

FREISTAAT THÜRINGEN

Kultusministerium



# ABITURPRÜFUNG 2002

## GRUNDFACH INFORMATIK (HAUPTTERMIN)

Arbeitszeit: 210 Minuten

Hilfsmittel: Formeln und Tabellen für die Sekundarstufen I und II.  
Berlin: Paetec, Ges. für Bildung und Technik mbH;  
PC mit Oberon- oder Turbo Pascal-System;  
Realisation des ADT Liste; Taschenrechner

Der Prüfungsteilnehmer wählt von den Aufgaben 1, 2, 3 und 4 **drei** Aufgaben zur Bearbeitung aus.

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

Die Aufgabe 4 ist mit Hilfe des PC zu lösen. Der Prüfungsteilnehmer sichert bei der praktischen Arbeit am PC mindestens alle 10 Minuten den von ihm erarbeiteten Quelltext. Er hat das von ihm erarbeitete Programm im Quelltext zu kommentieren. Der Quelltext ist zusammen mit der Abiturarbeit abzugeben.

## ÖFFNUNG AM 26. APRIL 2002

### Aufgabe 1

- 1.1 Ein Spiel wird von zwei Personen gespielt. Der eine Teilnehmer hat die Spielsteine X, der andere die Spielsteine O. Die beiden Spieler setzen abwechselnd einen Spielstein auf ein Spielbrett, das aus drei mal drei Quadraten aufgebaut ist. Das Spielbrett ist zu Beginn des Spieles leer. Auf ein Quadrat darf nur ein Spielstein gesetzt werden. Der Spieler, der als erster eine Zeile oder Spalte oder Diagonale mit seinen Spielsteinen besetzt, gewinnt das Spiel. Das Spiel ist dann beendet.

In der folgenden Abbildung hat der Spieler mit den Spielsteinen X gewonnen:

X	O	O
O	X	
		X

Ein Computer soll das Setzen der Spielsteine durch die beiden Spieler während des gesamten Spieles verwalten. Der Computer soll absichern, dass keine zwei Spielsteine auf ein Quadrat gesetzt werden. Der Computer soll auch den Gewinner ermitteln.

Entwerfen Sie einen Algorithmus, der diese Aufgaben realisiert! Geben Sie eine Datenstruktur an, in der das Spielbrett gespeichert werden kann!

12 BE
-------

- 1.2 Gegeben ist eine Prozedur  $\text{Kreis}(x, y, r)$ , die einen Kreis mit dem Mittelpunkt  $M(x, y)$  und dem Radius  $r$  zeichnet. Die Parameter  $x, y$  und  $r$  sind ganze Zahlen.

Gesucht sind zwei Programme, die jeweils die gleiche Grafik auf den Bildschirm zeichnen. Die Grafik besteht aus konzentrischen Kreisen mit den Radien 1, 3, 5, ..., 99. Dem ersten Programm soll ein iterativer Algorithmus, dem zweiten ein rekursiver Algorithmus zu Grunde liegen.

Geben Sie die beiden Programme in Oberon oder Turbo Pascal an!

Vergleichen Sie Iteration und Rekursion! Beziehen Sie sich bei dem Vergleich auf die beiden Programme!

8 BE
------

**Aufgabe 2**

2.1 Entscheiden Sie, ob die folgende These gültig ist:

Jedes Problem, das sich präzise beschreiben lässt, kann mit einem Computer gelöst werden.

Begründen Sie Ihre Entscheidung!

4 BE
------

2.2 An Ihrer Schule soll auf einem Forum über die Rolle des Computers in der Arbeitswelt diskutiert werden. Sie sollen das Forum mit einem Kurzvortrag eröffnen. Geben Sie für den Kurzvortrag eine Gliederung mit geeigneten Schwerpunkten an!

8 BE
------

2.3 Ein Automat bohrt Löcher in Leiterplatten. Der Gesamtweg des Bohrkopfes bei der Bearbeitung einer Leiterplatte soll möglichst kurz sein.

Entwerfen Sie einen Algorithmus, der einen möglichst kurzen Gesamtweg des Bohrkopfes findet!

Es sollen Leiterplatten mit 2000 Löchern hergestellt werden. Ermitteln Sie, ob sich der von Ihnen entworfene Algorithmus auch dafür eignet!

8 BE
------

### Aufgabe 3

- 3.1 Eine Gruppe von Schülern will ihre gemeinsame Seminarfacharbeit an Computern anfertigen. Alle Computer sind in einem lokalen Computernetz (LAN) eingebunden.

Erläutern Sie Möglichkeiten, welche die Arbeit in einem LAN mit sich bringt!

Ein Schüler erhält von einem Mitschüler ein Blatt, auf dem die folgenden Angaben stehen:

c:\semfach\entwurf\projekt.html

<http://www.th.schule.de/kyf/bsp gym/projekt.html>

Was bedeuten diese Angaben?

Die Seminarfacharbeit soll im WWW veröffentlicht werden. Erläutern Sie, wie vorzugehen ist!

10 BE
-------

- 3.2 In einem HTML-Dokument sollen die Umlaute ä, ö, ü und das Zeichen ß maskiert werden. In der folgenden Tabelle ist die jeweilige Maskierung angegeben:

Zeichen	Maskierung
ä	&auml;
ö	&ouml;
ü	&uuml;
ß	&szlig;

Ein Algorithmus leistet Folgendes:

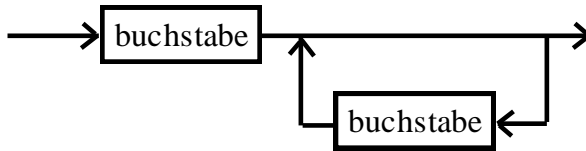
- Er liest eine Folge von Zeichen ein.
- Er ersetzt die enthaltenen Umlaute und das Zeichen ß nach den Festlegungen der Tabelle durch die Maskierung und lässt alle anderen Zeichen unverändert.
- Er gibt die erhaltene Folge von Zeichen aus.

Erläutern Sie, ob dieser Algorithmus in höheren Programmiersprachen implementiert werden kann!

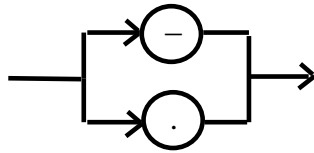
5 BE
------

- 3.3 Gegeben ist die folgende Definition der Syntax von E-Mail-Adressen. Die Definition erfolgt mit Syntaxdiagrammen und mit EBNF. Das Nichtterminalsymbol `buchstabe` steht für einen Kleinbuchstaben. Das Symbol `adresse` ist das Startsymbol.

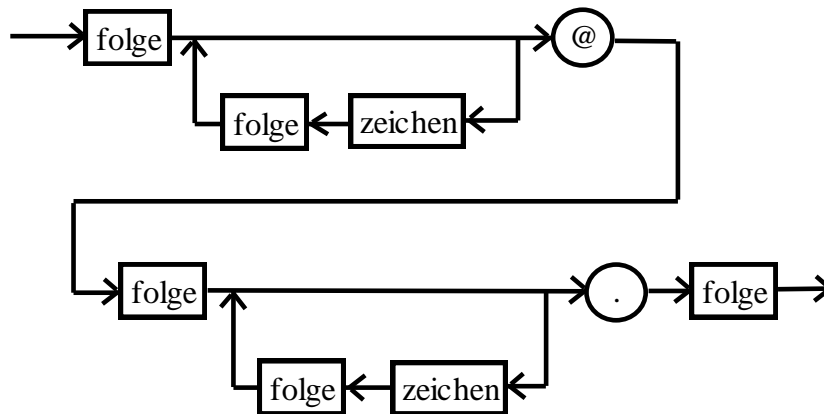
`folge`



`zeichen`



`adresse`



`folge` = `buchstabe` { `buchstabe` } .

`zeichen` = `"-"` | `"."` .

`adresse` = `folge` { `zeichen` `folge` } `"@"` `folge` { `zeichen` `folge` } `"."` `folge` .

Geben Sie eine syntaktisch korrekte E-Mail-Adresse an!

Entscheiden Sie, ob die folgenden zwei E-Mail-Adressen syntaktisch korrekt sind!

`hans.meier@software-gmbh.com`  
`ausstellung-a-@wartburg.de`

Begründen Sie Ihre Entscheidungen!

5 BE

### Aufgabe 4

Ausdrücke lassen sich in Infix-Notation und in Postfix-Notation angeben. Bei der Postfix-Notation steht ein Operator hinter seinen beiden Operanden. Klammern und Vorrangregeln werden nicht benötigt. Die folgende Tabelle enthält vier jeweils gleichwertige Ausdrücke in Infix- und Postfix-Notation:

Infix-Notation	Postfix-Notation
3-4	34-
(5+3)*2	53+2*
5+3*2	532*+
(5+3)*(6-9)	53+69-*

Zur Ermittlung des Wertes eines Ausdrucks in Postfix-Notation wird ein Stapel verwendet. Das Vorgehen wird an dem Ausdruck  $53+69-*$  erläutert.

Zuerst wird ein leerer Stapel angelegt. Dann wird der Ausdruck von links nach rechts Zeichen für Zeichen bearbeitet. Es werden die Zahlen 5 und 3 gestapelt. Der Operator + führt dazu, dass die Zahlen 3 und 5 entstapelt und dann addiert werden. Die Summe 8 wird gestapelt. Nun werden die Zahlen 6 und 9 gestapelt. Der Operator - führt dazu, dass die Zahlen 9 und 6 entstapelt und dann von einander subtrahiert werden. Die Differenz -3 wird gestapelt. Der Operator \* führt dazu, dass die Zahlen -3 und 8 entstapelt und dann multipliziert werden. Das Produkt -24 wird gestapelt.

Da der Ausdruck nun vollständig abgearbeitet ist, wird die Zahl -24 entstapelt. Der Stapel ist nun leer. Das Ergebnis -24 wird ausgegeben.

Entwerfen und implementieren Sie ein Programm in Oberon oder Turbo Pascal, das den Wert eines eingelesenen Ausdrucks in Postfix-Notation ermittelt und ausgibt!

Testen Sie das Programm an den vier Beispielen der Tabelle! Die Tests sind zu dokumentieren!

Beachten Sie die folgenden Festlegungen:

- Zulässige Operatoren sind +, - und \*. Als Operanden können 0, 1, 2, 3, 4, 5, 6, 7, 8 oder 9 angegeben werden.
- Vom Nutzer des Programms wird ein korrekter Ausdruck in Postfix-Notation eingegeben.
- Das Programm hat den ADT Liste zu importieren und zu verwenden. Die Spezifikation des ADT Liste ist im Anhang gegeben.

**Anhang:**  
**Spezifikation des abstrakten Datentyps "Liste" (ADT Liste)**

**(A) Wertebereich:**

Eine (einfach verkettete) Liste besteht aus Elementen, die in linearer Folge angeordnet sind. Eine solche Liste kann nur von vorn nach hinten durchlaufen werden.

Ist die Liste nicht leer, dann gibt es das **erste Element**, das **letzte Element** und ein **aktuelles Element**. Enthält eine Liste genau ein Element, dann ist dieses Element sowohl aktuelles, erstes als auch letztes Element. Für jedes Element einer Liste, mit Ausnahme des letzten Elementes, gibt es das **nächste Element**.

**(B) Operationen:**

**1. Operation: Erzeugen**

Eine leere Liste wird angelegt.

Diese Operation ist Voraussetzung für alle anderen Operationen.

**2. Operation: EinfuegenElement**

Ein Element wird vor dem aktuellen Element eingefügt.

Wird diese Operation auf eine leere Liste angewandt, dann enthält die Liste anschließend genau ein Element.

Das neue Element wird zum aktuellen Element.

**3. Operation: AnhaengenElement**

Ein Element wird nach dem letzten Element angefügt.

Wird diese Operation auf eine leere Liste angewandt, dann enthält die Liste anschließend genau ein Element.

Das neue Element wird zum aktuellen Element.

**4. Operation: LoeschenElement**

Das aktuelle Element wird gelöscht.

Das erste Element wird dann zum aktuellen Element.

Ist die Liste leer, dann passiert nichts.

Enthält die Liste genau ein Element, dann wird dieses gelöscht; dadurch entsteht eine leere Liste.

**5. Operation: GeheErstes**

Das erste Element wird zum aktuellen Element.

Ist die Liste leer, dann passiert nichts.

**6. Operation: GeheLetztes**

Das letzte Element wird zum aktuellen Element.  
Ist die Liste leer, dann passiert nichts.

**7. Operation: GeheNaechstes**

Das nächste Element wird zum aktuellen Element.  
Diese Operation darf nicht angewandt werden, wenn das letzte Element  
aktuelles Element ist.  
Ist die Liste leer, dann passiert nichts.

**8. Operation: HoleEintrag**

Der Inhalt des aktuellen Elementes wird bereit gestellt.  
Das aktuelle Element bleibt aktuelles Element.  
Diese Operation darf nicht angewandt werden, wenn die Liste leer ist.

**9. Operation: SchreibeEintrag**

Der Inhalt des aktuellen Elementes wird überschrieben.  
Das aktuelle Element bleibt aktuelles Element.  
Diese Operation darf nicht angewandt werden, wenn die Liste leer ist.

**10. Operation: ListeLeer**

Diese Operation liefert den Wert TRUE, wenn die Liste leer ist.  
Ansonsten liefert sie den Wert FALSE.

**11. Operation: ListenEnde**

Diese Operation liefert den Wert TRUE, wenn das letzte Element aktuel-  
les Element ist.  
TRUE wird auch geliefert, wenn die Liste leer ist.  
Ansonsten liefert die Operation den Wert FALSE.