

(Leicht aktualisierte Version: 15.3.2011)

Beispielabitur

INFORMATIK

Arbeitszeit: 180 Minuten

Der Fachausschuss wählt je eine Aufgabe aus den Gebieten
Inf1 und Inf2 zur Bearbeitung aus.

Der Fachausschuss ergänzt in folgendem Feld die erlaubten objektorientierten
Programmiersprachen:

Inf1. MODELLIERUNG UND PROGRAMMIERUNG

BE

I.

Für die neu gegründete Bank YoungMoney wird eine Software zur Verwaltung von Konten entwickelt; die Software soll folgende Vorgänge in der Bank unterstützen:

Eine Person will ein Girokonto eröffnen. Der Bankangestellte fragt diese Person, ob sie bereits Kunde der Bank ist. Wenn nicht, wird sie als Kunde „eingrichtet“; dazu werden ihr Name und ihre Adresse (Straße, Hausnummer, Postleitzahl, Ort) erfasst. Der Bankangestellte legt die Kontonummer (ganze Zahl) und den individuellen Habenzinssatz fest und schreibt die verpflichtende Ersteinzahlung von mindestens 10 € gut.

Der Kontoinhaber zahlt künftig am Schalter bei einem Bankangestellten Geld auf das Konto ein bzw. hebt Geld davon ab und informiert sich bei diesem über den Kontostand. Ein Konto darf zunächst bis zu 500 € überzogen werden; der Sollzinssatz beträgt bei einer Kontoeröffnung grundsätzlich 11 % pro Jahr. Die Sollzinsen werden monatlich, die Habenzinsen dagegen jährlich gebucht.

Ein Bankangestellter ist dafür zuständig, die Zinsbuchungen termingerecht mithilfe der Software auszulösen. Für die Zinsberechnung müssen für jede Transaktion, d. h. für jede Kontobewegung, Datum und Betrag gespeichert werden.

- 10 1. Entwickeln Sie – ausgehend von einer Klasse KONTO – ein Klassendiagramm für eine Software, die es den Bankangestellten ermöglicht, die oben genannten Vorgänge durchzuführen.
2. Nun soll der Kontoeröffnungsvorgang für eine Person betrachtet werden, die noch nicht Kunde der Bank ist.
- 5 a) Bilden Sie diesen Vorgang in einem Sequenzdiagramm ab.
- 6 b) Geben Sie ein erweitertes Klassendiagramm für die Klasse KONTO an.
- 11 c) Implementieren Sie einen Konstruktor der Klasse KONTO, in dem möglichst viele Attribute bereits mit den Eröffnungsdaten belegt werden. Geben Sie ggf. zusätzliche Maßnahmen bzw. Rahmenbedingungen an, die nicht direkt aus dem Konstruktor ersichtlich sind, jedoch im Zusammenhang mit der Erzeugung eines KONTO-Objekts von Bedeutung sind.

[Hinweis: Sie können eine Klasse DATUM verwenden. Der Konstruktor DATUM() erzeugt ein Objekt, das (u. a.) mit dem aktuellen Datum initialisiert wird.]

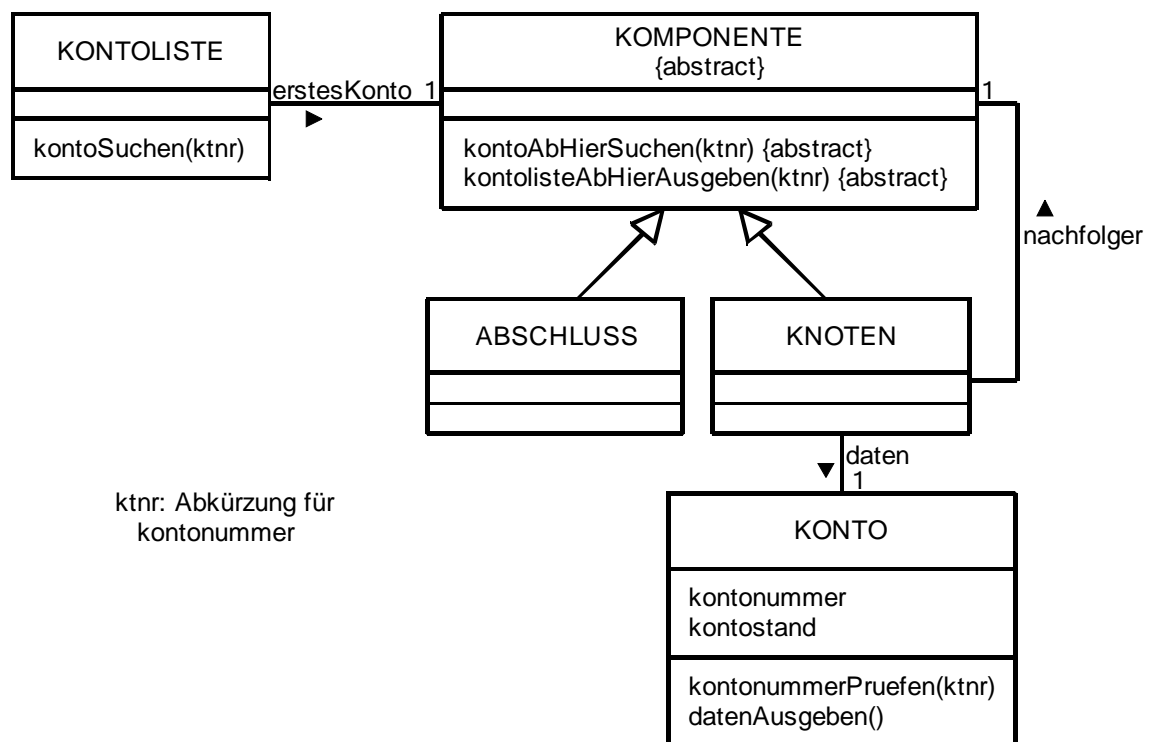
(Fortsetzung nächste Seite)

Inzwischen hat die Bankleitung entschieden, dass statt des bisher geplanten Einheitskontos nun verschiedene Arten von Konten angeboten werden sollen, nämlich Schülerkonten und Jobkonten. Habenzinsen werden nur noch bei Schülerkonten gewährt. Außerdem dürfen diese Konten maximal bis 50 € überzogen werden, der Sollzins beträgt dafür derzeit 6 % pro Jahr. Ein Jobkonto, auf dem das monatliche Gehalt eingeht, darf bis zum Dreifachen des monatlichen Einkommens überzogen werden; dafür ist aber ein Sollzins von derzeit 14 % pro Jahr zu entrichten.

- 5 3. Passen Sie Ihr Modell aus Aufgabe 1 durch eine geeignete Klassenhierarchie an die neue Situation an.

[Hinweis: Es müssen in Ihrem Modell nur die Klassen angegeben werden, die von dieser Anpassung unmittelbar betroffen sind.]

Gerade bearbeitete Konten werden im Arbeitsspeicher über eine nach der Kontonummer geordnete, einfach verkettete Liste verwaltet. Die Implementierung soll auf dem nachfolgenden Klassendiagramm basieren, bei dem nur die für die folgenden Aufgaben wichtigen Attribute und Methoden angegeben sind. Außerdem soll davon ausgegangen werden, dass die in Aufgabe 3 geforderte Klassenhierarchie bereits umgesetzt ist.



(Fortsetzung nächste Seite)

- 5 4. In der Kontoliste werden das Jobkonto (Kontonummer: 5; Kontostand: 340,52 €) sowie ein Schülerkonto (Kontonummer: 15; Kontostand: 123,50 €) verwaltet. Veranschaulichen Sie die entsprechende einfach verkettete Liste durch ein Objektdiagramm. Zur Vereinfachung können Sie sich bei Ihrer Darstellung auf die angegebenen KONTO-Attribute *kontonummer* und *kontostand* beschränken.
- 6 5. Ein weiteres Schülerkonto mit der Kontonummer 12 und einem Ersteinzahlungsbetrag von 10 € soll neu angelegt und in die nach der Kontonummer sortierte Liste aus Aufgabe 5 eingefügt werden. Beschreiben Sie mit eigenen Worten, welche Schritte – unter Beachtung der Reihenfolge – notwendig sind, um das Konto an der richtigen Stelle der Liste einzufügen.
- 9 6. Mithilfe der Methode *kontoSuchen(ktnr)* der Klasse KONTOLISTE kann die Liste nach einem Konto mit einer bestimmten Kontonummer durchsucht werden. Zurückgegeben wird die Referenz auf das gefundene Konto oder *null*, also einen Verweis, der sich auf kein Objekt bezieht.
- Implementieren Sie die dazu notwendigen Methoden in den Klassen KONTOLISTE, KNOTEN und ABSCHLUSS. Die Klasse KONTO stellt dazu eine Methode *kontonummerPruefen(ktnr)* bereit, die den Wahrheitswert *wahr* liefert, falls die übergebene Kontonummer mit der Kontonummer des ausführenden KONTO-Objekts übereinstimmt. Ansonsten wird der Wahrheitswert *falsch* zurückgegeben.
7. Nun werden zwei Operationen auf das Konto mit der Nummer 5 zeitgleich angestoßen:
- Operation 1: Abbuchen von 7,50 €
 - Operation 2: Überweisen von 100 € von Konto 5 auf Konto 15
- Bei solchen Vorgängen muss die Kontosoftware zuerst den derzeitigen Kontostand einlesen, dann die Berechnung durchführen und anschließend den neuen Kontostand abspeichern.
- 4 a) Zeigen Sie anhand eines möglichen Szenarios, dass die zeitgleiche Bearbeitung dieser beiden Operationen zu einem fehlerhaften Kontostand führen kann.
- 2 b) Geben Sie eine Strategie an, mit der man Probleme wie in Teilaufgabe 7a vermeiden kann.

(Fortsetzung nächste Seite)

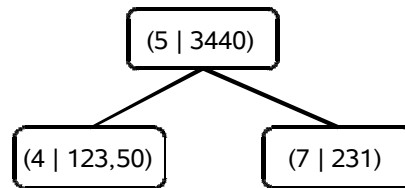
8. Eine andere Bank möchte nun die Software übernehmen. Für diese Bank sollen die Konten aber in einem nach der Kontonummer geordneten Binärbaum verwaltet werden.

2

a) Welchen Vorteil hat ein geordneter Binärbaum in der Regel gegenüber einer Listenstruktur?

4

b) Erweitern Sie den nachfolgend angegebenen, nach der Kontonummer geordneten Binärbaum, indem Sie nacheinander die Datensätze (3 | 24), (34 | 2000), (6 | 413,02) und (2 | -19,99), die in der Form (Kontonummer | Kontostand in €) gegeben sind, einfügen.



4

c) Zur Implementierung des Binärbaumes wird das Composite-Muster verwendet. Dieses entspricht im Wesentlichen dem Softwaremuster, das im Klassendiagramm vor Aufgabe 5 verwendet wird, wobei die Kardinalität 1 in der Beziehung *nachfolger* durch die Kardinalität 2 ersetzt wird.

Geben Sie die nötigen Attribute (inklusive des jeweiligen Datentyps) einer Implementierung der Klasse KNOTEN an.

7

d) Der Bankdirektor benötigt eine nach der Kontonummer aufsteigend sortierte Kontenliste. Dabei sollen alle abgespeicherten Daten ausgegeben werden. Wählen Sie einen dazu geeigneten Baumdurchlauf und geben Sie eine mögliche Implementierung der Methode *kontolisteAusgeben()* in den Klassen ABSCHLUSS und KNOTEN an. Sie können dabei eine Methode *datenAusgeben()* der Klasse KONTO verwenden, die alle Daten eines Kontos auf dem Bildschirm ausgibt.

Inf1. MODELLIERUNG UND PROGRAMMIERUNG

BE

II.

1. Das Versandhaus „Chic&Gut“ für Bekleidung unterhält einen Bestell-service per Internet. Jeder Benutzer muss sich zunächst als Kunde registrieren. Dabei werden Name, Geburtsdatum, Post-Adresse (Postleitzahl, Ort, Straße, Hausnummer) und E-Mail-Adresse erfasst; der Kunde bekommt eine eindeutige Kundennummer.

Für eine Bestellung meldet sich der Benutzer als Kunde an und kann dann Artikel kaufen, die zu dieser Bestellung hinzugefügt werden. Am Ende des Bestellvorgangs wird dem Benutzer eine Übersicht angeboten, in der jeder Artikel dieser Bestellung mit seinem Einzelpreis, der Artikelnummer und der Artikelbezeichnung aufgelistet ist. Diese Übersicht enthält auch das Datum der Bestellung und den Gesamtpreis aller Artikel.

Als Service kann der Benutzer alle seine in letzter Zeit getätigten Bestellungen in einer Übersicht einsehen.

- 10 a) Erstellen Sie ein Klassendiagramm für obige Situation unter Verwendung der Klassen KUNDE, BESTELLUNG und ARTIKEL.

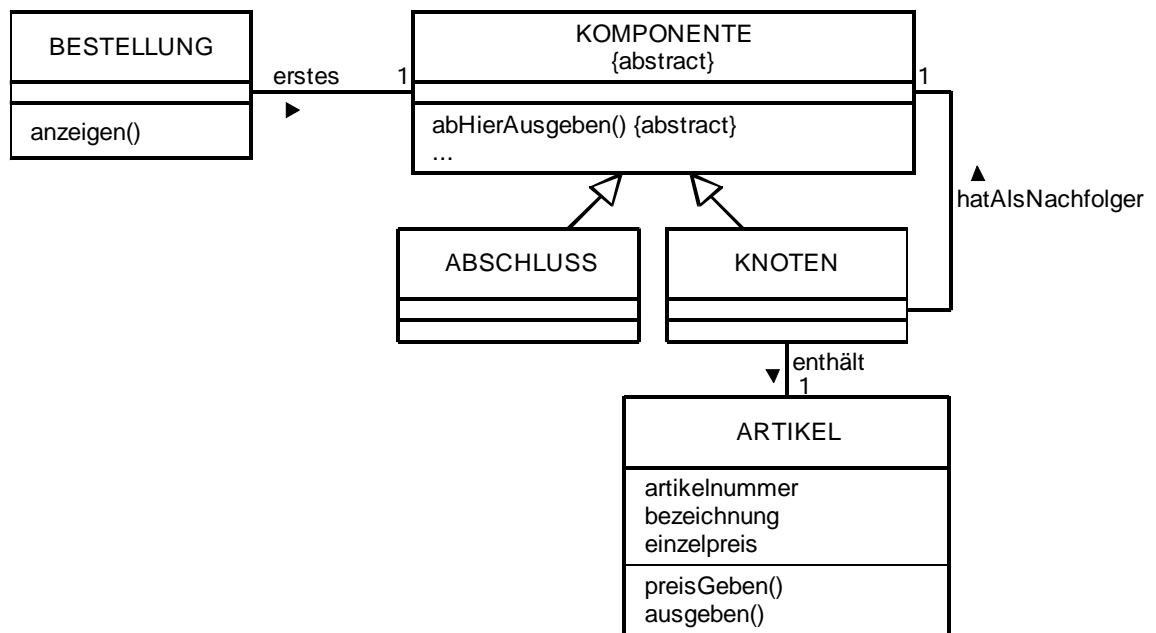
- 10 b) Die Bestellungen der Kunden werden im Versandhaus in der Reihenfolge des Eingangs bearbeitet. Es soll davon ausgegangen werden, dass sämtliche bestellten Artikel auf Lager sind. Die Wartezeit, bis eine Bestellung verschickt wird, hängt davon ab, wie schnell der Mitarbeiter die Ware verpacken kann. Die aufgelaufenen Bestellungen werden vom EDV-System in einer Warteschlange gespeichert.

i) Beschreiben Sie grundlegende Eigenschaften der Datenstruktur Schlange und erläutern Sie, weshalb sich diese zur Implementierung der Bestellliste eignet.

ii) Zeichnen Sie ein Objektdiagramm für eine Warteschlange, die vier Bestellungen enthält.

- 14 c) In einer möglichen Umsetzung werden die bestellten Artikel in einer einfach verketteten Liste verwaltet. Das zugehörige Klassendiagramm sieht im Wesentlichen folgendermaßen aus:

(Fortsetzung nächste Seite)



Die Methode `anzeigen()` der Klasse **BESTELLUNG** listet alle bestellten Artikel mit Artikelnummer, Bezeichnung und Einzelpreis auf und gibt dann auch den Gesamtpreis aller Artikel aus. Implementieren Sie diese Methode und alle dazu benötigten Methoden anderer Klassen, indem Sie erstens Methoden für die Ausgabe der Artikel und zweitens Methoden für die Preisberechnung schreiben. Verwenden Sie eine einfache textuelle Ausgabe auf dem Bildschirm.

12

- d) Das Versandhaus führt verschiedene Artikel, die alle eine Artikelnummer, eine Bezeichnung und einen Preis haben. Darüber hinaus werden bei Kleidungsstücken noch Farbname und Konfektionsgröße, bei Schuhen die Schuhgröße angegeben. Bei Schmuck gibt es keine zusätzlichen Informationen.
- Stellen Sie diese Situation in einem Klassendiagramm dar.
 - Implementieren Sie nun eine Methode `etikettieren()` der Klasse **ARTIKEL**, die einen Artikel mit allen Attributwerten auf ein Etikett druckt. Es darf davon ausgegangen werden, dass eine Methode `drucken(String text)` für die Klasse **ARTIKEL** vorhanden ist, die eine Zeile des Etiketts beschriftet.
 - Implementieren Sie für die Klasse **KLEIDUNG** einen Konstruktor mit den Parametern Artikelnummer, Bezeichnung, Preis, Farbname und Größe und eine Methode `etikettieren()` analog zur Aufgabe ii. Berücksichtigen Sie das Konzept der Datenkapselung.

(Fortsetzung nächste Seite)

6

- e) Bei der Lagerverwaltung werden alle vorhandenen Artikel in einer Datenbank gespeichert. Es sollen im Folgenden alle Artikel aus der Tabelle *artikel_tab* mit der Bezeichnung „Hose“ ausgegeben werden.
- Geben Sie den hierfür geeigneten SQL-Befehl an.
 - Geben Sie an, welche grundsätzlichen Schritte nötig sind, um diese Datenbankauswertung von einem Programm auszuführen. Gehen Sie insbesondere auf das Auslesen des von der Datenbank gelieferten Ergebnisses ein.

2. Hans bewältigt seine üblichen Wege in der Stadt mit dem Fahrrad und benötigt dafür jeweils für den Hin- und Rückweg folgende Zeiten:

Von der eigenen Wohnung (W) zur Schule (S): 15 min

Von der Schule (S) zur Wohnung des Freundes Thomas (T): 7 min

Von der Schule (S) zur Pizzeria (P): 5 min

Von der Pizzeria (P) zum Schwimmbad (B): 25 min

Vom Schwimmbad (B) zur Wohnung der Freundin Anne (A): 13 min

Von Anne (A) nach Hause (W): 12 min

Von Anne (A) zum Fußballplatz (F): 5 min

Von der Schule (S) zum Fußballplatz (F): 20 min

Vom Fußballplatz (F) nach Hause (W): 30 min

5

- a) Stellen Sie diese Situation als Graph dar. Verwenden Sie für die Orte die angegebenen Abkürzungen und verzichten Sie auf Einheiten. Um welche Art von Graph handelt es sich?

4

- b) Stellen Sie diesen Graph als Adjazenzmatrix dar.

4

- c) In einer allgemeinen Klasse GRAPH soll diese Adjazenzmatrix realisiert werden. Geben Sie dafür nötige Attribute der Klasse GRAPH sowie deren Datentypen an.

3

- d) Unter welcher Bedingung ist ein Graph ein Baum? Untersuchen Sie, ob es sich bei obigem Beispiel um einen Baum handelt.

12

- e) Hans möchte mit einer Fahrt, beginnend bei sich zuhause, die Schülerzeitung an allen oben genannten Orten vorbeibringen.

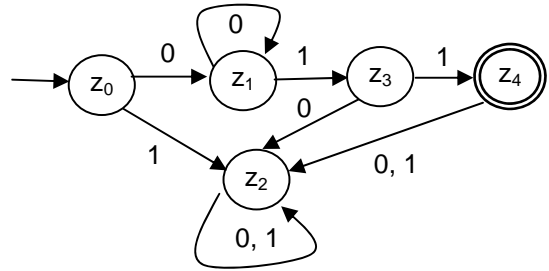
Beschreiben Sie einen Algorithmus, bei dessen Abarbeitung ausgehend von einem Startknoten alle Knoten eines Graphen besucht werden.

Inf2. THEORETISCHE UND TECHNISCHE INFORMATIK

BE

III.

1. Ein elektronisches Schloss kann über einen Code, bestehend aus einer Folge von Nullen und Einsen, geöffnet werden. Der Code ist dabei aber nicht fest vorgegeben, sondern muss lediglich einem gewissen Muster entsprechen, das durch den nebenstehenden Automaten repräsentiert wird.



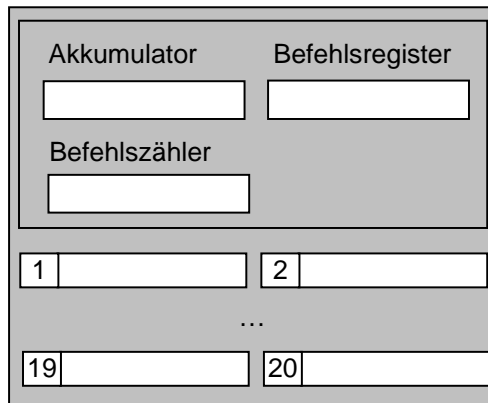
- 3 a) Öffnen folgende Codes das Schloss? Begründen Sie Ihre Entscheidung.
- 0110
 - 00011
- 4 b) Beschreiben Sie in Worten, welches Codemuster vom Automaten erkannt wird. Geben Sie dieses Muster auch in einfacher Textnotation wie beispielsweise der Backus-Naur-Form an.
- 11 c) Implementieren Sie einen entsprechenden Automaten mithilfe einer auf dem Deckblatt vermerkten objektorientierten Sprache. Dabei soll es u. a. eine Methode *codeTesten* geben, die für einen eingegebenen Code überprüft, ob er von dem elektronischen Schloss akzeptiert wird und eine entsprechende Meldung auf dem Bildschirm ausgibt.
- [Hinweis: Sie dürfen folgende Methoden einer Klasse String verwenden:
- *int length()* liefert die Länge des Strings;
 - *char charAt(n)* liefert das n-te Zeichen des mindestens ein Zeichen langen Strings; Zählung beginnt bei 0.]

(Fortsetzung nächste Seite)

2. Gegeben ist eine Registermaschine, die nachfolgend aufgelisteten Befehlssatz zur Verfügung stellt:

load xx lädt den Wert aus xx in den Akkumulator.
 store xx schreibt den Wert aus dem Akkumulator in xx.
 add xx addiert den Wert aus xx zum Wert im Akkumulator.
 dec Wert im Akkumulator wird um 1 erniedrigt.
 end Ende des Programms
 jap xx Falls der Wert im Akkumulator positiv ist, springe zum Befehl
 in xx.
 goto xx springt zum Befehl in xx.

xx ist die Registeradresse, d. h. die Adresse der Speicherzelle
 $xx \in \{1; 2; \dots; 20\}$.



6

a) Grundlage der (verallgemeinerten) Registermaschine ist das von-Neumann-Modell. Geben Sie die grundlegenden Funktionseinheiten dieses Modells an und erläutern Sie das Prinzip des zugrundegelegten Speichers sowie die Bedeutung von Sprungbefehlen in diesem Modell.

2

b) Im Speicher mit der Adresse 19 ist eine ganze Zahl abgespeichert. Beschreiben Sie den Speicherinhalt nach dem Durchlauf des folgenden Programms, das in den Speicherzellen 1 bis 4 abgespeichert ist:

1 LOAD 19 2 ADD 19 3 STORE 20 4 END

(Fortsetzung nächste Seite)

Die Registermaschine soll zwei natürliche Zahlen a , abgespeichert im Register mit der Adresse 18, und b , abgespeichert im Register mit der Adresse 19, multiplizieren und das Ergebnis anschließend in Zelle 20, die mit 0 vorbelegt ist, schreiben. Da kein Multiplikationsbefehl zur Verfügung steht, wird folgender Algorithmus angewandt:

```
c = 0;
wiederhole solange b > 0
    c = c + a;
    b = b - 1;
endeWiederhole
```

- 3 c) Erklären Sie das Prinzip des Algorithmus für $a = 4$ und $b = 3$.
- 6 d) Schreiben Sie ein Programm für die Registermaschine, das den obigen Algorithmus für die Zahlen 4 und 3 umsetzt. Geben Sie dabei (analog zu Aufgabe 2b) die Adresse der Register an, in der der jeweilige Befehl abgespeichert ist.
3. Brute-Force-Verfahren können in verschiedenen Anwendungsbereichen der Informatik eine Rolle spielen.
- 2 a) Erläutern Sie kurz das Grundprinzip des Brute-Force-Verfahrens im Zusammenhang mit Verschlüsselung.
- 3 b) Sie möchten ein Passwort „knacken“, das aus 6 Kleinbuchstaben (ohne Umlauten) besteht. Ihr Rechner kann ca. 20 000 000 solcher Passwörter pro Sekunde erzeugen und testen. Wie lange braucht der Rechner ungefähr, um alle Möglichkeiten zu überprüfen?

Inf2. THEORETISCHE UND TECHNISCHE INFORMATIK

BE

IV.

1. Bei den Computernetzwerken in Schulen kommt heute meist das IP-Protokoll zum Einsatz. Protokolle spielen beispielsweise bei Schichtenmodellen eine große Rolle.

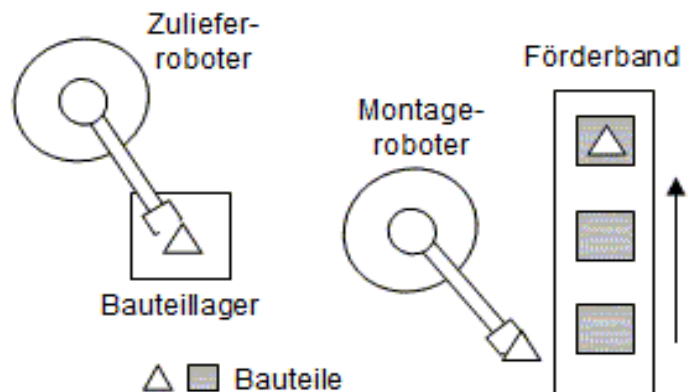
4 a) Legen Sie kurz dar, welche Vorteile sich durch deren Einsatz von Schichtenmodellen ergeben.

8 b) Das Schichtenmodell kann man sich an folgendem „Alltagsbeispiel“ verdeutlichen:

Ein chinesischer und ein deutscher Geschäftspartner möchten miteinander per (Post-)Brief kommunizieren. Beide Personen beherrschen leider nur ihre Muttersprache. Dem Chinesen steht aber ein Übersetzer mit den Sprachen Chinesisch und Französisch zur Verfügung. Der deutsche Partner kann auf einen Übersetzer mit den Sprachen Deutsch und Französisch zurückgreifen.

Geben Sie ein Schichtenmodell mit (mindestens) 4 Schichten für diesen Kommunikationsablauf an und erklären Sie anhand dieses Modells beispielhaft den Ablauf der Kommunikation zwischen den Partnern.

2. Ein Montageroboter setzt aus zwei Bauteilen ein Produkt zusammen. Dabei holt er sich ein dreieckiges Bauteil aus einem Bauteillager und montiert dieses dann auf ein quadratisches Bauteil, das durch ein Förderband zugeführt



wird. Das Bauteillager, das maximal 10 Stück dreieckiger Bauteile fassen kann, wird durch einen zweiten Roboter, den Zulieferroboter, befüllt. Das System lässt die beiden Roboter nie gleichzeitig auf das Bauteillager zugreifen, um eine Zerstörung der Greifarme zu verhindern. Außerdem ist gewährleistet, dass genügend viele dreieckige und quadratische Bauteile von außen zugeführt werden.

(Fortsetzung nächste Seite)

Das Bauteillager ist zu Beginn der Montage ausreichend gefüllt. Der Füllstand des Bauteillagers kann von den Robotern erst nach dem Einschwenken in das Bauteillager geprüft werden. Außerdem können sie das Bauteillager erst wieder verlassen, wenn sie ein Bauteil abgelegt bzw. aufgenommen haben.

3 a) Erläutern Sie kurz ein informatisches Konzept, wie der kritische Bereich „Bauteillager“ geschützt werden kann.

3 b) Was versteht man in der Informatik allgemein unter einer Verklemmung?

6 c) Zeigen Sie durch Beschreibung und Erklärung einer bestimmten Situation, dass es im oben beschriebenen Szenario zu einer Verklemmung kommen kann, wenn der Zulieferroboter

i) langsamer arbeitet als der Montageroboter,

ii) schneller arbeitet als der Montageroboter.

Folgender Vorschlag wird zur Verklemmungsvermeidung gemacht: Das Bauteillager soll seinen aktuellen Status ausgeben können; dabei gibt es genau eine Statusmeldung, nämlich „Lager voll“. Die Roboter können den Status vor dem Eintritt in den Lagerbereich abfragen und ggf. das Einschwenken in das Lager abbrechen.

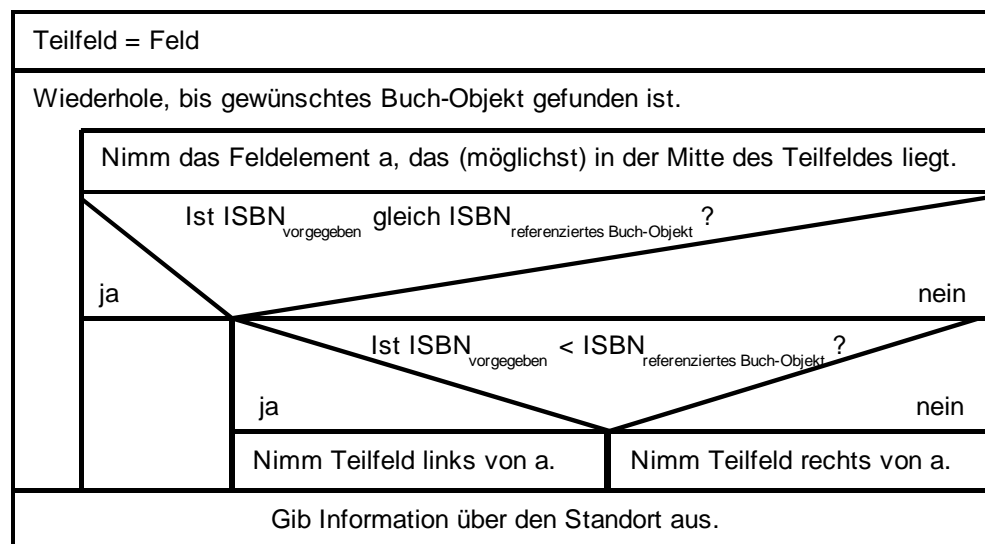
6 d) Diskutieren Sie, ob sich damit die Verklemmungen aus Teilaufgabe 2c tatsächlich vermeiden lassen. Gehen Sie davon aus, dass die Statusabfrage nur möglich ist, wenn sich kein Roboter im Lager befindet und der Roboter, der die Statusabfrage zuerst stellt, bei einem eventuellen Zugriff auf das Bauteillager bevorzugt wird.

(Fortsetzung nächste Seite)

3. Eine Buchverwaltungssoftware verwaltet n Bücher einer Bibliothek über ein Feld der Länge n , dessen Elemente Referenzen auf die Buch-Objekte enthalten. Die Buch-Objekte enthalten u. a. die ISBN-Nummer und den Standort des entsprechenden Buches. Die Elemente des Feldes sind nach der ISBN-Nummer der Buch-Objekte geordnet.

Nun soll der Standort eines (in der Bibliothek vorhandenen) Buches unter Angabe der ISBN-Nummer gesucht werden, wobei zwei Strategien angewandt werden:

- Strategie 1: Das Feld wird linear, d. h. aufsteigend von Feldelement zu Feldelement, durchsucht. Das Verfahren wird abgebrochen, sobald das gesuchte Buch-Objekt gefunden ist.
- Strategie 2:



- 5 a) Der Rechner benötigt nun für die Überprüfung eines Buch-Objekts (inklusive aller dazu notwendigen Maßnahmen) t Millisekunden. Die Feldlänge beträgt 30 und die Referenz auf das gesuchte Buch-Objekt befindet sich im 19-ten Feldelement. Wie lange dauert jeweils die Suche nach dem Buch-Objekt in beiden Strategien? Begründen Sie Ihre Antwort.
- 3 b) Im Fall von Strategie 2 dauert die Suche im ungünstigsten Fall ungefähr $(t \cdot \log_2 n)$ Millisekunden, wobei n die Feldlänge ist. $\log_2 n$ ist auch die ungefähre Höhe eines binären Baumes, bei dem alle Pfade annähernd gleich lang sind. Erklären Sie diesen Zusammenhang.
- 2 c) Nennen Sie zwei Gründe, warum die Verwendung eines Feldes im Falle einer Bibliothekssoftware nicht zweckmäßig ist.