken auch von Hand anfertigen und dann eine gescannte oder abfotografierte Kopie hochladen.
- Der abgegebene C Code muss dem C99-Standard entsprechen und darf nur die C Standard-Bibliothek verwenden (falls nicht anders angegeben). Wenn Sie nur die in der Vorlesung gezeigten Konstrukte verwenden, ist dies der Fall. Das Abgabesystem Exclaim verwendet die folgenden Optionen, um Ihre Abgaben zu übersetzen und auszuführen (dabei steht **xxx.c** für Ihre Abgabe):

```
clang -Wall -Werror -fsanitize=address -fsanitize=undefined -g xxx.c  
ASAN_OPTIONS=detect_leaks=1 ./a.out
```

Diese Optionen überprüfen Ihr Programm auf zusätzliche mögliche Fehler. Dazu gehören zum Beispiel auch Probleme mit Speicherzugriffen und undefiniertem Verhalten. Daher kann es sein, dass ein Programm zwar bei Ihnen ohne Fehler kompiliert und ausgeführt werden kann, aber vom Exclaim nicht akzeptiert wird, da es noch versteckte Probleme enthält.

Sie können diese Optionen selbst verwenden, wenn Sie den Clang-Compiler auf einem Linux oder Mac-Rechner installiert haben. Unter Windows werden die Optionen leider nicht alle unterstützt. Sie können aber die Tux-Rechner des SCI verwenden, auf denen die nötigen Tools bereits für Sie installiert sind. Auf diese Rechner können Sie sich übrigens auch von zu Hause verbinden (<https://sci.cs.uni-kl.de/rechnerzugang/remote/>).

- Wenn Sie sekundäre Quellen, wie Bücher oder das Internet verwenden, müssen Sie immer die Quelle angeben. Das einfache Kopieren aus anderen Quellen ist für die Übungen nicht gestattet. Wenn Sie andere Quellen benutzen, versuchen Sie diese erst zu verstehen und dann die Idee selbstständig umzusetzen. Den größten Lerneffekt erhalten Sie aber natürlich durch komplett eigenständiges Lösen der Aufgaben.

Wenn wir in einer Übungsabgabe kopierten Code finden, wird die gesamte Abgabe mit 0 Punkten bewertet.

- Sie können Übungsaufgaben gerne mit den Mitgliedern anderer Teams diskutieren. Sie sollten jedoch Ihren Code nie an andere Teams weitergeben.
- Wenn Code von anderen Teams kopiert wurde, werden die Abgaben **von beiden Teams** mit 0 Punkten bewertet.
- Wir behalten uns vor Punkte auch nachträglich abzuziehen, wenn ein Verstoß erst später bemerkt wird.

## 1 Übersetzen und Ausführen eines C Programms (Abgabe: Exclaim)

1. Schreiben Sie das folgende Programm in einem Text-Editor ab und speichern Sie es unter dem Dateinamen `hello.c` in einem Ordner `uebung_0`.

```
#include <stdio.h>

int main (void)
{
    printf ("Hello world!\n");
    return 0;
}
```

2. Übersetzen Sie das Programm und führen Sie es aus. Beachten Sie dazu den Hinweis am Ende des Übungsblattes.
3. Laden Sie Ihr Programm im Exclaim-System hoch.

## 2 Compiler-Ausgaben verstehen (Keine Abgabe)

Verändern Sie das Programm `hello.c` aus Aufgabe 1 wie folgt und übersetzen Sie es erneut. Speichern Sie die Änderungen unter den Dateinamen `hello1.c` bis `hello5.c`. Schreiben Sie für jede Teilaufgabe die Fehlermeldung ab als Kommentar am Anfang

des jeweiligen Programmtextes und erläutern Sie diese (bzw. erläutern Sie, weshalb keine Fehlermeldung auftritt).

- Ändern Sie `main` in `MAIN`.
- Lassen Sie die Zeile mit `\#include <stdio.h>` weg.
- Lassen Sie `return 0;` weg.
- Ändern Sie `"Hello world!\n"` nach `'Hello world!\n'`.
- Ändern Sie `printf ("Hello world!\n");` nach

```
printf ("Hello ,  
       world!\n");
```

Wie verändert sich die Ausgabe? Wie kann man das Programm ändern, so dass die Ausgabe in zwei Zeilen erfolgt?

## Hinweis zur Erstellung von C-Programmen

C-Programme werden mit einem Text-Editor erstellt und nicht mit einem Textverarbeitungsprogramm. Viele Texteditoren verfügen über so genanntes Text-Highlighting zur Unterstützung der Programmierer. Empfohlen werden können `gedit` und `gvim`. Sie können aber auch jederzeit einen anderen Texteditor Ihrer Wahl verwenden. Hier zwei ausgewählte Anleitungen:

<https://help.ubuntu.com/community/gedit>

<https://lug.fh-swf.de/vim/vim-kurzanleitung/vim-kurzanleitung.pdf>

Das C-Programm wird in einer Datei mit der Endung `.c` abgespeichert (z.B. `prog.c`), da dies vom C-Compiler so erwartet wird. Das fertige Programm wird unter Linux in einem Konsolen-Terminal mit der folgenden Anweisung compiliert:

```
gcc -o prog prog.c
```

`gcc` heißt dabei der GNU-C-Compiler und in der Datei `prog.c` steht der zu übersetzende Quellcode. Das übersetzte Programm wird dann ausgeführt durch Eingabe von `./prog`. Lässt man den Teil `-o prog` der Anweisung aus, dann steht das übersetzte Programm in der Datei `a.out`<sup>1</sup>. Bitte vergessen Sie nicht `./` zu Beginn der Programmnamens anzugeben! Diese Angabe ist wichtig in Bezug auf die Position im Dateisystem, d.h. wo das Programm zu finden ist.

Beide Kommandos können auch kombiniert werden durch folgende Eingabe:

```
gcc -o prog prog.c && ./prog
```

---

<sup>1</sup>Der Name `a.out` hat sich aus historischen Gründen etabliert.