



Lehrveranstaltung "Informatik II für TI-Bachelor"

Übungsblatt 6

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 **28. Juni 2013**.

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

Aufgabe: In der sechsten Übungsaufgabe soll in der Terminverwaltung die Datenstruktur geändert werden: Anstelle vom Array von `TAppointment` sollen die Termine in einer doppelt verketteten Liste gespeichert werden.

Folgende Anpassungen müssen für diese Übung durchgeführt werden:

`datastructure.h`:

Die Konstante

```
#define MAXAPPOINTMENTS 100
```

kann gelöscht werden. Entsprechend müssen alle Vorkommen dieser Konstante beseitigt werden. Das externe Array

```
extern TAppointment Calendar[];
```

wird ersetzt durch

```
extern TAppointment *First, *Last;
```

die globale Variable `AppointmentCount` kann (muss aber nicht) bestehen bleiben.

In der Struktur `TAppointment` kommen `Next`- und `Prev`-Zeiger auf die eigene Struktur hinzu.

Neu hinzu kommt das Modul `list.c` (mit Headerdatei `list.h`). Hier werden die Funktionen `insertInList` (zum sortierten Einfügen in die verkettete Liste) sowie `removeFromList` (zum Entfernen aus der verketteten Liste) erstellt. Die verkettete Liste soll immer nach Datum und Uhrzeit sortiert sein; hierfür kann die Vergleichsfunktion vom Sortieren nach Datum und Uhrzeit wiederverwendet werden.

Nun müssen die Module `calendar.c` (neuen Termin eingeben, Terminliste ausgeben und bei Programmende wieder alle Termine löschen) und `files.c` (Laden und Speichern der Daten bei

gleichbleibendem Dateiaufbau) an die Listenstruktur angepasst werden; das Sortieren der Termine entfällt. Die Bildschirmausgaben bleiben unverändert, d.h. der Anwender sollte die Änderung der Datenstruktur nicht erkennen können.

Generell soll immer 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.).