

---

## Lehrveranstaltung "Objektorientierte Programmierung"

### Übungsblatt 7

#### 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!).

Der Abgabetermin für diese Aufgabe ist der **20. Januar 2023**.

---

**Aufgabe:** In der 7. Übungsaufgabe des Projektes „Belegungen“ soll zum Einen der Vektor `Persons` in der Klasse `CBookings` in eine doppelt verkettete Liste mit Iteratoren geändert werden. Zum Anderen sollen einige Operatoren überladen werden.

Die vorgegebene Headerdatei `clist.h` wird in der `cbookings.cpp` per `#include` eingebunden. Der Vektor `vector<CPerson *> Persons;` wird dann in `CList<CPerson *> Persons` geändert. Alle for-Schleife der Personen sollen nun mittels der in der Klasse `CList` eingebetteten Iteratoren durchlaufen werden. Zum Vergleich soll mindestens ein Vektor auch mit Iteratoren durchlaufen werden.

In `CBookings` soll der Funktionsoperator überladen werden, mit dem jeweils eine der verschiedenen Ausgaben aufgerufen werden soll. Als Parameter bekommt dieser Operator einen Wert eines Aufzählungstyps, der außerhalb der Klasse in `cbookings.h` (ähnlich wie `CWeekday`) definiert werden und folgende Werte beinhalten soll (dabei steht „of“ jeweils für „OutputFormat“):

- `ofPersons` (alle Personen)
- `ofStudents` (alle Studenten inkl. Tutoren)
- `ofTeachers` (alle Dozenten inkl. Tutoren)
- `ofBookings` (alle Belegungen)

Die folgenden beiden Formate sind freiwillig (siehe weiter unten):

- `ofScheduleOfStudents` (Stundenpläne von allen Studenten / Tutoren)
- `ofScheduleOfStudies` (Stundenpläne von allen Studiengängen).

Die Details zu den Ausgabeformaten entnehmen Sie bitte der Beispiel-Ausgabe.

In der Klasse `CPerson` und in den davon abgeleiteten Klassen `CStudent`, `CTeacher` und `CTutor` sollen unter Berücksichtigung des Polymorphismus die Ausgabe-Operatoren überladen werden.

Dann soll für die Klasse `CTime` ein `+`-Operator überladen werden. Mit diesem Operator soll zu einer Uhrzeit eine beliebige Anzahl von Minuten hinzuaddiert werden können. Dies soll bei der Berechnung des Endes ei-

nes Blocks zum Einsatz kommen, z.B.

Ende = Anfang + 90;

Dabei sind Ende und Anfang vom Typ CTime; der Operator soll jetzt im Destruktor von CBlock zum Einsatz kommen (siehe Beispiel-Ausgabe).

### Freiwillig:

Wer möchte, kann auch noch den Stundenplan (Klasse CSchedule) realisieren.

Die Klasse CSchedule bekommt als private Eigenschaft einen Titel und ein zweidimensionales Array von Zeigern auf die Events (5 Spalten für die 5 Werktage und 7 Zeilen für die 7 Blöcke am Tag). Der Standard-Konstruktor soll vom Compiler automatisch erstellt werden; es wird dann noch eine reset-Methode benötigt, die den Titel angibt und das Array mit nullptr-Zeigern füllt. Ferner soll für diese Klasse der Ausgabeoperator überladen werden (siehe Beispielausgabe).

In der Klasse CBookings wird dann ein Objekt der Klasse CSchedule hinzugefügt. Bei Aufruf des Funktionsoperators mit den Ausgabeformaten ofScheduleOfStudents und ofScheduleOfStudies sollen dann die Stundenpläne wie in der Beispiel-Ausgabe ausgegeben werden. Da die Ausgabe sehr breit ist (mehr als 80 Zeichen), kann die Ausgabe auch in eine Textdatei umgeleitet und dann betrachtet werden.

Zum Testen der Klassen soll das vorgegebene Hauptprogramm verwendet werden, dass Objekte der angegebenen Klassen erzeugt, auf verschiedene Werte setzt und wieder auf dem Bildschirm ausgibt.

### vorgegebenes Hauptprogramm:

```
#include <iostream>

using namespace std;

#include "cbookings.h"

int main()
{
    CBookings Belegungen("belegungen.xml");

    Belegungen(ofPersons);
    Belegungen(ofStudents);
    Belegungen(ofTeachers);
    Belegungen(ofBookings);
    // freiwillig:
    // Belegungen(ofScheduleOfStudents);
    // Belegungen(ofScheduleOfStudies);

    return 0;
}
```

### Beispiel-Ausgabe:

Datei wurde erfolgreich eingelesen!

```
Personen:
Anna Albrecht (* 15.09.1989; ID 1; MatrNr. 191278)
Sabine Sauber (* 03.01.1962; ID 2; PersNr. 68)
Dettlef Dicht (* 27.05.1991; ID 3; MatrNr. 191115)
Frank Faust (* 03.11.1987; ID 4; MatrNr. 191492)
Kevin Kaufmann (* 15.06.1959; ID 5; PersNr. 12)
Nathan Neuling (* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)
Willi Witzig (* 09.11.1976; ID 7; PersNr. 34)
```

Studenten:

Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
Detlef Dicht (\* 27.05.1991; ID 3; MatrNr. 191115)  
Frank Faust (\* 03.11.1987; ID 4; MatrNr. 191492)  
Nathan Neuling (\* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)

Dozenten:

Sabine Sauber (\* 03.01.1962; ID 2; PersNr. 68)  
Kevin Kaufmann (\* 15.06.1959; ID 5; PersNr. 12)  
Nathan Neuling (\* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)  
Willi Witzig (\* 09.11.1976; ID 7; PersNr. 34)

Belegungen:

Belegung Nr. 1:

Am 15.10.2022 um 08:12:27  
hat Detlef Dicht (\* 27.05.1991; ID 3; MatrNr. 191115)  
das Fach 'OOP (31101; Technische Informatik)' belegt.

Belegung Nr. 2:

Am 15.10.2022 um 09:53:49  
hat Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
das Fach 'Mathe III (31103; Technische Informatik)' belegt.

Belegung Nr. 3:

Am 16.10.2022 um 22:07:31  
hat Frank Faust (\* 03.11.1987; ID 4; MatrNr. 191492)  
das Fach 'OOP (31101; Technische Informatik)' belegt.

Belegung Nr. 4:

Am 16.10.2022 um 22:11:53  
hat Frank Faust (\* 03.11.1987; ID 4; MatrNr. 191492)  
das Fach 'Systemprogr. (33102; Technische Informatik)' belegt.

Belegung Nr. 5:

Am 16.10.2022 um 22:18:29  
hat Frank Faust (\* 03.11.1987; ID 4; MatrNr. 191492)  
das Fach 'Englisch (31209; Technische Informatik)' belegt.

Belegung Nr. 6:

Am 16.10.2022 um 22:23:03  
hat Frank Faust (\* 03.11.1987; ID 4; MatrNr. 191492)  
das Fach 'Mathe III (31103; Technische Informatik)' belegt.

Belegung Nr. 7:

Am 15.10.2022 um 08:17:21  
hat Detlef Dicht (\* 27.05.1991; ID 3; MatrNr. 191115)  
das Fach 'Physik III (33119; Technische Informatik)' belegt.

Belegung Nr. 8:

Am 17.10.2022 um 19:21:03  
hat Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
das Fach 'OOP (31101; Technische Informatik)' belegt.

Belegung Nr. 9:

Am 17.10.2022 um 23:46:12  
hat Nathan Neuling (\* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)  
das Fach 'OOP (31101; Technische Informatik)' belegt.

Belegung Nr. 10:

Am 17.10.2022 um 23:52:53  
hat Nathan Neuling (\* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)  
das Fach 'Systemprogr. (33102; Technische Informatik)' belegt.

Belegung Nr. 11:

Am 17.10.2022 um 23:59:07  
hat Nathan Neuling (\* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)  
das Fach 'Englisch (31209; Technische Informatik)' belegt.

Belegung Nr. 12:

Am 17.10.2022 um 23:59:18  
hat Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
das Fach 'Physik III (33119; Technische Informatik)' belegt.

Belegung Nr. 13:

Am 17.10.2022 um 23:59:21  
hat Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
das Fach 'Englisch (31209; Technische Informatik)' belegt.

Belegung Nr. 14:

Am 17.10.2022 um 23:59:34  
hat Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
das Fach 'Systemprogr. (33102; Technische Informatik)' belegt.

Belegung Nr. 15:

Am 17.10.2022 um 23:59:41  
hat Anna Albrecht (\* 15.09.1989; ID 1; MatrNr. 191278)  
das Fach 'Elektrische Systeme (33109; Technische Informatik)' belegt.

Belegung Nr. 16:  
 Am 17.10.2022 um 23:59:47  
 hat Detlef Dicht (\* 27.05.1991; ID 3; MatrNr. 191115)  
 das Fach 'Mathe III (31103; Technische Informatik)' belegt.

Belegung Nr. 17:  
 Am 17.10.2022 um 23:59:59  
 hat Nathan Neuling (\* 09.09.1969; ID 6; MatrNr. 191919; PersNr. 101)  
 das Fach 'Elektrische Systeme (33109; Technische Informatik)' belegt.

## Folgende Ausgaben der Stundenpläne sind freiwillig:

Stundenpläne der Studenten:

Stundenplan fuer Anna Albrecht:

	Mo	Di	Mi	Do	Fr
1					
2		SU Englisch Sabine Sauber D 419		UEB Mathe III Nathan Neuling D 209	
3		SU Systemprogr. Willi Witzig D 117 (H3)			
4	SU ES Willi Witzig D 136 (H5)	UEB Systemprogr. Willi Witzig D 114			SU OOP Kevin Kaufmann D 136 (H5)
5	UEB ES Willi Witzig D 114		SU 31209 Sabine Sauber D 419		UEB OOP Kevin Kaufmann D 113
6	SU Phy III Sabine Sauber D 136 (H5)				UEB OOP Kevin Kaufmann D 113
7	UEB Phy III Willi Witzig D 113				

Stundenplan fuer Detlef Dicht:

	Mo	Di	Mi	Do	Fr
1					
2				UEB Mathe III Nathan Neuling D 209	
3					
4					SU OOP Kevin Kaufmann D 136 (H5)
5					UEB OOP Kevin Kaufmann D 113
6	SU Phy III Sabine Sauber D 136 (H5)				UEB OOP Kevin Kaufmann D 113
7	UEB Phy III Willi Witzig D 113				

Stundenplan fuer Frank Faust:

	Mo	Di	Mi	Do	Fr
1					
2		SU Englisch Sabine Sauber		UEB Mathe III Nathan Neuling	

		D 419		D 209	
3		SU Systemprogr. Willi Witzig D 117 (H3)			
4		UEB Systemprogr. Willi Witzig D 114			SU OOP Kevin Kaufmann D 136 (H5)
5			SU 31209 Sabine Sauber D 419		UEB OOP Kevin Kaufmann D 113
6					UEB OOP Kevin Kaufmann D 113
7					

Stundenplan fuer Nathan Neuling:

	Mo	Di	Mi	Do	Fr
1					
2		SU Englisch Sabine Sauber D 419			
3		SU Systemprogr. Willi Witzig D 117 (H3)			
4	SU ES Willi Witzig D 136 (H5)	UEB Systemprogr. Willi Witzig D 114			SU OOP Kevin Kaufmann D 136 (H5)
5	UEB ES Willi Witzig D 114		SU 31209 Sabine Sauber D 419		UEB OOP Kevin Kaufmann D 113
6					UEB OOP Kevin Kaufmann D 113
7					

Stundenplaene der Studiengaenge:

Stundenplan fuer Technische Informatik:

	Mo	Di	Mi	Do	Fr
1					
2		SU Englisch Sabine Sauber D 419		UEB Mathe III Nathan Neuling D 209	
3		SU Systemprogr. Willi Witzig D 117 (H3)			
4	SU ES Willi Witzig D 136 (H5)	UEB Systemprogr. Willi Witzig D 114			SU OOP Kevin Kaufmann D 136 (H5)
5	UEB ES Willi Witzig D 114		SU 31209 Sabine Sauber D 419		UEB OOP Kevin Kaufmann D 113
6	SU Phy III Sabine Sauber D 136 (H5)				UEB OOP Kevin Kaufmann D 113
7	UEB Phy III Willi Witzig D 113				

## Die Ausgaben der Destruktoren ist wieder Pflicht:

```
Bloecke freigeben ...
  Block Nr. 1 (08:00-09:30) wird vernichtet.
  Block Nr. 2 (10:00-11:30) wird vernichtet.
  Block Nr. 3 (12:15-13:45) wird vernichtet.
  Block Nr. 4 (14:15-15:45) wird vernichtet.
  Block Nr. 5 (16:00-17:30) wird vernichtet.
  Block Nr. 6 (17:45-19:15) wird vernichtet.
  Block Nr. 7 (19:30-21:00) wird vernichtet.
Bloecke freigeben ok
Raeume freigeben ...
  Raum D 113 wird vernichtet.
  Raum D 114 wird vernichtet.
  Raum D 117 (H3) wird vernichtet.
  Raum D 136 (H5) wird vernichtet.
  Raum D 209 wird vernichtet.
  Raum D 419 wird vernichtet.
  Raum B 301 wird vernichtet.
Raeume freigeben ok
Studiengaenge freigeben ...
  Studiengang Technische Informatik wird vernichtet.
Studiengaenge freigeben ok
Personen freigeben ...
  Student*in Anna Albrecht wird vernichtet.
  Person Anna Albrecht wird vernichtet.
  Lehrer*in Sabine Sauber wird vernichtet.
  Person Sabine Sauber wird vernichtet.
  Student*in Detlef Dicht wird vernichtet.
  Person Detlef Dicht wird vernichtet.
  Student*in Frank Faust wird vernichtet.
  Person Frank Faust wird vernichtet.
  Lehrer*in Kevin Kaufmann wird vernichtet.
  Person Kevin Kaufmann wird vernichtet.
  Tutor freigegeben: Nathan Neuling
  Lehrer*in Nathan Neuling wird vernichtet.
  Student*in Nathan Neuling wird vernichtet.
  Person Nathan Neuling wird vernichtet.
  Lehrer*in Willi Witzig wird vernichtet.
  Person Willi Witzig wird vernichtet.
Personen freigeben ok
Faecher freigeben ...
  Studienfach OOP wird vernichtet.
  Studienfach Mathe III wird vernichtet.
  Studienfach Englisch wird vernichtet.
  Studienfach Systemprogr. wird vernichtet.
  Studienfach Elektrische Systeme wird vernichtet.
  Studienfach Physik III wird vernichtet.
Faecher freigeben ok
Belegungen freigeben ...
  Belegung Nr. 1 wird vernichtet.
  Belegung Nr. 2 wird vernichtet.
  Belegung Nr. 3 wird vernichtet.
  Belegung Nr. 4 wird vernichtet.
  Belegung Nr. 5 wird vernichtet.
  Belegung Nr. 6 wird vernichtet.
  Belegung Nr. 7 wird vernichtet.
  Belegung Nr. 8 wird vernichtet.
  Belegung Nr. 9 wird vernichtet.
  Belegung Nr. 10 wird vernichtet.
  Belegung Nr. 11 wird vernichtet.
  Belegung Nr. 12 wird vernichtet.
  Belegung Nr. 13 wird vernichtet.
  Belegung Nr. 14 wird vernichtet.
  Belegung Nr. 15 wird vernichtet.
  Belegung Nr. 16 wird vernichtet.
  Belegung Nr. 17 wird vernichtet.
Belegungen freigeben ok
```