

6.8.A Extras/Modularisierung in C – Arbeitsblatt

Aufgabe 1: gcc-Aufruf für ein Programm mit mehreren Modulen

Das Programm unter `src/a1janein` bestand ursprünglich nur aus der Datei `voll.c`. Es wurde aufgeteilt in ein Modul mit den Dateien `janein.c` und `janein.h` und in ein Modul mit der Datei `main.c`.

- Die Datei `voll.c` hat weniger Bytes als die anderen Dateien zusammen. Woran könnte das liegen?
- Wie lauten die gcc-Aufrufe, wenn man die Dateien `main.c`, `janein.h` und `janein.c` zu einer ausführbaren Datei mit dem Namen `janeintest` zusammenführen will?
- Compilieren Sie `voll.c`, und nennen Sie das Ergebnis in `janeinvoll` um. Befreien Sie `janeintest` und `janeinvoll` mit Hilfe des Programms `strip` von überflüssigen Symbolen. Welche Datei ist größer?

Aufgabe 2: Nutzen eines Moduls durch ein Programm mit **main**-Funktion I

In dieser Aufgabe sollen Sie ein bestehendes Modul nutzen. Das Modul `sagwas` besteht aus den folgenden beiden Dateien:

`sagwas.c`

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 void sagzeit(void)
4 {
5     printf("Es_ist_");
6     fflush(stdout);
7     system("date");
8 }
```

`sagwas.h`

```
1 void sagzeit(void);
```

- Entwerfen Sie ein kurzes Modul `sagwas_main.c` mit einer `main`-Funktion, die die Funktion `sagzeit` aufruft!
- Erstellen Sie anschließend aus den beiden Modulen das Programm `sagwas_beispiel`! Wie lauten die gcc-Aufrufe?

Aufgabe 3: Nutzen eines Moduls II

Auch in dieser Aufgabe sollen Sie ein bestehendes Modul nutzen. Unter `src/a3myrandom` befindet sich das Modul `myrandom`.

- Erstellen Sie ein kurzes Modul `userandom.c` mit einer `main`-Funktion, die die Funktionen des Moduls `myrandom` aufruft und die Ergebnisse ausgibt!

- b) Erstellen Sie anschließend aus den beiden Modulen das Programm `myrandom_beispiel`! Wie lauten die `gcc`-Aufrufe?

Aufgabe 4: Aufteilen eines Programmes in Module

Im Verzeichnis `src/a4haeufigkeit` findet man eine Quelltextdatei mit einem kompletten Programm.

- a) Welchen Programmteil kann man in ein Modul auslagern?
- b) Teilen Sie das Programm auf in ein Modul mit `main` und ein Modul, das aus Inhalten besteht, die ausgelagert wurden.
- c) Erstellen Sie anschließend aus den beiden Modulen das Programm `haeufigkeit_modular`! Wie lauten die `gcc`-Aufrufe?

Aufgabe 5: Schreiben eines Programmes mit mehreren Modulen

Nun sollen Sie selbst ein Programm mit mehreren Modulen schreiben.

- a) Das Modul `zweihoch` soll eine Funktion `int zweihoch(int x, int y)` enthalten, die den Wert x^y zurückgibt.
- b) Das Modul `fakultaet` soll eine Funktion `int fakultaet(int x)` enthalten, die den Wert $x! = x \cdot (x-1) \cdot (x-2) \cdot \dots \cdot 1$ zurückgibt.
- c) Im Hauptmodul sollen beide Funktionen aufgerufen werden.
- d) Erstellen Sie anschließend aus den beiden Modulen das Programm `mathe`!