

ABITURPRÜFUNG 2003

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;
Zufallszahlengenerator; 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 09. MAI 2003

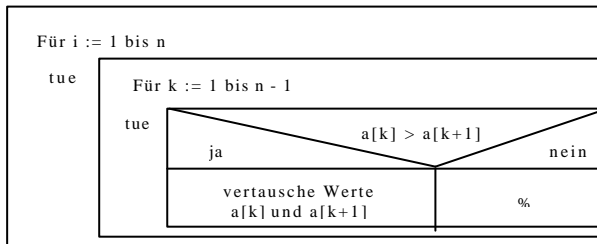
Aufgabe 1

- 1.1 In einer Reihung sind n ganze Zahlen gespeichert. Die Zahlen sollen sortiert werden. Beschreiben Sie am Beispiel der folgenden Reihung, wie dies mit Hilfe des Algorithmus „Sortieren durch Auswählen“ abläuft!

a_1	a_2	a_3	a_4	a_5	a_6
10	17	-36	48	0	20

4 BE

- 1.2 Der folgende Algorithmus sortiert n ganze Zahlen in aufsteigender Reihenfolge:



a ist eine Reihung von n ganzen Zahlen.
 i und k sind ganze Zahlen.

Geben Sie jeweils die Anzahl der Vergleiche und die Anzahl der Vertauschungen für die folgenden Fälle an:

1. Fall: Die n ganzen Zahlen der Reihung a sind bereits in aufsteigender Reihenfolge gegeben.
2. Fall: Die n ganzen Zahlen der Reihung a sind in absteigender Reihenfolge gegeben.

Geben Sie eine Möglichkeit zum Verbessern des Zeitverhaltens des gegebenen Algorithmus an!

8 BE

1.3 Gegeben ist der Quelltext der Funktion f:

Oberon:

```
PROCEDURE f(x: INTEGER): INTEGER;  
BEGIN  
  IF x = 0 THEN RETURN 0  
  ELSIF x > 0 THEN RETURN f(x - 3)  
  ELSE RETURN f(x + 3)  
  END  
END f;
```

Turbo Pascal:

```
FUNCTION f(x: INTEGER): INTEGER;  
BEGIN  
  IF x = 0 THEN f := 0  
  ELSE  
    IF x > 0 THEN f := f(x - 3)  
    ELSE f := f(x + 3)  
  END;  
END;
```

Begründen Sie, dass es sich bei der Funktion f um eine rekursive Funktion handelt!

Berechnen Sie die Funktionswerte f(3), f(9), f(8), f(-12) und f(-7) !

Nennen Sie Eigenschaften eines Algorithmus!

Überprüfen Sie, ob diese Eigenschaften von der Funktion f erfüllt werden!

8 BE

Aufgabe 2

- 2.1 Die Allgegenwart von Computern in der Gesellschaft vermittelt den Eindruck, als sei alles mit Computern berechenbar.

Beurteilen Sie diese These!

4 BE

- 2.2 Würdigen Sie die Leistungen einer Persönlichkeit, die als Wegbereiter der Informatik bekannt ist!

4 BE

- 2.3 Vor rund fünfzig Jahren erarbeitete John von Neumann ein später nach ihm benanntes Rechnermodell. Dieses Modell beschreibt auch heute noch die grundlegende Arbeitsweise eines PC.

Erläutern Sie das von-Neumann-Rechnermodell!

6 BE

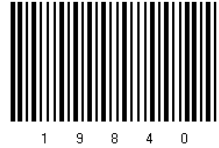
- 2.4 Erläutern Sie Auswirkungen des Einsatzes von Computern in der Arbeitswelt der Gegenwart!

6 BE

Aufgabe 3

Der Barcode 2/5 Industrial wird zur Identifizierung von Objekten verwendet. Mit dem Barcode lassen sich beliebige nichtnegative ganze Zahlen codieren.

Nachfolgend wird erläutert, wie die Zahl 1984 codiert wird (siehe nebenstehende Abbildung). Die Ziffer 0 ist eine Prüfziffer.



1. Schritt:

Die Zahl wird ziffernweise in eine Folge von Nullen und Einsen überführt. Dabei wird die folgende Tabelle benutzt:

Ziffer	1. Stelle	2. Stelle	3. Stelle	4. Stelle	5. Stelle
0	0	0	1	1	0
1	1	0	0	0	1
2	0	1	0	0	1
3	1	1	0	0	0
4	0	0	1	0	1
5	1	0	1	0	0
6	0	1	1	0	0
7	0	0	0	1	1
8	1	0	0	1	0
9	0	1	0	1	0

Die Zahl 1984 wird in die Folge 10001010101001000101 überführt.

2. Schritt:

Zur Berechnung der Prüfziffer werden die Ziffern der gegebenen Zahl von links nach rechts abwechselnd mit dem Faktor 3 oder mit dem Faktor 1 multipliziert. Begonnen wird links mit dem Faktor 3. Die entstandenen Produkte werden addiert. Die Prüfziffer ist die kleinste nichtnegative ganze Zahl, die zu dieser Summe zu addieren ist, um ein Vielfaches von 10 zu erhalten.

Für 1984 ergibt sich die Summe $1*3 + 9*1 + 8*3 + 4*1 = 40$ und daher die Prüfwert 0. Die Prüfwert wird mit Hilfe der Tabelle in eine Folge von Nullen und Einsen überführt und an die bisherige Folge angehängt. Für 1984 erhält man die Folge 1000101010100100010100110.

3. Schritt:





Am Anfang und am Ende der Folge werden ein Start- und ein Stoppzeichen ergänzt. Das Startzeichen wird durch 110, das Stoppzeichen durch 101 dargestellt. Für 1984 ergibt sich nun die vollständige Folge 1101000101010100100010100110101. Bei der Ausgabe des Barcodes wird für jede 1 ein breiter und für jede 0 ein schmaler schwarzer Strich gedruckt.

In den folgenden Aufgaben wird unter dem Barcode stets die vollständige Folge von Nullen und Einsen verstanden.

- a) Erzeugen Sie den Barcode für die Zahl 286! 3 BE
- b) Die Funktion $ziffer(z)$ soll die Prüfwert für die Zahl z ermitteln. Geben Sie den Quelltext der Funktion in Oberon oder Turbo Pascal an! 6 BE
- c) Geben Sie eine Datenstruktur an, die sich zum Speichern eines Barcodes eignet! Begründen Sie Ihre Antwort! 3 BE
- d) Entwerfen Sie einen Algorithmus, der einen Barcode einliest und die durch ihn codierte Zahl ermittelt und ausgibt! 4 BE
- e) Beim Einlesen eines Barcodes können Eingabefehler auftreten. Erläutern Sie, wie beim Barcode 2/5 Industrial Eingabefehler erkannt werden können! Gehen Sie in Ihrer Erläuterung auf die Codierung der einzelnen Ziffern (siehe Tabelle) sowie auf die Verwendung einer Prüfwert, eines Start- und eines Stoppzeichens ein! 4 BE

Aufgabe 4

An einem Spiel nehmen vier Personen teil. Wir nennen diese Personen A, B, C und D. Für das Spiel werden ein Spielbrett, eine Münze und für jeden Spieler ein Spielstein benötigt. Das Spielbrett ist eine lineare Anordnung von 60 Feldern. Die Felder tragen die Nummern von 1 bis 60. Feld Nr. 1 ist das Startfeld, Feld Nr. 60 das Zielfeld. Die Münze hat eine Vorder- und eine Rückseite.

Startfeld			 				Zielfeld
1	2	3	4	5	...	59	60

Zu Beginn des Spiels stehen alle Spielsteine auf dem Startfeld. In jeder Spielrunde wird die Münze von den Spielern in der Reihenfolge A, B, C und D geworfen. Liegt die Vorderseite der Münze oben, so wird der Spielstein des jeweiligen Spielers um zwei Felder in Richtung Zielfeld gerückt. Liegt die Rückseite oben, wird der Spielstein um ein Feld in Richtung Startfeld gerückt. Der Spielstein wird nicht gerückt, wenn er sich auf dem Startfeld befindet und die Rückseite der Münze oben liegt oder wenn er sich auf dem Feld Nr. 59 befindet und die Vorderseite oben liegt. Das Spiel wird beendet, wenn in einer Spielrunde mindestens ein Spieler das Zielfeld erreicht. Gewinner des Spiels sind alle Spieler, deren Spielsteine am Ende dieser Spielrunde auf dem Zielfeld stehen.

Entwerfen und implementieren Sie ein Oberon- oder Turbo Pascal-Programm, das dieses Spiel für vier Personen simuliert! Das Rücken der Spielsteine soll während des gesamten Spiels auf dem Monitor dargestellt werden. Der Zufallszahlengenerator des Oberon- oder Turbo Pascal-Systems ist zur Simulation des Werfens der Münze zu verwenden.

20 BE
