

Übungsblatt 12

Abgabe: 05.02.2007

Ein Auftrag vom Prüfungsamt

Da die Universität Bremen in allen Bereichen schneller/ höher/ weiter/ exzellenter werden soll, wird in den Reformbestrebungen auch vor dem Prüfungsamt des FB3 nicht haltgemacht. Eine BUB (**B**ekannte **U**nternehmens**B**eratung – ohne geht es ja heutzutage nicht mehr) hat auch das Potential zur Effizienzsteigerung erkannt. Die vorgeschlagene Anschaffung weiterer SAP-Module war allerdings diesmal den Verantwortlichen deutlich zu teuer.

Alles bleibt gut

Damit alle bisherigen Abläufe weiterhin ungestört möglich sind, sollen die Daten der Studierenden wie bisher alphabetisch sortiert gespeichert werden, damit bei der jährlichen Archivierung alle Daten in der richtigen Reihenfolge zum Abheften zur Verfügung stehen. Die Studierendendaten bestehen dabei aus einem Namen in der Form *Nachname*, *Vorname*, der Matrikelnummer, und einer Referenz auf den eigentlichen Datensatz, der bestandene und nicht bestandene Prüfungen in Bremen, anerkannte Studienleistungen an anderen Orten, Einträge über Täuschungsversuche usw. enthält.

Hier geht es schnell

Da das Tagesgeschäft im Prüfungsamt doch vorwiegend im Bearbeiten der Daten von einzelnen Studierenden besteht, soll der Zugriff auf einzelne Datensätze beschleunigt werden. Hier soll der Empfehlung der BUB gefolgt werden und ein *String-Hashing* verwendet werden, mit dem in einem Array von Listen von Studentendaten die zu durchsuchende Liste schnell gefunden wird.

Die Aufgabe

Es soll sowohl eine Listenstruktur für die sortierte Ausgabe der Daten als auch ein Hash-Array von Listen implementiert werden. Dabei sollen die Studentendaten **nicht** doppelt im Speicher gehalten werden, sondern nur die *Referenzen* auf die Einträge sollen in den zwei erwähnten Ordnungen organisiert werden!

Dabei soll bei der Wahl der Datenstruktur zusätzlich berücksichtigt werden, dass das Löschen eines Eintrags möglichst effizient implementiert wird. Es wird angenommen, dass alle Namen eindeutig sind, dass also keine zwei Studierenden den gleichen Namen haben.

Insgesamt sollen die folgenden vier Methoden mit geeignet zu definierenden Signaturen implementiert werden:

immatrikulieren fügt einen neuen Studenten in die Hash-Struktur sowie in die alphabetisch sortierte Liste ein.

exmatrikulieren löscht einen Eintrag zu einem gegebenen Namen sowohl aus der Hash – Struktur als auch aus der Liste.

verwalten sucht zu einem gegebenen Namen über die Hash-Liste die dazugehö-

de Daten heraus und liefert diese als Rückgabewert.

`archivieren` gibt alle Namen und dazugehörigen Daten alphabetisch sortiert auf dem Bildschirm aus.

Zeigt mit geeigneten Testfällen, dass Eure Methoden die gewünschte Funktionalität besitzen und dem Prüfungsamt helfen.

Anmerkung: Wenn eine einfach verkettete Liste benötigt wird, so existiert mit `MyList.java` bereits eine gute – weil beweisbar korrekte – Implementierung.