



# Lehrveranstaltung für TI-Bachelor: "Vertiefung objektorientierte Programmierung in C++" Übungsblatt 1

## Hinweise:

Dieses Übungsblatt ist zur Zulassung zu der Klausur erfolgreich zu bearbeiten ("Erfolgreich" bedeutet: Keine Programmabstürze bzw. Endlosschleifen, Aufgabenstellung einschl. der Nebenbedingungen müssen eingehalten sowie Kommentierung und Einrückung korrekt sein!).

Die Aufgaben werden überwiegend in den Übungszeiten bearbeitet. Allerdings genügt die Zeit hierfür unter Umständen nicht, so dass Sie auch außerhalb dieser Zeiten die Aufgaben bearbeiten müssen. Der Abgabetermin für diese Aufgabe ist **spätestens** der **18. Oktober 2012**.

Nutzen Sie die Übungen auch, um ggf. Fragen, die sich in den Vorlesungen ergeben haben, anzusprechen.

**Aufgabe:** Zuerst erhalten Sie Ihren Account sowie eine kurze Einführung für die PCs im Labor.

In den Übungsaufgaben soll ein Programm entstehen, das Banken mit Konten und Buchungen abbildet. Dazu wollen wir zuerst gemeinsam uns ein Überblick über das Projekt verschaffen und dabei ermitteln, welche Klassen und Strukturen benötigt werden.

Als erste Klasse soll dann eine Datums-Klasse `TDate` (alle Bezeichnungen sollen möglichst in Englisch geschrieben werden) erstellt werden. Hier werden Tag, Monat und Jahr als private Eigenschaften angelegt und dazu Methoden zum Setzen eines beliebigen / des heutigen Datums, zum Abfragen der Eigenschaften sowie zum Ausgeben des Datums im Format "tt.mm.jjjj" (jeweils mit führenden Nullen) erstellt. Ferner sollen zwei Konstruktoren implementiert werden: Der erste erhält keinen Parameter und setzt das Datum auf das heutige Datum, der zweite Konstruktor erhält drei Zahlen, die als Tag, Monat und Jahr in dem Objekt gesetzt werden.

Analog dazu soll eine Klasse eine Uhrzeit-Klasse `TTime` mit Stunden, Minuten und Sekunden als private Eigenschaften und mit Methoden zum Setzen und Abfragen sowie zur Ausgabe (Format "hh:mm:ss" jeweils mit führenden Nullen) der Uhrzeit erstellt werden.

Zum Ermitteln des aktuellen Datums und der aktuellen Uhrzeit können die Funktionen `time` und `localtime` aus der Headerdatei `<ctime>` verwendet werden.

Zum Testen der Klassen soll ein kleines Hauptprogramm geschrieben bzw. kann das vorgegebene Beispielprogramm verwendet werden, dass Objekte der Datums- und Uhrzeit-Klassen erzeugt, auf verschiedene Werte setzt und wieder auf dem Bildschirm ausgibt.

### Beispiel für ein Hauptprogramm:

```
#include <iostream>

using namespace std;

#include "tdate.h"
#include "ttime.h"

int main()
{
    TDate D1, D2(3, 10, 2012);
    TTime T1, T2(16, 0);

    cout << "\nKlasse TDate:" << endl;
    cout << "Standardkonstruktor Heutiges Datum D1:      "; D1.print(); cout << endl;
    cout << "Konstruktor          Tag der Einheit (D2):      "; D2.print(); cout << endl;

    cout << "\nKlasse TTime:" << endl;
    cout << "Standardkonstruktor Aktuelle Uhrzeit T1:      "; T1.print(); cout << endl;
    cout << "Konstruktor          Uebungsbeginn (T2):      "; T2.print(); cout << endl;

    return 0;
}
```

### Beispiel-Ausgabe:

```
Klasse TDate:
Standardkonstruktor Heutiges Datum D1:      04.10.2012
Konstruktor          Tag der Einheit (D2):   03.10.2012

Klasse TTime:
Standardkonstruktor Aktuelle Uhrzeit T1:     18:25:33
Konstruktor          Vorlesungsbeginn (T2):  16:00:00
```