

Matrikelnr. 

--	--	--	--	--	--	--	--

Name \_\_\_\_\_

Vorname \_\_\_\_\_

KLAUSUR: Entwurf und Implementierung von Informationssystemen (32561)  
TERMIN: 14.09.2016, 14:00–16:00 Uhr  
PRÜFER: Univ.-Prof. Dr. rer. pol. habil. S. Strecker

**Aufbau und Bewertung der Klausur**

Aufgabe	1	2	3	Summe
Maximal erreichbare Punktzahl	32	37	31	100
Erreichte Punktzahl				

DATUM:

NOTE:

## Allgemeine Hinweise



Tragen Sie bitte jetzt Ihre Matrikelnummer, Ihren Namen und Vornamen auf dem Deckblatt ein.  
Versehen Sie bitte zusätzlich jede Seite mit Ihrer Matrikelnummer (oben rechts).

## Hinweise zur Bearbeitung

**Für die Bearbeitung der insgesamt 3 Klausuraufgaben auf den folgenden 13 Seiten dieser Klausur stehen Ihnen 120 Minuten zur Verfügung.**

1. Außer Schreibgeräten sind keine Hilfsmittel zugelassen.
2. Die Lösungen müssen in den vorgesehenen Raum auf den Aufgabenblättern eingetragen werden.
3. Notizen können auf den Rückseiten der Aufgabenblätter gemacht werden. Diese Anmerkungen werden in die Bewertung nicht einbezogen.
4. Bei Beendigung der Klausur müssen das Deckblatt und die Aufgabenblätter abgegeben werden. Trennen Sie bitte nicht einzelne Blätter ab.

Viel Erfolg!

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

**Aufgabe 1 (Objektorientierter Entwurf)****(32P)**

a) Erläutern Sie in eigenen Worten, was unter dem im Lehrbrief dargestellten Begriff „Framework“ zu verstehen ist. Erläutern Sie im Anschluss, welche Vorteile mit Klassenbibliotheken und Frameworks einhergehen. Welcher Nachteil ist mit der Verwendung verbunden? **(6P)**

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

b) Erläutern Sie in eigenen Worten, was unter dem im Lehrbrief dargestellten Konzept des „Polymorphismus“ zu verstehen ist. Erläutern Sie weiter, welche Arten des Konzeptes zu unterscheiden sind und welche Eigenschaften diese aufweisen. **(6P)**

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

c) Geben Sie an, ob die nachfolgend aufgeführten Aussagen zutreffen oder nicht. Tragen Sie hierzu jeweils in dem vorgegebenen Kreis ein "R" für richtig oder ein "F" für falsch ein. Für diese Aufgabe gibt es maximal 20 Punkte. Die Bestehensgrenze dieser Aufgabe von 10 Punkten erreichen Sie mit 6 richtigen Antworten. **(20P)**

Von einer Wald-Topologie wird gesprochen, wenn eine Klassenbibliothek in mehrere Vererbungshierarchien zerfällt, wobei jede Hierarchie dabei mehrere logisch zusammenhängende Klassen zu einer abhängigen Komponente bündelt.

Eine zentrale Aufgabe des objektorientierten Entwurfs einer Anwendung ist die Spezifikation ihrer Architektur.

Eine verteilte Anwendung gliedert sich in mehrere selbständig lauffähige Programme, die auf verschiedenen, räumlich getrennten Computern gleichzeitig ausgeführt werden.

Dynamischer Polymorphismus setzt eine formale Spezifikation der Operationen von Klassenhierarchien voraus.

Ein wesentlicher Vorzug des objektorientierten Paradigmas der Softwareentwicklung gegenüber dem strukturierten Paradigma besteht in der Umsetzung der Datenkapselung.

Durch die Nutzung der Vererbung in den Topologien für Klassenbibliotheken, wie z. B. der Baum- und der Waldtopologie, wird die Laufzeit verkürzt.

Neben der einfachen Vererbung mit nur einer Oberklasse kann im objektorientierten Entwurf auch die Mehrfachvererbung verwendet werden, bei der abgeleitete Klassen mehrere Oberklassen besitzen.

In einer Fünf-Schichten-Architektur übernimmt die Fachkonzept-Zugriffsschicht Aufgaben der Datenkonvertierung, um die GUI-Schicht hiervon zu entlasten.

Das Singleton-Muster ist ein objektbasiertes Erzeugungsmuster, das gewährleistet, dass auf das erzeugte Objekt ausschließlich lokal zugegriffen werden kann.

Die internationale Einsatzfähigkeit einer Anwendung ist eine nicht-funktionale Anforderung, die die Auswahl der Implementierungssprache beeinflusst.

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

**Aufgabe 2 (Algorithmen & Datenstrukturen)****(37P)**

In der gesamten Aufgabe 2 wird von Ihnen erwartet, dass Sie die im Lehrbrief dargestellte, an PASCAL angelehnte Pseudocode-Notation ausnahmslos anwenden. Für das Algorithmen mit diesem Pseudocode stehen damit die spezifischen Konzepte von PASCAL zur Verfügung, d. h. verschiedene einfache und zusammengesetzte Datentypen, Konstrukte der strukturierten Programmierung und das Prozedurkonzept. Alle Teilaufgaben sind als Codefragmente in der im Lehrbrief dargestellten Pseudocode-Notation zu erstellen. Andere Pseudocode-Notationen oder Programmiersprachen werden nicht bewertet.

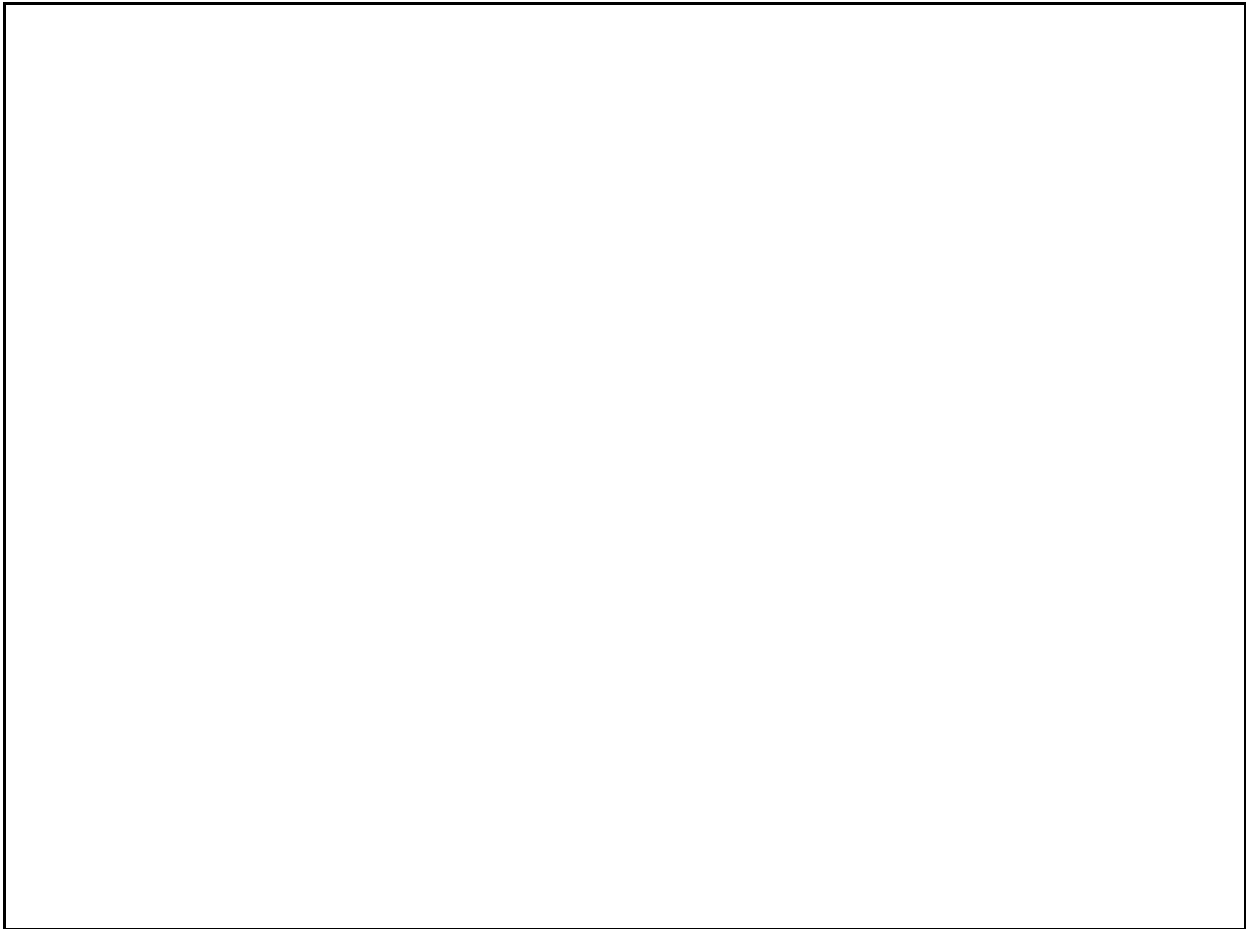
a) Die rekursive Definition der Fakultät  $f(n)$  lautet für alle natürlichen Zahlen  $n$ : **(4P)**

$$f(n) = f(n - 1) \cdot n \text{ mit } f(0) = 1$$

Schreiben Sie ein Programm, das die Fakultät gemäß der rekursiven Definition berechnet und testen Sie das Programm für  $n=4$ . Verwenden Sie den Datentyp `INTEGER` für  $n$ .

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

b) Schreiben Sie ein Programm, das die Fakultät aus Aufgabenteil a) ohne Rekursion berechnet. **(4P)**



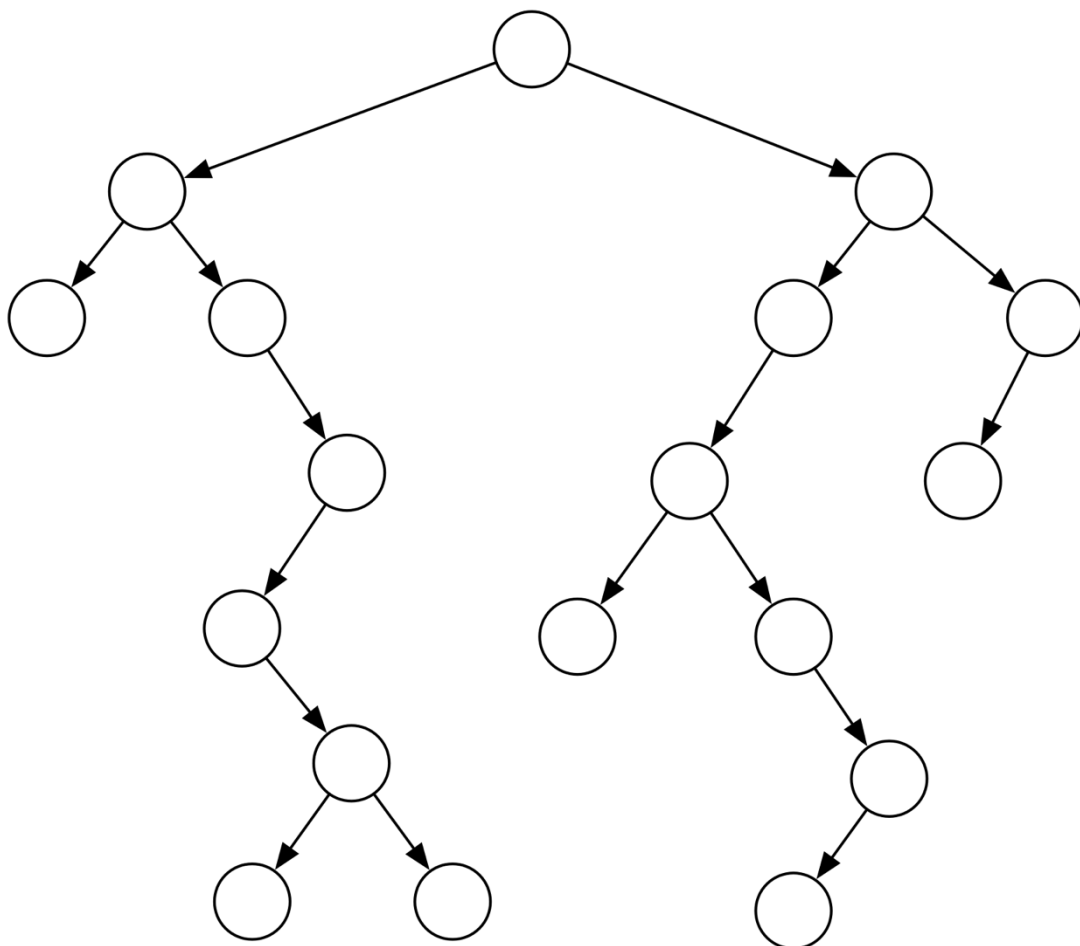
c) Erläutern Sie kurz, was unter den Begriffen *Rekursion* und *Iteration* zu verstehen ist. An welche Bedingungen ist ein *rekursiver Aufruf* einer Prozedur geknüpft? **(4P)**



Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Mit dem Begriff „Traversieren“ wird das Durchlaufen sämtlicher Knoten eines Baumes in einer bestimmten Reihenfolge bezeichnet. In der Regel wird mit dem Traversieren die Bearbeitung vieler oder aller Knoten bzw. Datenobjekte verbunden sein. Für Binärbäume eignen sich insbesondere auch rekursive Traversierungsverfahren.

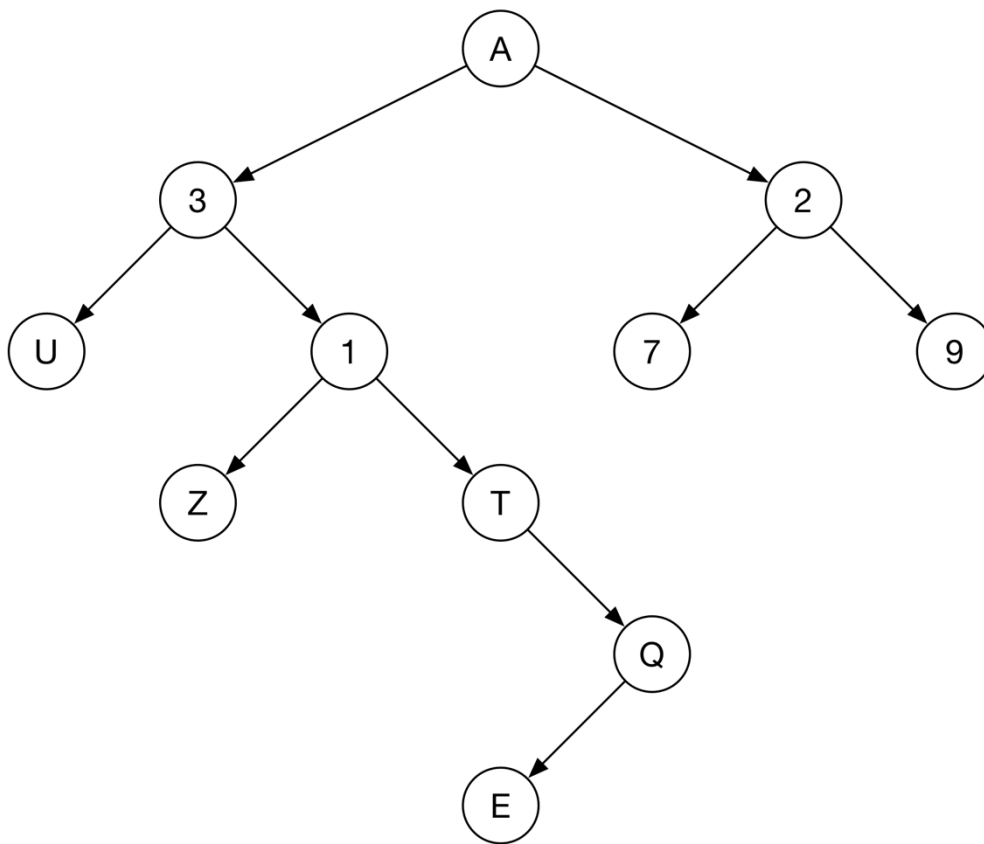
d) Tragen Sie in den nachstehenden Binärbaum die Reihenfolge, beginnend mit der Zahl 1, des Ansprechens der Knoten für den Fall des Traversierens in *Postordnung* ein. **(9P)**



Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.



e) Geben Sie die Ausgabe des Traversierens des dargestellten Binärbaumes in (1) symmetrischer Ordnung, (2) Präordnung und (3) Postordnung an und skizzieren Sie Ihr Vorgehen in drei Schritten. **(6P)**



	Ausgabe:	Vorgehen:
(1) Symmetrischer Ordnung:		
(2) Präordnung:		
(3) Postordnung:		

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

f) Betrachtet sei ein Sortieralgorithmus für sequentiell gespeicherte Objekte. Die Tabelle unten zeigt die Schritte eines solchen Algorithmus sowie die Ausgangs- und Zielfolge. **(10P)**

(1) Benennen Sie den Algorithmus und erläutern Sie das Vorgehen des Sortierens.

(2) Entwickeln Sie für den dargestellten Sortieralgorithmus eine Prozedur

	Ausgangsfolge								
Ausgangszustand	:	42	51	13	48	92	16	09	68
1. Schritt	:	42	13	48	51	16	09	68	92
2. Schritt	:	13	42	48	16	09	51	68	92
3. Schritt	:	13	42	16	09	48	51	68	92
4. Schritt	:	13	16	09	42	48	51	68	92
5. Schritt	:	13	09	16	42	48	51	68	92
6. Schritt	:	09	13	16	42	48	51	68	92
7. Schritt	:	09	13	16	42	48	51	68	92
		Zielfolge							

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

```
PROCEDURE sortalgorithmus(VARIABLE feld: FELD);
DATA
  VARIABLE i,j :INDEX;
           element :INTEGER;
BEGIN

END sortalgorithmus;
```

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

**Aufgabe 3 (Programmieren in C)****(35P)**

In der gesamten Aufgabe 3 wird von Ihnen erwartet, dass Sie die im Lehrbrief dargestellte Programmiersprache C ausnahmslos anwenden. Für die Implementierung mit dieser Programmiersprache stehen Ihnen damit die spezifischen Sprachkonzepte von C zur Verfügung.

a) Entwerfen Sie eine Funktion, die eine Zahl  $n$  einlesen und dann die Summe aller Quadratzahlen bis zu dieser Zahl ausgeben kann. Dazu ruft das entwickelte C-Programm die Funktion `quadsum()` auf, die diese Summe rekursiv ermitteln muss, wobei sie auch zusätzlich noch die einzelnen Quadratzahlen ausgeben soll. **(8P)**

```
#include <stdio.h>

int quadsum(int n)
{

}

int main(void){
int n;
printf("Bitte eine Zahl eingeben: ");
scanf("%d", &n);
printf("%d\n", quadsum(n));
return 0;
}
```

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

b) „Der Einsatz verketteter Datenstrukturen sollte statischen Datenstrukturen stets vorgezogen werden.“ Diskutieren Sie diese Aussage unter verschiedenen Gesichtspunkten. **(5P)**

c) Schreiben Sie ein Programm, das die Bücher einer Buchhandlung verwalten kann. Dazu benötigen Sie eine Typenvereinbarung bzw. Struktur (`buch`), die Inventarnummer, Titel (maximal 30 Zeichen), Preis, Mwst und Stückzahl für bis zu 100 Bücher speichern kann. Das Programm liest als *erstes* die Mehrwertsteuer ein und dann in einer `for`-Schleife die restlichen Daten *aller* Bücher. **(10P)**

```
#include <stdio.h>
```

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.



Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

d) Schreiben Sie folgende `for`-Schleife als `while`-Schleife mit gleicher Bedeutung: **(4P)**

```
int summand, summe;
for (summand=10, summe=0; summand>0; summand--)
{
    printf(„\nZwischensumme: %d“, summe);
    summe+=summand;
    summand--;
}
```

```
#include <stdio.h>
```

e) Nennen Sie *drei* Typen verketteter Datenstrukturen und erläutern Sie kurz, wie die Knoten eines Baumes mit „Vorgängern“ und „Nachfolgern“ zusammenhängen. Orientieren Sie sich an den im Lehrbrief dargestellten Typen *verketteter Datenstrukturen*. **(4P)**

Diese Klausur ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der FernUniversität reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.