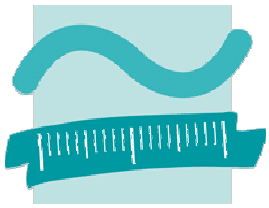
**Dipl.Phys. Gerald Kempfer**

Beuth Hochschule für Technik Berlin
- University of Applied Sciences -
Fachbereich VI – Informatik und Medien
Studiengang Technische Informatik Bachelor



# Lehrveranstaltung "Informatik II für TI-Bachelor"
## Übungsblatt 4

**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 **31. Mai 2013**.

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

---

**Aufgabe:**

In der vierten Übungsaufgabe sollen in der Terminverwaltung die Daten beim Beenden des Programmes in einer Datei gespeichert werden, damit sie beim nächsten Programmstart wieder eingelesen werden können.

Für das Laden und Speichern der Daten sollen einige Funktionen in einem neuen Modul `files.c` erstellt werden. Erzeugen Sie das Modul samt Headerdatei und passen Sie das Projekt an.

Für das Speichern der Daten soll im neuen Modul die Funktion `saveCalendar` geschrieben werden. Hier soll nach dem erfolgreichen Öffnen der Datei die erste Zeile mit der Startkennung der Daten (`<Calendar>`; siehe Beispiel) geschrieben werden. Als nächstes wird in einer eigenen Zeile die Anzahl der Termine in die Datei geschrieben, z.B. `<AppointmentCount>5</AppointmentCount>`. Dann werden in einer Schleife die Daten der Termine geschrieben; das Schreiben eines Termins kann eine Funktion namens `saveAppointment` übernehmen. Nach der Startkennung eines Termins (`<Appointment>`) werden die Daten des Termins (Datum, Uhrzeit, Beschreibung, Ort und Dauer) jeweils mit Start- und Endkennung in eine Zeile geschrieben (z.B: `<Date>17.05.2013</Date>`). Nach den Feldern des Termins wird die Endkennung (`</Appointment>`) in einer eigenen Zeile geschrieben. Nachdem alle Termine geschrieben wurden, wird noch die Endkennung der Daten (`</Calendar>`) geschrieben.

Für das Laden der Daten soll eine Funktion `loadCalendar` geschrieben werden. In dieser Funktion sollen nach dem erfolgreichen Öffnen der Datei die Daten eingelesen werden. Immer wenn die Startkennung eines Termins gefunden wird, soll die Funktion `loadAppointment` aufgerufen

werden. Diese liest die Daten dieses einen Termins ein (bis zur Endkennung `</Appointment>`).

Durch die Start- und Endkennungen können die Felder in beliebiger Reihenfolge stehen. Ferner kann darüber beim Einlesen geprüft werden, ob die Struktur der Daten gültig ist. Um die Datenbank kontrollieren zu können, sollte auf eine Verschlüsselung der Daten verzichtet werden. Überlegen Sie sich, welche Maßnahmen noch nötig sind, um angemessen auf eine fehlerhafte Datenbank zu reagieren. Und was muss getan werden, wenn in einem Termin-Datensatz ein Feld fehlt (z.B: in der Beispieldatenbank fehlen im ersten und im letzten Termin verschiedene Felder), damit dieser Datensatz trotzdem noch korrekt eingelesen und angezeigt werden kann.

Wem diese ganze Beschreibung zu kompliziert ist, guckt sich am besten erst das unten stehende Beispiel an; dies sollte vieles erklären.

Noch ein paar Hinweise:
1. Nach dem Laden und Speichern sollte die Datei natürlich jeweils wieder geschlossen werden.
2. Das Laden der Daten erfolgt bei Programmstart; das Speichern (optional mit Benutzerabfrage) bei Programmende. Wer möchte, kann alternativ Laden und Speichern als Menüpunkte einbauen (das Laden sollte dann die Daten zu den bereits eingegebenen Daten hinzufügen!).
3. Bei Programmstart (noch vor dem Einlesen der Daten) sollten alle Datensätze mit Nullen (0 bzw. NULL) gefüllt werden.
4. Bei Programmende müssen natürlich (sofern nicht bereits in der vorigen Übungsaufgabe erledigt) alle reservierten Speicherbereiche wieder freigegeben werden!
5. Beim Einlesen kann immer eine ganze Zeile eingelesen werden. Um herauszufinden, welche Daten in der Zeile stehen (dieser Vorgang wird „parsen“ genannt), kann z.B. die Funktion `strncmp` aus der `string.h` verwendet werden; z.B. wird mit

```
if (strncmp(Zeile, "<Date>", 6) == 0)
```

geprüft, ob die ersten 6 Zeichen in der Zeichenkette `Zeile` gleich `<Date>` sind.

Generell soll immer möglichst mit Zeigern anstelle von Arrays gearbeitet werden!

Kommentieren Sie das Programm. Dazu gehören auch Modul- und Funktionsheader (siehe Skript „Grundlagen der Informatik“ Kapitel 5.3 und 5.4)! Achten Sie auch auf Ihre Programmstruktur (Einrückungen; Leerzeichen und Leerzeilen, usw.).

## Beispieldatenbank

```
<Calendar>
 <AppointmentCount>5</AppointmentCount>
 <Appointment>
  <Date>10.05.2013</Date>
  <Description>ausschlafen</Description >
 </Appointment>
 <Appointment>
  <Date>17.05.2013</Date>
  <Time>08:00</Time>
  <Description>IN 2 - Uebung Gruppe 1</Description >
  <Location>Raum D 113</Location>
  <Duration>01:30</Duration>
 </Appointment>
 <Appointment>
  <Date>17.05.2013</Date>
  <Time>10:00</Time>
  <Location>Raum H5</Location>
  <Duration>01:30</Duration>
  <Description>IN 2 - Vorlesung</Description >
 </Appointment>
 <Appointment>
  <Time>12:15</Time>
  <Description>IN 2 - Uebung Gruppe 2</Description >
  <Location>Raum D 113</Location>
  <Date>17.05.2013</Date>
  <Duration>01:30</Duration>
 </Appointment>
 <Appointment>
  <Date>18.05.2013</Date>
  <Time>18:00</Time>
  <Description>Party</Description >
  <Duration>10:00</Duration>
 </Appointment>
</Calendar>
```

## Ausgabe der Beispieldatenbank:

```
Liste der Termine
=================

--------------------------------------------------------------------------------
Freitag, der 10.05.2013:
------------------------------
        ->                | Ausschlafen
--------------------------------------------------------------------------------
Freitag, der 17.05.2013:
------------------------------
   08:00 -> D 113         | IN2 Uebung Gruppe 1
   10:00 -> H5            | IN2 Vorlesung
   12:15 -> D 113         | IN2 Uebung Gruppe 2
--------------------------------------------------------------------------------
Samstag, der 18.05.2013:
------------------------------
   18:00 ->               | Party
--------------------------------------------------------------------------------
Bitte Eingabetaste druecken ...
```