



# Grundzüge der objektorientierten Programmierung

WS2001/2002 – Übungsblatt 6

Abgabetermin: 10. Dezember 2001

## **Aufgabe 1.** *CT\_ASSERT, 5 Punkte*

Benutzen Sie das Makro `CT_ASSERT()` der Vorlesung, um mittels der in `climits`, `cstdint`, `cfloat`, ... definierten Konstanten abzusichern, daß Ihr Compiler für `long double` größere Genauigkeit anbietet als für `double`.

## **Aufgabe 2.** *Textdateien, 5 Punkte*

Schreiben Sie ein Programm, das den Nutzer nach dem Namen einer Textdatei fragt, diese dann öffnet, die Anzahl der darin vorhandenen Zeichen zählt und diese zusammen mit einem Kontrollausdruck der Textdatei auf dem Bildschirm ausgibt.

## **Aufgabe 3.** *Templatefunktionen und Runtimefehler, 5 Punkte*

Schreiben Sie eine Templatefunktion `T average(const T v[], int length)`, die das arithmetische Mittel der Komponenten des Vektors `v` der Länge `length` ermittelt. Testen Sie in einem Hauptprogramm.

Vergessen Sie nicht einen Fehlerabbruch zu produzieren, wenn `length` unsinnige Werte erhält.

## **Aufgabe 4.** *cassert, 5 Punkte*

Schreiben sie eine Funktion `long int fakultaet(int n)` zur Berechnung der Fakultät; benutzen Sie dabei eine Schleife, *keine* Rekursion (warum?).

Erzeugen Sie eine Runtimefehlermeldung bei unsinnigen Werten von `n`. Benutzen Sie dazu zunächst eine Ausgabe auf `cerr` und Programmabbruch mittels `exit()` und dann die Funktion `assert()`. Testen Sie jeweils.

Schalten Sie danach die Auswertung der Zusicherungen mittels `NDEBUG` mal ein und mal aus. Wie geht das?