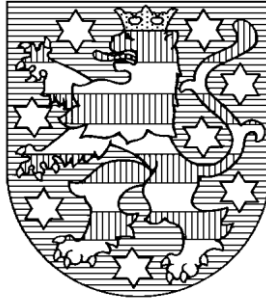


Thüringer Kultusministerium



Abiturprüfung 1999

Informatik

als Leistungsfach
(Haupttermin)

Arbeitszeit: 270 Minuten

Hilfsmittel: Formeln und Tabellen für die Sekundarstufen I und II/
Paetec, Gesellschaft für Bildung und Technik mbH, Berlin;
PC mit Prolog-System und
Turbo Pascal- oder Oberon-System;
Taschenrechner

Der Prüfungsteilnehmer löst **alle** Aufgaben.

Rechts unten neben jeder Teilaufgabe steht die für diese Teilaufgabe maximal erreichbare Anzahl von Bewertungseinheiten (BE).

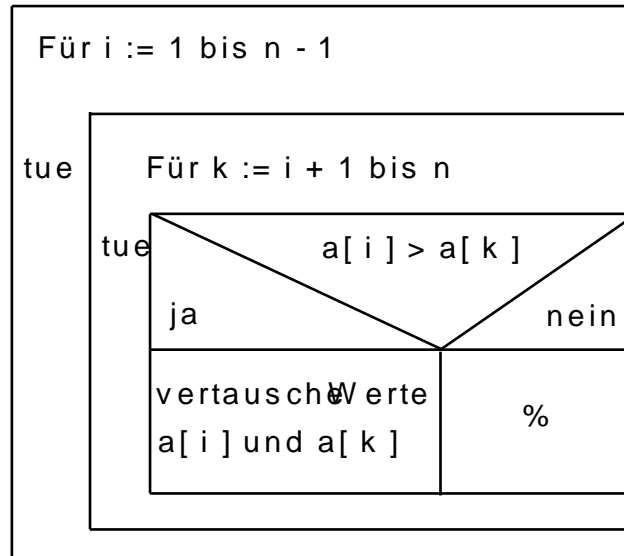
Der Prüfungsteilnehmer sichert bei der praktischen Arbeit am PC mindestens alle 10 Minuten die von ihm erarbeiteten Programme.

Der Prüfungsteilnehmer hat die von ihm erarbeiteten Programme und Module im Quelltext zu kommentieren.

Der Prüfungsteilnehmer hat die von ihm erarbeiteten Quelltexte abzugeben.

1 Zeitkomplexität von Algorithmen

- 1.1 Der folgende Algorithmus sortiert n gegebene Zahlen in aufsteigender Reihenfolge.



i und k sind INTEGER-Zahlen.
 a ist eine Reihung (Array) von n Zahlen.

Geben Sie die Anzahl der Vergleiche und die Anzahl der Vertauschungen für den folgenden Fall an: Die n Zahlen der Reihung a sind bereits in aufsteigender Reihenfolge gegeben.

Geben Sie die Anzahl der Vergleiche und die Anzahl der Vertauschungen für den folgenden Fall an: Die n Zahlen der Reihung a sind in absteigender Reihenfolge gegeben.

4 BE

- 1.2 Es sind 100.000 Zahlen gegeben, die sortiert werden sollen. Dafür stehen zwei Programme zur Verfügung. Zum einen ein Programm, das Sortieren durch direktes Einfügen realisiert. Zum anderen ein Programm, das Quicksort realisiert. Welches der beiden Programme sortiert die 100.000 Zahlen schneller?

Begründen Sie Ihre Antwort!

5 BE

- 1.3 Gegeben ist ein binärer Baum minimaler Höhe.

Welche Höhe hat der binäre Baum in Abhängigkeit von der Anzahl der Knoten?

Begründen Sie Ihre Antwort!

5 BE

2 Hash-Verfahren

In einem einzügigen Gymnasium gibt es in jeder Klassenstufe von 5 bis 12 genau eine Klasse.

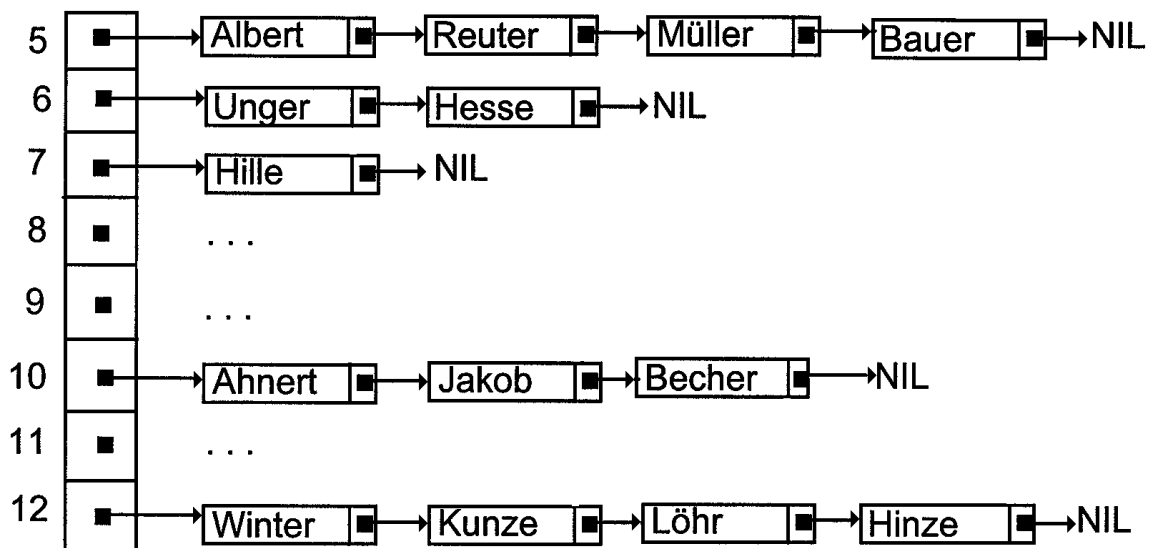
Die Klassenbücher enthalten die Nachnamen der Schülerinnen und Schüler der jeweiligen Klasse.

Die Klassenbücher werden in einem Regal aufbewahrt, das acht Fächer besitzt. Jedes Fach ist mit der Nummer der Klasse beschriftet, deren Klassenbuch in dem Fach aufbewahrt wird.

Die Registrierung von Nachnamen in Klassenbüchern und deren Aufbewahrung in Fächern lässt sich auf einem Computer simulieren. Dazu kann ein Hash-Verfahren eingesetzt werden.

Von Bedeutung bei dem Hash-Verfahren ist die sogenannte Hash-Tabelle. Die Hash-Tabelle ist eine eindimensionale Reihung (Array). Die Reihung besteht aus acht Elementen. Für jede Klasse von 5 bis 12 ist ein Element vorhanden. Jedes Element der Reihung enthält einen Verweis auf eine einfach verkettete Liste. In jeder Liste sind die Nachnamen der Schülerinnen und Schüler der jeweiligen Klasse verkettet.

Die folgende Abbildung stellt einen Ausschnitt aus einer möglichen Hash-Tabelle dar:



- 2.1 Entwerfen Sie ein Modul, das die Nachnamen der Schülerinnen und Schüler dieses Gymnasiums in einer Hash-Tabelle verwaltet!

Dieses Modul soll die folgenden Operationen realisieren:

- Anlegen der leeren Hash-Tabelle,
- Einfügen eines bestimmten Schülernamens einer bestimmten Klasse in die Hash-Tabelle,
- Löschen eines bestimmten Schülernamens einer bestimmten Klasse aus der Hash-Tabelle,
- Ermitteln, ob es einen bestimmten Schüler in einer bestimmten Klasse gibt,
- Ermitteln, in welcher Klasse ein bestimmter Schüler ist.

Implementieren Sie das Modul in **Turbo Pascal** oder **Oberon**!

Beachten Sie die folgenden Festlegungen:

- Sie können davon ausgehen, dass keine zwei Schüler den gleichen Nachnamen besitzen.
- Die Reihenfolge bei den Nachnamen spielt in den Listen keine Rolle.
- Das Modul hat den Namen Hash.

16 BE

- 2.2 Entwerfen Sie ein Programm, das die folgenden Handlungsfolgen realisiert:

1. Die leere Hash-Tabelle ist anzulegen.
2. Für die Klasse 5 sind die Namen Schulze, Müller, Bauer, Schneider, Albert und Reuter in die Hash-Tabelle einzufügen.
3. Für die Klasse 10 sind die Namen Neumann, Becher, Ahnert, Meier und Jakob in die Hash-Tabelle einzufügen.
4. Für die Klasse 12 sind die Namen Hinze, Kunze, Löhr und Winter einzufügen.
5. Es ist zu ermitteln, ob Bauer ein Schüler der Klasse 5 ist.
6. Die Namen Müller aus Klasse 5 und Hinze aus Klasse 12 sind zu löschen.
7. Es ist zu ermitteln, ob es einen Schüler Ritter an diesem Gymnasium gibt.
8. Für die Klasse 10 sind die Namen Heller und Fischer einzufügen.

9. Es ist zu ermitteln, in welcher Klasse der Schüler Becher ist.

Implementieren Sie das Programm in **Turbo Pascal** oder **Oberon**!

Beachten Sie die folgenden Festlegungen:

- Das Programm hat den Namen `Demo`.
- Das Programm importiert das Modul `Hash` aus Teilaufgabe 2.1.

12 BE

3 Parser

Gegeben ist die folgende Definition der Syntax von Bezeichner in dem erweiterten Backus-Naur-Formalismus (EBNF):

bezeichner = buchstabe { buchstabe | ziffer }.

buchstabe = "x" | "y".

ziffer = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9".

3.1 Geben Sie für die Definition dieser Syntax von Bezeichner an

- a) die Terminalsymbole,
- b) die Nichtterminalsymbole,
- c) das Startsymbol!

3 BE

3.2 Überführen Sie die gegebene Definition in Syntaxdiagramme!

5 BE

3.3 Entwerfen und implementieren Sie ein **Prolog-Programm** für die gegebene Syntax von Bezeichner! Das Programm soll erkennen, ob ein bei der Abfrage angegebener Bezeichner syntaktisch korrekt ist.

Beachten Sie die folgende Festlegung:

Der Bezeichner ist als Liste anzugeben.

Beispiel: ?- bezeichner([x, x, 1, y, x, y, 2, 3, 0]).

10 BE
