Lehrstuhl für Informatik 3
Rheinisch-Westfälische Technische Hochschule
Aachen
Univ.-Prof. Dr.-Ing. M. Nagl

Aachen, 06. 07. 2007

Nagl, Ranger, Wörzberger

Übungen zur Vorlesung „Grundgebiete der Informatik 2: Algorithmen und Programmiertechniken“

— Blatt 13 —

Dies ist Zusatzaufgabenblatt, das nicht abgegeben und korrigiert wird. Die Lösung befindet sich auf der Website. Ein selbständiger Lösungsversuch ist für die Klausurvorbereitung *dringend* empfohlen. Die Besprechung mit Lösungsvorschlag erfolgt am 18.07.2006 beim zweiten Tutoriumstermin.

30. Aufgabe *Generische doppelt verkettete Liste*:

In Aufgabe 21 sollten Sie einen ADT für eine doppelt verkettete Liste implementieren. Diese kann allerdings als Werte nur ganze Zahlen (`int`) aufnehmen. In dieser Aufgabe soll nun eine generische doppelt verkettete Liste `GenDLList` implementiert werden, die Werte beliebigen Typs aufnehmen kann.

Laden Sie sich die Dateien `DLList.h` und `DLList.cc` herunter. Ändern Sie beide so, dass sie einen ADT `GenDLList` einer generischen doppelt verketteten Liste erhalten, der Werte beliebigen Typs aufnehmen kann.

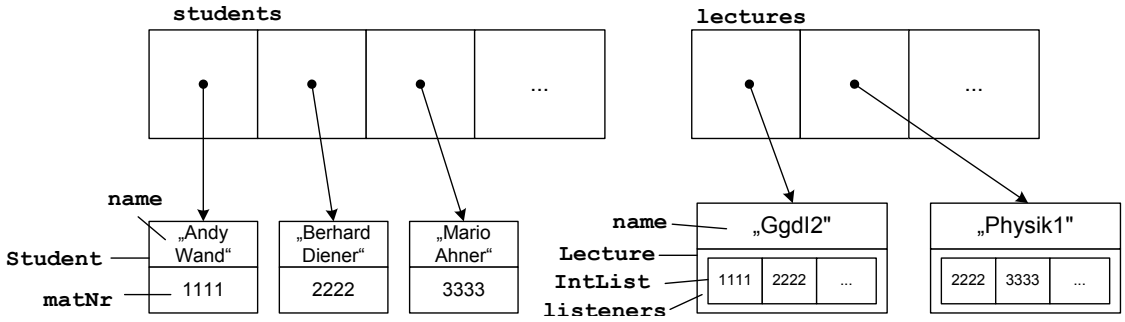
Hinweis: Für syntaxbezogene Fragen (z.B. “Wo müssen `template` und die generischen Parameter stehen?”) schauen Sie sich die Code-Rahmen bzw. Lösungen aus der Aufgabe 27 an. Sie müssen außerdem in beiden zu erstellenden Dateien `GenDLList.h` *und* `GenDLList.cc` per `#ifndef` und `#define` dafür sorgen, dass der Code nur einmal eingebunden wird. Auch das können Sie sich aus den Rahmen in `DynamicBuffer.h` und `DynamicBuffer.cc` der Aufgabe 27 abschauen.

31. Aufgabe *Vorlesungsverwaltung*:

In dieser Aufgabe soll eine einfache Vorlesungsverwaltung realisiert werden. Dazu ist ein ADT `LecAdm` als Klasse zu realisieren, der die auf der Rückseite im öffentlichen Teil angegebene Schnittstelle besitzt. Die interne Realisierung des ADT basiert auf mehreren Listen verschiedener Typ-Instanzen der generischen Liste. Die Liste `students` vom Typ `StudentList` beinhaltet *Zeiger* auf Studenten-Datenobjekte, die Liste `lectures` vom Typ `LectureList` Zeiger auf Vorlesungs-Datenobjekte. Studenten-Datenobjekte sind vom Typ `Student` und bestehen aus einem `string` für den Studentennamen und einer ganzen Zahl für die Matrikelnummer. Vorlesungs-Datenobjekte sind vom Typ `Lecture` und bestehen aus einem `string` für den Vorlesungsnamen und wiederum aus einer Liste aus Matrikelnummern (Typ `IntListe`, also mit ganzen Zahlen als Einträgen) für teilnehmende Studenten.

```
#pragma once
#include "stdafx.h"

#include "GenDLList.cc" // Nicht 'GenDLList.h' einbinden
#include <string>

class LecAdm {
public:
    // Fuegt neue Vorlesung hinzu
    void AddLecture(std::string lecName);

    // Loescht vorhandene Vorlesung
    void RemoveLecture(std::string lecName);

    // Fuegt neuen Studenten hinzu
    void AddStudent(int matNr, std::string name);

    // Loescht vorhandenen Studenten
    void RemoveStudent(int matNr);

    // Fuegt neue 'hoert'-Beziehung hinzu
    bool AddStudentToLecture(int matNr, std::string lecName);

    // Gibt 'true' zurück gdw. Student mit 'matNr' Vorlesung 'lecName' hoert
    bool IsStudentInLecture(int matNr, std::string lecName);

private:
    // 1. Deklaration der Typ-Instanz 'IntList'
    // 2. Deklaration der Verbundtypen 'Student' und 'Lecture'
    // 3. Deklaration der Typ-Instanzen 'StudentList' und 'LectureList'
    // 4. Deklaration der Listen (Datenobjekte) 'students' und 'lectures'
};
```

(a) Vervollständigen Sie den privaten Teil der Schnittstelle. Es fehlen (1) die Deklaration für die Typ-Instanz `IntListe` der generischen Liste, (2) Deklarationen für die Verbundtypen `Student` und `Lecture`, (3) für die Typ-Instanzen `StudentList` und `LectureList` der generischen Liste sowie (4) für die Listen-Datenobjekte `students` und `lectures`. Verwenden Sie die Abbildung als Hilfe.



(b) Implementieren sie den Rumpf in `LecAdm.cc`. Achten Sie darauf, dass die Implementierung sich auf die im privaten Teil der Schnittstelle deklarierten Listen abstützt. Testen Sie Ihre Implementierung mit der Datei `LecAdmTest.cc`, die sie von der Webseite herunterladen können.

Abgabe: 13. 07. 2007 vor der Vorlesung..