



# ABITURPRÜFUNG 2004

## LEISTUNGSFACH

### INFORMATIK (HAUPTTERMIN)

Arbeitszeit: 270 Minuten

Hilfsmittel: Formeln und Tabellen für die Sekundarstufen I und II,  
Berlin: Paetec, Ges. für Bildung und Technik mbH  
oder  
Das große Tafelwerk, Volk und Wissen Verlag GmbH,  
Berlin  
PC mit Prolog-System und Oberon- oder Turbo Pascal-  
System  
Realisation des ADT Liste, Zufallszahlengenerator,  
Taschenrechner

Der Prüfungsteilnehmer löst die Aufgaben 1 und 2 und wählt von den  
Aufgaben 3.1 und 3.2 eine Aufgabe zur Bearbeitung aus.

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 Quelltexte. Er hat die von ihm  
erarbeiteten Programme und Module im Quelltext zu kommentieren.  
Die Quelltexte sind zusammen mit der Abiturarbeit abzugeben.

**ÖFFNUNG AM 12. MAI 2004**

**Aufgabe 1**

- a) Gegeben ist die Anweisungsfolge:

`r := 0.93;`

`REPEAT`

`r := r + 0.01`

`UNTIL r = 1.00;`

Ein Programm in Oberon oder Turbo Pascal enthält diese Anweisungsfolge.

Begründen Sie, warum es keine Garantie dafür gibt, dass das Programm terminiert.

3 BE

- b) Der Zuverlässigkeit von Informatiksystemen kommt in der Praxis große Bedeutung zu.

Erläutern Sie, durch welche Maßnahmen ein hohes Maß an Zuverlässigkeit eines Informatiksystems gesichert werden kann.

4 BE

- c) Ist es im Allgemeinen möglich, dass ein Computerprogramm den optimalen Stunden-, Raum-, Lehrereinsatz- und Pausenaufsichtsplan für eine Schule in akzeptabler Zeit erstellt? Begründen Sie Ihre Antwort.

5 BE

- d) Erläutern Sie, welche Auswirkungen der Computereinsatz in einem konkreten Bereich der Arbeitswelt hat.

3 BE

## Aufgabe 2

Die Sprache L ist durch folgende Syntax definiert:

satz = subjekt praedikat objekt.  
 subjekt = "schoolgirls" | "schoolboys".  
 praedikat = [hilfsverb] vollverb.  
 hilfsverb = "can".  
 vollverb = "use" | "handle".  
 objekt = "computers" | "calculators".

Das Startsymbol ist satz.

- a) Erläutern Sie die Begriffe Terminalsymbol und Nicht-terminalsymbol. 2 BE
- b) Überführen Sie die Definition der Syntax der Sprache L in Syntaxdiagramme. 2 BE
- c) Geben Sie die Anzahl der Sätze der Sprache L an. 1 BE
- d) Entwerfen und implementieren Sie ein Programm in Prolog, das erkennt, ob ein bei der Anfrage angegebener Satz syntaktisch korrekt ist.  
 Ein Satz ist als Liste anzugeben.  
 Beispiel: ?- satz([schoolgirls,can,use,computers]). 6 BE
- e) Entwerfen und implementieren Sie ein weiteres Programm in Prolog, das bei Anfrage alle Sätze der Sprache L ausgibt.  
 Geben Sie zu dem Programm eine derartige Anfrage an. 4 BE

**Aufgabe 3.1**

- a) Erläutern Sie das FIFO-Prinzip, nach dem eine Schlange arbeitet.

2 BE
------

- b) Geben Sie eine Spezifikation für den ADT Schlange an.

7 BE
------

- c) Realisieren Sie den ADT Schlange in Oberon oder Turbo Pascal.

Als Realisation ist ein Modul adtschlange anzugeben.

Für die Realisation ist die Spezifikation von Teilaufgabe b) maßgebend. Das vorgegebene Modul adtliste ist zu importieren und zu verwenden.

9 BE
------

- d) Entwerfen Sie eine Funktion, die mit Hilfe eines Zufallszahlengenerators eine Zufallszahl aus der Menge  $\{ 1, 2, 3, \dots, n \}$  ermittelt.

Implementieren Sie die Funktion in Oberon oder Turbo Pascal und testen Sie die Funktion.

Dokumentieren Sie die Tests.

2 BE
------

- e) Für den Flug EF 105 wurden 80 Tickets verkauft. Am Abflugtag checken die Passagiere ab 11:00 Uhr ein. Die Passagiere, die zum Einchecken eintreffen, erhalten eine fortlaufende Nummer von 1 bis 80. Sie stellen sich am Schalter 1, 2 oder 3 an.

Entwerfen und implementieren Sie ein Programm in Oberon oder Turbo Pascal, das die Abfertigung der Passagiere des Flugs EF 105 simuliert.

Das Programm soll für jede Minute ab 11:00 Uhr ausgeben, welche Passagiere an welchem Schalter anstehen.

Das Programm soll zusätzlich für jeden der 80 Passagiere ausgeben, wann er zum Einchecken eintrifft, an welchem Schalter er sich anstellt und wann er eingeecheckt ist.

Das Programm soll das Modul adtschlange von Teilaufgabe c) importieren und nutzen sowie zusätzlich die Funktion von Teilaufgabe d) verwenden.

Dokumentieren Sie einen Testlauf des Programms.

Die Simulation soll auf Grundlage der folgenden vereinfachenden Annahmen erfolgen:

- Ab 11:00 Uhr trifft in jeder Minute ein Passagier mit einer Wahrscheinlichkeit von 75% zum Einchecken ein.
- In einer Minute kann höchstens ein Passagier eintreffen.
- Die eintreffenden Passagiere stellen sich mit gleicher Wahrscheinlichkeit an einem der drei Schalter 1, 2 oder 3 an. Die Länge der Schlangen an den drei Schaltern spielt für die Wahl des Schalters keine Rolle.
- Das Abfertigen eines Passagiers dauert 2 Minuten mit einer Wahrscheinlichkeit von 20%, 3 min mit einer Wahrscheinlichkeit von 60% und 4 min mit einer Wahrscheinlichkeit von 20%.
- Nach dem Einchecken verlässt ein Passagier sofort den Schalter und der nächste wird abgefertigt.
- Kein Passagier wechselt beim Warten die Schlange, in der er sich angestellt hat.
- Alle 80 Passagiere checken ein.

10 BE
-------

### Aufgabe 3.2

- a) Im Jahr 2004 jährt sich zum 50. Mal der Todestag von Alan Turing. Aus diesem Grund sollen Sie in einem Vortrag Leben und Wirken des Wissenschaftlers Alan Turing darstellen. Geben Sie für Ihren Vortrag Gliederung und inhaltliche Schwerpunkte an.

3 BE
------

- b) Beschreiben Sie den Aufbau und erläutern Sie die Arbeitsweise einer Turing-Maschine.

5 BE
------

- c) Eine Turing-Maschine  $T_1$  soll die Summe zweier positiver ganzer Zahlen  $a$  und  $b$  berechnen. Das Eingabewort besteht aus zwei Folgen von Einsen und einem Stern zwischen beiden Folgen. Die erste Folge besteht aus  $a$  Einsen und stellt die Zahl  $a$  dar. Die zweite Folge besteht aus  $b$  Einsen und stellt die Zahl  $b$  dar. Ansonsten befinden sich nur Sterne auf dem Band. Das Ausgabewort von  $T_1$  besteht aus  $a + b$  Einsen und stellt die Summe  $a + b$  dar.

Beispiel:

Eingabewort	Ausgabewort
* 1 1 1 1 * 1 1 1 *	* 1 1 1 1 1 1 1 *

Konstruieren Sie eine Turing-Maschine  $T_1$ , welche die Summe  $a + b$  für beliebige positive ganze Zahlen  $a$  und  $b$  berechnet. Erstellen Sie ein Ablaufprotokoll der Turing-Maschine  $T_1$  für  $a = 4$  und  $b = 3$ .

7 BE
------

- d) Eine Turing-Maschine  $T_2$  soll eine positive ganze Zahl  $a$  verdoppeln. Das Eingabewort besteht aus einer Folge von  $a$  Einsen und stellt die Zahl  $a$  dar. Ansonsten befinden sich nur Sterne auf dem Band. Das Ausgabewort von  $T_2$  besteht aus  $2 \cdot a$  Einsen.

Beispiel:

Eingabewort	Ausgabewort
* 1 1 1 *	* 1 1 1 1 1 1 *

Konstruieren Sie eine Turing-Maschine  $T_2$ , welche eine beliebige positive ganze Zahl  $a$  verdoppelt. Erstellen Sie ein Ablaufprotokoll der Turing-Maschine  $T_2$  für  $a = 3$ .

5 BE

- e) Entwerfen Sie ein Programm zur Simulation der Turing-Maschine  $T_1$  aus Teilaufgabe c) oder der Turing-Maschine  $T_2$  aus Teilaufgabe d).

Implementieren Sie das Programm in Oberon oder Turbo Pascal. Beschreiben Sie die Handlungsfolge, die vom Benutzer des Programms auszuführen ist, damit die Abarbeitung der von Ihnen gewählten Turing-Maschine durch das Programm simuliert wird.

10 BE