

Übung 1 : Einführung in die Programmiersprache C

Vorbereitung: Vor der ersten Übungsstunde

- a) Installieren Sie eine C – Entwicklungsumgebung (z.B. Codeblocks). Unterstützung zu ausgewählten Entwicklungsumgebungen für Windows und Linux erhalten Sie im Lea Online-Kurs.
- b) Testen Sie die installierte Entwicklungsumgebung mit einem Hello World Programm. Dazu schreiben Sie eine C-Datei.

Um die Installation des Compilers zu prüfen, compilieren und linken Sie das Hello World-Programm zunächst von der Konsole, wie folgt beschrieben. Wenn dies erfolgreich war, prüfen Sie die Konfiguration ihrer eingesetzten IDE (integrated development environment), z.B. Codeblocks, und führen Sie den Compiler und Linker über die IDE aus.

I:

Dazu schreiben Sie eine C-Datei mit folgendem Inhalt:

```
# include <stdio.h>
int main(){
    printf("Hello World!");
    return 0;
}
```

Sie benennen die Datei hello.c. In einer Konsole (cmd.exe, command.com, bash, ...) wechseln Sie in das Verzeichnis, das ihre C-Datei enthält mit dem Befehl "cd".

Führen Sie nacheinander in der Kommandozeile drei Schritte aus.

1. Compilieren: gcc hello.c -c
2. Linken: gcc hello.o -o hello.exe
3. Ergebnis Ausführen: hello.exe

II:

Probieren Sie die graphische Menue-Führung ihrer IDE aus, um aus der C-Datei ein Programm generieren zu lassen.

- c) Verschaffen Sie sich einen Überblick über die C – Dokumentation und die freiwilligen Übungsaufgaben im Lea Online-Kurs.

Pflichtaufgaben:

Aufgabe 1: Einfache Funktionen

Sie werden gebeten, die Software eines einfachen Taschenrechners zu implementieren. Der Taschenrechner soll die vier Grundrechenarten unterstützen. Zudem möchte Ihr Auftragsgeber, dass der Taschenrechner die Funktionalität „Runden“ einer Fließkommazahl unterstützt.

Übung 1 : Einführung in die Programmiersprache C

- a) Erstellen Sie vier Funktionen *summe*, *differenz*, *produkt* und *quotient*, welche die entsprechenden mathematischen Operationen mit zwei übergebenen Zahlen ausführen und das Ergebnis zurückgeben.
- b) Schreiben Sie zusätzlich eine Funktion *runde*, die eine übergebene *float*-Zahl mathematisch korrekt auf eine Ganzzahl rundet und als *int* zurückgibt.
- c) Testen Sie die erstellten Funktionen in der *main*-Funktion (das Haupt-Programm) mit konkreten Werten. Geben Sie die Testergebnisse mit *printf* auf der Kommandozeile aus.

Aufgabe 2: Minimum und Maximum

- a) Erstellen Sie die Funktionen *max* und *min*, welche das Maximum bzw. Minimum von drei übergebenen Zahlen berechnen und zurückgeben.
- b) Testen Sie die in a) erstellten Funktionen in der *main*-Funktion mit konkreten Werten. Geben Sie die Testergebnisse mit *printf* auf der Kommandozeile aus.

Empfohlene Aufgaben:

Aufgabe 3: Größter gemeinsamer Teiler (ggT)

- a) Ein häufig zu lösendes Problem in der Mathematik ist das Finden eines größten gemeinsamen Teilers zweier ganzer Zahlen. Schreiben Sie eine Funktion, die den größten gemeinsamen Teiler zweier ganzer Zahlen findet und den Ergebniswert zurückgibt. Die Funktion soll folgenden Funktionskopf besitzen:

```
int ggT(int a, int b);
```

Hinweis: ggT von 34 und 85 ist 17.

- b) Schreiben Sie ein Test-Programm, das in seiner *main*-Funktion die *ggT*-Funktion von a) aufruft und die Eingabewerte und den Rückgabewert auf der Konsole ausgibt.

Aufgabe 4: Primzahlzwillinge

Schreiben Sie ein Programm welches alle Primzahlzwillinge zwischen 2 und 500 ausgibt. Primzahlzwillinge sind zwei Primzahlen p und q , für die gilt $|p-q| = 2$.

Gehen Sie dazu wie folgt vor:

1. Erstellen Sie eine Funktion *istPrim*, welche prüft ob eine Zahl p prim ist, oder nicht. Als Kriterium hierfür können Sie die Teilerfremdheit zu allen Zahlen von 2 bis $p-1$ verwenden ($\text{ggT}(p, <\text{zahl}>) == 1$). Nutzen Sie hier die Funktion *ggT* von Aufgabe 3a.
2. Definieren Sie in Ihrem Hauptprogramm eine *int*-Variable, in der Sie die erste Primzahl speichern. Verwenden Sie eine Schleife um alle Zahlen von 3 bis 500 zu durchlaufen. Prüfen Sie den Wert der *int*-Variable mit der nächsten Primzahl in der Schleife. Wenn die Bedingung für Primzahlzwillinge erfüllt ist, geben Sie die beiden Werte mit *printf* aus.