

**Sammlung aller Staatsexamensaufgaben der  
Prüfungsnummer**

**66111**

**Betriebssysteme / Datenbanksysteme /  
Rechnerarchitektur (vertieft)**

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1989**

Prüfungsteilnehmer
Kennzahl: _____
Kennwort: _____
Arbeitsplatz-Nr.: _____

Prüfungstermin
<b>HERBST</b>
<b>1989</b>

Einzelprüfungsnummer
<b>66111</b>

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (nicht vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architektur

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 6

bitte wenden!

**Sämtliche Teilaufgaben sind zu bearbeiten!****Teilaufgabe 1**

Für einen endlichen nicht-leeren Zeichenvorrat  $Z$  sei eine Binärcodierung

$$c: Z \rightarrow \{O, L\}^*$$

gegeben.

1.1 Geben Sie eine Datenstruktur (durch Typdefinitionen) an, mit deren Hilfe ein beliebiger Binärcode  $c$  im Hinblick auf die folgende Teilaufgabe 1.2 geeignet dargestellt werden kann.

1.2 Schreiben Sie eine Funktionsprozedur *fano*, mit deren Hilfe festgestellt werden kann, ob ein Binärcode  $c$  der Fano-Bedingung genügt. Die Funktionsprozedur *fano* soll dabei folgender Spezifikation genügen:

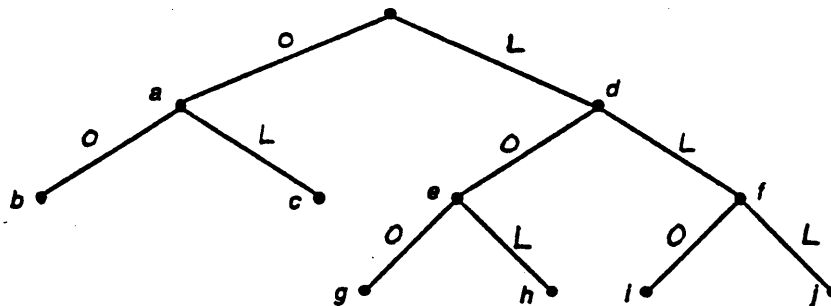
1.2.1 Eingabeparameter:  $c$  vom Typ *CODE*

1.2.2 Der Funktionswert ist vom Typ *BOOLEAN* und hat den Wert *TRUE* genau dann, wenn  $c$  der Fano-Bedingung genügt.

Hinweise: - Der Typ *CODE* muß in Ihrer Antwort zu 1.1 definiert werden.

- Ein Code erfüllt die Fano-Bedingung genau dann, wenn in ihm kein Codewort Anfang eines von ihm verschiedenen Codewortes ist.

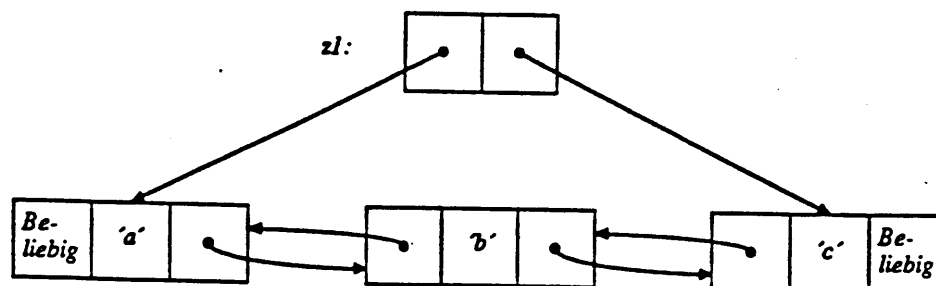
1.3 Der Zeichenvorrat  $Z = \{a, b, c, d, e, f, g, h, i, j\}$  läßt sich durch Wörter aus  $\{O, L\}^*$  binär codieren, die höchstens die Länge 3 haben. Ein Beispiel dafür ist durch den folgenden Code-Baum gegeben:



Zeigen Sie, daß es keine 3-stellige Binärcodierung (d.h. mit Codewortlänge  $\leq 3$ ) für  $Z$  gibt, die die Fano-Bedingung erfüllt.

**Teilaufgabe 2**

Zeichenreihen über dem endlichen Alphabet  $A = ('a', 'b', \dots, 'z')$  sollen durch vorwärts und rückwärts verkettete Listen dargestellt werden, für deren Anfang und Ende Anker (Verweise) vorgesehen sind. Die Darstellung der Zeichenreihe  $z1 = 'abc'$  läßt sich also beispielsweise folgendermaßen graphisch veranschaulichen:



- 2.1 Geben Sie eine Datenstruktur mit Hilfe geeigneter Typdefinitionen an, durch die sich die o.a. Darstellung von Zeichenreihen realisieren läßt.
- 2.2 Wie kann die leere Zeichenreihe  $\epsilon$  dargestellt werden?
- 2.3 Schreiben Sie eine Funktionsprozedur *istepsilon*, die feststellt, ob eine Zeichenreihe gleich  $\epsilon$  ist, also folgender Spezifikation genügt:
  - 2.3.1 Eingabeparameter:  $x1$  vom Typ *ZEICHENREIHE*  
 Hinweis: *ZEICHENREIHE* muß als Typ in Ihrer Antwort zu 2.1 definiert sein.
  - 2.3.2 Der Funktionswert ist vom Typ *BOOLEAN* und ist genau dann gleich *TRUE*, wenn  $x1 = \epsilon$  ist.

2.4 Schreiben Sie zwei Funktionsprozeduren *anfang* und *teil*, die feststellen, ob  $x$  Anfangs- bzw. Teilzeichenreihe einer Zeichenreihe  $y$  ist. Die entsprechenden Spezifikationen lauten:

2.4.1 Eingabeparameter (für beide Funktionsprozeduren):  $x, y$  vom Typ *ZEICHENREIHE*

2.4.2 Der Funktionswert ist jeweils vom Typ *BOOLEAN*. Er hat den Wert *TRUE* für die Funktion *anfang* bzw. *teil* genau dann, wenn  $x$  Anfangs- bzw. Teilzeichenreihe von  $y$  ist.

Hinweise: -  $x$  ist genau dann Anfangszeichenreihe von  $y$ , wenn es ein  $y'$  so gibt, daß  $y = xy'$  ist.

-  $x$  ist genau dann Teilzeichenreihe von  $y$ , wenn es ein  $y'$  und ein  $y''$  so gibt, daß  $y = y'xy''$  ist.

- Es ist empfehlenswert, erst die Funktionsprozedur *anfang* zu schreiben und dann in der Funktionsprozedur *teil* die Funktion *anfang* zu verwenden. Dabei darf auch die in 2.3 spezifizierte Funktion *istepsilon* verwendet werden.

2.5 Ein Palindrom ist eine Zeichenreihe, die mit ihrer rückwärts gelesenen identisch ist. Beispiele sind

'otto', 'reliefpfeiler',  $\epsilon$  (leere Zeichenreihe).

Schreiben Sie eine Funktionsprozedur *palin*, die feststellt, ob eine Zeichenreihe ein Palindrom ist, also folgender Spezifikation genügt:

2.5.1 Eingabeparameter:  $x1$  vom Typ *ZEICHENREIHE*

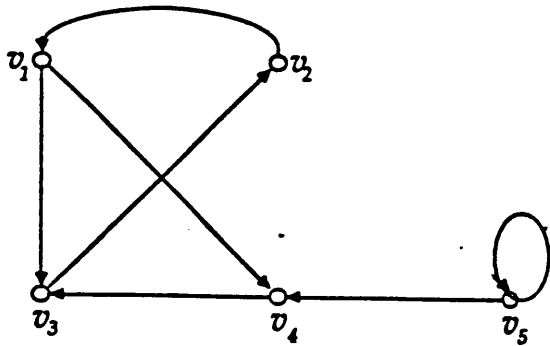
2.5.2 Der Funktionswert ist vom Typ *BOOLEAN* und ist genau dann gleich *TRUE*, wenn  $x1$  ein Palindrom ist.

Hinweis: In der Funktionsprozedur *palin* darf die unter 2.3 spezifizierte Funktion *istepsilon* verwendet werden.

Fortsetzung nächste Seite!

Teilaufgabe 3

Ein endlicher gerichteter Graph  $G = (V, E)$  mit der Knotenmenge  $V = \{v_1, v_2, \dots, v_n\}$  und der Menge gerichteter Kanten  $E \subseteq V \times V$  kann u.a. durch seine direkte graphische Darstellung und durch seine Adjazenzmatrix dargestellt werden. Ein Beispiel:



Dieser direkt dargestellte Graph mit  $V = \{v_1, v_2, v_3, v_4, v_5\}$  und  $E = \{(v_1, v_3), (v_1, v_4), (v_2, v_1), (v_3, v_2), (v_4, v_3), (v_5, v_4), (v_5, v_5)\}$  läßt sich auch durch seine Adjazenzmatrix

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

darstellen. Allgemein gilt für die Komponenten  $a_{ij}$  der Adjazenzmatrix  $A$ :

$$a_{ij} = \begin{cases} 1 & \text{falls es eine Kante von } v_i \text{ nach } v_j \text{ gibt} \\ 0 & \text{sonst} \end{cases}$$

Hat ein gerichteter Graph die Eigenschaft, daß von wenigstens einem Knoten zwei gerichtete Kanten und von keinem Knoten mehr als zwei gerichtete Kanten ausgehen, so sagt man, daß er den Verzweigungsgrad 2 hat. Der angegebene Beispiel-Graph hat also den Verzweigungsgrad 2. Solche Graphen können in direkter Darstellung durch einen Verweis auf ein Geflecht von Verbunden der Art *knoten* entsprechend folgender Typ-Vereinbarung aufgebaut werden:

```
TYPE graph    = ↑ knoten;
    knoten    = RECORD inhalt: INTEGER;
                      nachf1, nachf2: graph
    END;
```

Mit VAR  $g$ : *graph* sei nun eine Variable gegeben, die auf einen endlichen gerichteten Graphen mit  $n$  Knoten und mit Verzweigungsgrad 2 verweist.

3.1 Geben Sie einen terminierenden Algorithmus an, der die Adjazenzmatrix von  $g$  berechnet.

Hinweis: Es wird nicht verlangt, daß dafür ein Programm geschrieben wird; der Algorithmus soll aber klar und korrekt beschrieben werden.

3.2 Als Wegematrix  $W$  zu einem endlichen gerichteten Graphen  $G$  bezeichnet man die Matrix, die durch ihre Komponente  $w_{ij}$  angibt, ob es in  $G$  einen gerichteten Kantenzug von  $v_i$  nach  $v_j$  gibt.  $W = (w_{ij})_{1 \leq i, j \leq n}$  ist also folgendermaßen definiert:

$$w_{ij} = \begin{cases} 1 & \text{falls ein gerichteter Kantenzug von } v_i \text{ nach } v_j \text{ existiert} \\ 0 & \text{sonst} \end{cases}$$

Zu dem o.a. Graphen ist beispielsweise

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

die zugehörige Wegematrix.

Schreiben Sie eine Prozedur *weg*, die zur Adjazenzmatrix  $A$  eines endlichen gerichteten Graphen mit  $n$  Knoten und mit einem beliebigen Verzweigungsgrad ( $\leq n$ ) die Wegematrix  $W$  berechnet. Die Prozedur *weg* soll also folgender Spezifikation genügen:

3.2.1 Eingabeparameter: Adjazenzmatrix  $a$  vom Typ `ARRAY[1..n, 1..n] OF 0..1`

3.2.2 Ausgabeparameter: Wegematrix  $w$  vom Typ `ARRAY[1..n, 1..n] OF 0..1`

Dabei soll  $w$  die Wegematrix zu dem durch  $a$  dargestellten gerichteten endlichen Graphen sein.



**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Frühjahr 1990**

Prüfungsteilnehmer	Prüfungstermin	Einzelprüfungsnummer
Kennzahl: _____	<b>FRÜHJAHR</b> <b>1990</b>	<b>66111</b>
Kennwort: _____		
Arbeitsplatz-Nr.: _____		

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 5

bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

### Teilaufgabe 1

- a) Welche zwei Arten von Unterbrechungen unterscheidet man aufgrund ihrer Auslösung?  
Erklären Sie diese und geben Sie jeweils drei Beispiele für jede Art an.
- b) In welcher Situation im Befehlszyklus (Mikroprogramm des Rechnerkerns) wird abgefragt, ob ein Unterbrechungswunsch vorliegt?  
In welchem Fall wird dieser zurückgestellt?  
Für welche Unterbrechungsart kann er nicht zurückgestellt werden?
- c) Bei einer Unterbrechung wird außer dem Unterbrechungswunsch noch weitere Unterbrechungsinformation übertragen, die von der speziellen Unterbrechung abhängig ist. Auf welche Weise wird diese Information dem Betriebssystem übermittelt?  
Geben Sie die Unterbrechungsinformation für drei verschiedene Unterbrechungen Ihrer Wahl an.

### Teilaufgabe 2

Geben Sie diejenigen der nachfolgend genannten Bedingungen (1) bis (17) an, die gleichzeitig erfüllt sein müssen, damit bei der Vergabe von Betriebsmitteln (BM) eine Verklemmung entstehen kann.

- (1) BM sind entziehbar
- (2) BM sind nicht entziehbar
- (3) BM-Zugriff wird nicht synchronisiert
- (4) BM-Zugriff erfolgt durch Synchronisationsprozeß
- (5) BM sind zugriffsbeschränkt
- (6) BM sind gemeinsam verwendbar
- (7) Prozesse fordern BM in beliebiger Reihenfolge an
- (8) BM dürfen nicht aus mehreren unabhängig vergebbaren Einheiten bestehen
- (9) BM sind wiederverwendbar
- (10) BM sind nicht wiederverwendbar
- (11) Prozesse, die auf Zuteilung weiterer BM warten, geben bereits belegte BM nicht frei
- (12) Prozesse fordern BM schrittweise an
- (13) Prozesse fordern alle benötigten BM auf einmal an
- (14) Die maximalen BM-Forderungen sind bekannt
- (15) Die maximalen BM-Forderungen sind nicht bekannt
- (16) Die Auslastung des Systems ist hoch
- (17) Die Auslastung des Systems ist gering

Fortsetzung nächste Seite!

Teilaufgabe 3

Gegeben seien zwei Betriebsmittel  $x$  und  $y$  mit zugehörigen Koordinationsvariablen (Semaphoren)  $sx$  und  $sy$  sowie zwei Prozesse  $p1$  und  $p2$  mit:

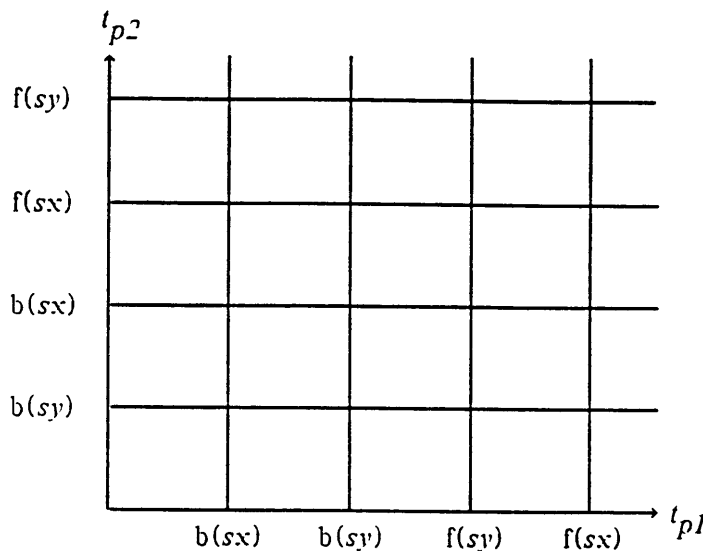
$p1$ : **begin**

SVC belegen ( $sx$ );  
 SVC belegen ( $sy$ );  
 < benutze  $x$  und  $y$  >;  
 SVC freigeben ( $sy$ );  
 SVC freigeben ( $sx$ );  
**end**

$p2$ : **begin**

SVC belegen ( $sy$ );  
 SVC belegen ( $sx$ );  
 < benutze  $x$  und  $y$  >;  
 SVC freigeben ( $sx$ );  
 SVC freigeben ( $sy$ );  
**end**

Zur Darstellung des Fortschrittes der beiden Prozesse diene das folgende Prozeß-Zeit-Diagramm:



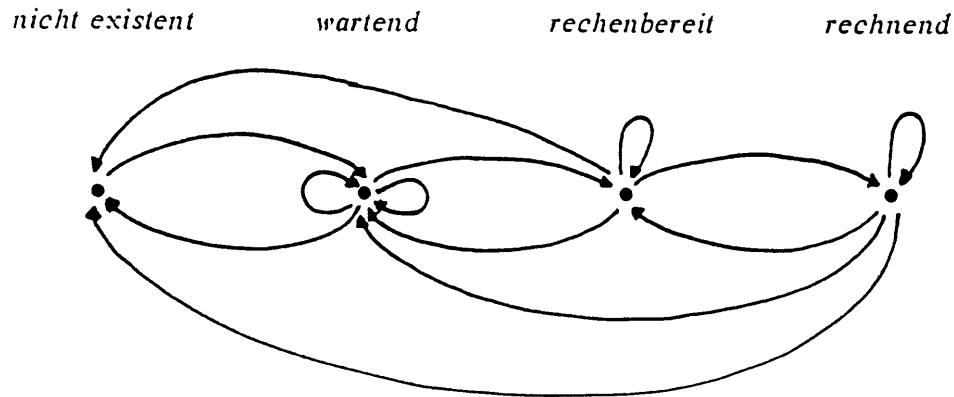
(Dabei gilt: b = belegen, f = freigeben.)

- Schraffieren Sie (///) die nicht betretbaren Gebiete.
- Schraffieren Sie (\\) die verklemmungsbedrohten Gebiete.
- Zeichnen Sie (---) einen Verlauf ein, der zur Verklemmung führt, und markieren Sie den Punkt der Verklemmung mit V.
- Zeichnen Sie (—) einen Verlauf ein, der die Verklemmungsmöglichkeit umgeht und beide Prozesse zu Ende führt.
- Ändern Sie das angegebene Programm für  $p2$  so ab, daß Verklemmung ausgeschlossen ist.
- Begründen Sie Ihre Lösung zu e) mit Hilfe des zugehörigen Prozeß-Zeit-Diagramms, in das Sie Einträge gemäß a) und b) machen.

Fortsetzung nächste Seite!

Teilaufgabe 4

- a) Das folgende Diagramm zeigt die möglichen Arbeitszustände eines Prozesses nebst Übergängen:



Beschriften Sie die Übergänge anhand folgender Liste von Diensten ( $p$  = Prozeß,  $e$  = Ereignis).

- (1) erzeugen( $p$ )
  - (2) löschen( $p$ )
  - (3) anhalten\_bis\_Ereignis( $p, e$ )
  - (4) melden\_Ereignis( $p, e$ )
  - (5) Prozessorzuteilung( $p$ )
  - (6) Prozessorentzug( $p$ )
- b) Nennen Sie - über den Arbeitszustand hinaus - 5 weitere Bestandteile der Zustandsinformation (Status) eines Prozesses.

Teilaufgabe 5

Welche programmiersprachlichen Konstrukte zur Beschreibung von Synchronisationsproblemen kennen Sie für die Systemprogrammierung?

Beschreiben Sie diese, beginnend mit den hardwarenahen Sprachmitteln.

Teilaufgabe 6

- a) Wodurch sind Transaktionen charakterisiert?
- b) Mit welchen Techniken können Transaktionen realisiert werden?

Fortsetzung nächste Seite!

Teilaufgabe 7

- a) Skizzieren Sie die wesentlichen Charakteristiken der Rechnerarchitektur, die als *von-Neumann-Rechner* bekannt ist.
- b) Was versteht man unter dem von Neumann'schen Flaschenhals?
- c) Unter Speicherbesuchszahl versteht man die Anzahl der Arbeitsspeicherzugriffe pro Maschinenbefehl.  
Wovon ist die Speicherbesuchszahl abhängig?  
Was ist ein typischer mittlerer Wert?  
Gibt es eine obere Schranke?
- d) Welche architekturellen Erweiterungen der von Neumann-Architektur kennen Sie, die den Flaschenhalseffekt abmildern?

Teilaufgabe 8

- a) Für welches konzeptuelle Datenbank-Schema sind Normalformen von Bedeutung?  
Wozu dienen sie?
- b) Welche Normalformen kennen Sie?  
Wie überführt man eine Normalform in eine andere?
- c) Aus welchen wesentlichen Komponenten sind im allgemeinen Expertensysteme aufgebaut?

Teilaufgabe 9

- a) Nennen Sie drei wichtige Fehlertypen, die bei der Datenübertragung in Rechnernetzen vorkommen, und geben Sie Fehlerursachen dafür an.
- b) Nennen Sie für die Fehlertypen übliche Erkennungsmethoden und Auswege aus der Fehlersituation (recovery).
- c) *Ethernet* und *Token Ring* sind wichtige Beispiele von lokalen Netzen.  
Kennzeichnen Sie kurz die zugehörigen Medien-Zugriffsverfahren.  
Wie beurteilen Sie die Verfahren unter Last?

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1990**

Prüfungsteilnehmer	Prüfungstermin	Einzelprüfungsnummer
Kennzahl: _____	<b>HERBST</b>	<b>66111</b>
Kennwort: _____	<b>1990</b>	
Arbeitsplatz-Nr.: _____		

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**  
**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)  
**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt  
**Anzahl der gestellten Themen (Aufgaben):** 1  
**Anzahl der Druckseiten dieser Vorlage:** 3

Sämtliche Teilaufgaben sind zu bearbeiten!

Teilaufgabe 1

- 1.1 Beschreiben Sie die Verfahren der Segmentadressierung und der Seitenadressierung! Diskutieren Sie die durch diese Verfahren entstehenden Vorteile!
- 1.2 Skizzieren Sie die Organisation des Adreßraums bei der kombinierten Segment-Seiten-Adressierung!  
Beschreiben Sie, wie der Zugriff auf eine Adresse erfolgt!
- 1.3 Beschreiben Sie das *Working Set* – Modell!
- 1.4 Beschreiben Sie 4 Ihnen bekannte Seiten-Verdrängungsstrategien und stellen Sie diese bewertend gegenüber!

Fortsetzung nächste Seite!



Teilaufgabe 2

- 2.1.1 Datenorganisation ist auch mit algorithmischen Sprachen (Programmiersprachen) möglich. Nennen Sie Gründe, die für die Verwendung von Datenbanken sprechen!
- 2.1.2 Erklären Sie den Begriff der Integritätsbedingung anhand eines selbstgewählten Beispiels!
- 2.1.3 Warum ist Datenredundanz ein Problem in Datenbanken?
- 2.2.1 Mit welchen 3 grundsätzlichen Operationen (Relationsoperationen) arbeiten Abfragesprachen von relationalen Datenbanken, um Information aus der Datenbank zu erhalten? Welche Funktionalität haben diese jeweils?
- 2.2.2 Betrachten Sie die Komplexitäten dieser Relationsoperationen! Berücksichtigen Sie auch eventuelle Fallunterscheidungen!
- 2.3.1 Was ist der Sinn von Normalformen?
- 2.3.2 Welche Normalformen kennen Sie? Definieren Sie diese!  
Geben Sie jeweils ein Beispiel an, das die jeweilige Normalform erfüllt, höhere Normalformen aber nicht!
- 2.4 Gegeben sind folgende Relationen:  
 $\text{Teile} \subseteq \text{Tname} \times \text{Tfarbe} \times \text{Tgewicht} \times \text{T\#}$   
 $\text{Lieferung} \subseteq \text{L\#} \times \text{T\#} \times \text{Anzahl}$   
 $\text{Lieferant} \subseteq \text{L\#} \times \text{Lname} \times \text{Status} \times \text{Stadt}$   
Gesucht ist eine SQL-Abfrage, die folgendes erfüllt:  
*„Lieferanten mit Status, die Welle mit Teilenummer 5 liefern“.*

Teilaufgabe 3

- 3.1 Geben Sie die allgemeine Architektur eines Expertensystems an und beschreiben Sie die Aufgaben der einzelnen Komponenten !
- 3.2.1 Erläutern Sie die beiden grundlegenden Problemlösungsverfahren **Breitensuche** und **Tiefensuche** ! - Stellen Sie Vor- und Nachteile des jeweiligen Verfahrens gegenüber !
- 3.2.2 Wählen Sie ein geeignetes Beispiel und erklären Sie daran die beiden Verfahren!
- 3.3 Erklären Sie den Mechanismus der Unifikation in Prolog allgemein und anhand eines selbstgewählten Beispiels !

Teilaufgabe 4

- 4.1 Erläutern Sie kurz die Aufgaben der Schicht 3 des OSI-Referenzmodells!
- 4.2 Beschreiben Sie kurz die verschiedenen Schicht-3-Protokolle!
- 4.3 Nennen Sie die gängigsten Vermittlungsverfahren und erläutern Sie deren Wirkungsweise!
- 4.4 Geben Sie Beispiele für Netze an, die mit den genannten Vermittlungsverfahren arbeiten!
- 4.5 Diskutieren Sie Vor- und Nachteile der Verfahren und leiten Sie daraus sinnvolle Anwendungen für sie ab!
- 4.6 Erklären Sie die Begriffe "virtuelle Verbindung" und "Datagramm"!

Teilaufgabe 5

- 5.1.1 Beschreiben Sie die wesentlichen Prinzipien der von-Neumann-Maschine (architekturell, operationell)!
- 5.1.2 Geben Sie einen einfachen Leitwerkzyklus für die von-Neumann-Maschine an!
- 5.2 Was versteht man unter "Mikroprogrammierung"?  
Erläutern Sie den Unterschied zwischen horizontaler und vertikaler Mikroprogrammierung!  
Welche Alternative zur Mikroprogrammierung gibt es?
- 5.3 Diskutieren Sie zwei Erweiterungen der von-Neumann-Prinzipien, die zu einer Leistungssteigerung beitragen!

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1991**

<b>Prüfungsteilnehmer</b>	<b>Prüfungstermin</b>	<b>Einzelprüfungsnummer</b>
<b>Kennzahl:</b> _____	<b>HERBST</b> <b>1991</b>	<b>66111</b>
<b>Kennwort:</b> _____		
<b>Arbeitsplatz-Nr.:</b> _____		

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**  
**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)  
**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt  
**Anzahl der gestellten Themen (Aufgaben):** 1  
**Anzahl der Druckseiten dieser Vorlage:** 4

**Bitte wenden!**

Sämtliche Teilaufgaben sind zu bearbeiten!

### Aufgabe 1 (Virtueller Speicher)

In einem 'Demand-Paging'-System sei die folgende Seitenreferenzfolge (von links nach rechts gelesen) zu bearbeiten:

0 1 2 3 0 1 4 0 1 2 3 4

- 1.1 Ermitteln Sie, um wieviel Prozent die Zahl der Einlagerung bei Verwendung der Ersetzungsstrategie LRU (least recently used) höher liegt als bei einer optimalen Ersetzungsstrategie, und zwar bei Verwendung von
  - a) drei Kacheln,
  - b) vier Kacheln.
- 1.2 Führen Sie die gleiche Untersuchung für die Strategie 'Second Chance' durch!

### Aufgabe 2 (Prozessorvergabestrategien)

Gegeben sei ein System mit einer Bedienstation, die eine Reihe unabhängiger Aufträge zu bearbeiten hat, deren Bedienzeiten bekannt sind.

- 2.1 Zeigen Sie, daß die Abarbeitung nach steigender Bedienzeitanforderung (shortest job first) die mittlere Verweilzeit minimiert, wenn sämtliche zu bearbeitenden Aufträge bereits beim Start des Systems vorhanden sind!
- 2.2 Wie müßte die Zuordnungsstrategie geändert werden, damit (eventuell unter

Fortsetzung nächste Seite!

einschränkenden Annahmen) die mittlere Verweilzeit auch dann minimiert wird, wenn nicht alle Aufträge bereits beim Systemstart vorhanden sind, sondern teilweise erst danach in das System gelangen? Falls Sie einschränkende Annahmen machen, sind sie kurz zu diskutieren.

### Aufgabe 3 (Betriebsmittelverwaltung)

Ein Betriebsmittelsystem bestehe aus  $n$  Prozessen (numeriert von 0 bis  $n-1$ ) und  $m$  wiederverwendbaren Betriebsmitteltypen (numeriert von 0 bis  $m-1$ ). Zur Darstellung der Betriebsmittelgraphen sollen folgende Datenstrukturen verwendet werden:

- ein Vektor  $v$  mit  $m$  Komponenten, der im jeweiligen Zustand angibt, wieviele Exemplare jedes Betriebsmitteltyps noch frei verfügbar sind,
- eine  $(n, m)$ -Matrix, die im jeweiligen Zustand angibt, wieviele Exemplare eines jeden Betriebsmittels den einzelnen Prozessen zugeordnet sind,
- eine  $(n, m)$ -Matrix, die im jeweiligen Zustand angibt, wieviele Exemplare eines jeden Betriebsmittels von den einzelnen Prozessen angefordert, aber noch nicht zugeteilt sind, und
- einen Vektor  $ps$  von  $n$  Semaphoren, dessen  $i$ -te Komponente als privater Semaphor dem  $i$ -ten Prozeß (zu seiner Blockierung bzw. Deblockierung) zugeordnet ist.

Das Betriebssystem stelle folgende Prozeduren zur Verfügung:

- P- und V-Operationen für Semaphore und
- die Funktion 'this\_process', die als Ergebnis die Nummer des aufrufenden Prozesses liefert.

Formulieren Sie unter diesen Voraussetzungen in einer geeigneten Programmiersprache folgende von den Prozessen gemeinsam benutzbaren Prozeduren (Beachten Sie, daß die obigen Datenstrukturen von den Prozessen gemeinsam benutzt werden!):

1. Eine Prozedur 'anfordern', die von den Prozessen aufgerufen werden kann und als Parameter den Vektor der angeforderten Betriebsmittel besitzt. Falls die Anforderungen insgesamt erfüllbar sind, soll die entsprechende Zuteilung vorgenommen und zum aufrufenden Programm zurückgekehrt werden. Anderm-

Fortsetzung nächste Seite!

- falls sind die Anforderungen vorzumerken und der aufrufende Prozeß zu blockieren.
2. Eine Prozedur 'freigeben', die von den Prozessen aufgerufen werden kann, und als Parameter den Vektor der freizugebenden Betriebsmittel besitzt. Ist für einen Betriebsmitteltyp die Angabe der freizugebenden Exemplare größer als die Zahl der an den aufrufenden Prozeß zugeteilten, so werden lediglich die zugeteilten Exemplare 'freigegeben'. Sind nach der Freigabe blockierte Prozesse vorhanden, deren Anforderungen nun abgedeckt werden können, so sollen (nach einer beliebigen Strategie) so lange Prozesse deblockiert werden, bis weitere Deblockierungen nicht mehr möglich sind.
  3. Eine Prozedur 'prüfen', die feststellt, ob partielle Verklemmungen vorliegen (Berücksichtigen Sie, daß diese Prozedur die Daten, auf denen die Prozeduren 'freigeben' und 'anfordern' operieren, nicht modifizieren darf!).
  4. Geben Sie die Anfangsbesetzung aller benutzten Semaphore an!

#### Aufgabe 4 (Relationale Datenbanken)

- 4.1 Welche Eigenschaften muß eine Relation besitzen, damit sie die erste bzw. zweite bzw. dritte Normalform erfüllt?
- 4.2 Das Bundesamt für Kfz-Wesen habe in einem ersten Entwurf folgende unnormalisierte Relationen aufgestellt, um Verkehrssünder zu registrieren.

EIGENTÜMER(PKZ, NAME, ADR, AUTO)

AUTO(KFZNR, TYP, FARBE, EINTRAG)

EINTRAG(LFDNR, DATUM, ART, STRAFPUNKTE, FAHRER)

FAHRER(PKZ, NAME, ADR, FÜHRERSCHEIN, FÜHRERSCHEINENTZUG)

FÜHRERSCHEIN(FNR, ADATUM, KLASSE, AUSSTELLUNGSSORT)

Die Primärschlüssel sind durch Unterstreichung kenntlich gemacht.

Man normalisiere diese Relationen, und überführe sie in die dritte Normalform, wobei folgende Semantik zu unterstellen ist:

- Jedes Auto kann nur einen Eigentümer, ein Eigentümer aber mehrere Autos haben.
- Ein Auto und ein Fahrer sind jeweils durch einen Eintrag betroffen. Pro Auto und pro Fahrer können mehrere verschiedene Einträge existieren, d. h. ein Fahrer kann sich Einträge mit verschiedenen Autos einhandeln. Die LFDNR wird in Bezug auf Auto geführt.
- Pro Fahrer sind verschiedene Führerscheine (auch Ersatzführerscheine) registriert.

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Frühjahr 1993**



---

Prüfungsteilnehmer

Prüfungstermin

Einzelprüfungsnummer

---

Kennzahl: \_\_\_\_\_

**FRÜHJAHR**

**66111**

Kennwort: \_\_\_\_\_

**1993**

Arbeitsplatz-Nr.: \_\_\_\_\_

---

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 3

**Sämtliche Teilaufgaben sind zu bearbeiten!**

**Aufgabe 1: Virtuelle Speicherverwaltung**

- a) Als Verfahren zur Zuordnung von virtuellen Adreßräumen zum realen Speicher sind Ihnen Segmentierung und Paging bekannt. Was haben diese Verfahren gemeinsam, und wo liegen die Unterschiede?
- b) Bei einem System mit Paging habe eine Seite die Länge  $k$  (Wörter). Die durchschnittliche Segmentlänge sei  $L$ . Für den Eintrag in die Seiten-Kachel-Tabelle sei jeweils ein Wort pro Seite erforderlich. Wie ist  $k$  zu wählen, um bei einer durchschnittlichen Segmentlänge  $L$  den durchschnittlichen Speicherverschnitt pro Segment zu minimieren?
- c) Wie kommt es beim Paging zum Seitenflattern, und was kann man dagegen tun?

**Aufgabe 2 : Synchronisationssysteme**

Versetzen wir uns für diese Aufgabe in den Alltag einer Familie mit drei kleinen Kindern, die auch im Winter draußen spielen wollen. Damit sie sich nicht erkälten, braucht jedes Kind eine Mütze, ein Paar Handschuhe und einen Schal. Leider ist ihre Oma mit dem Stricken noch nicht so weit, so daß jedes Kind im Moment erst eines dieser drei Kleidungsstücke besitzt und zwar jedes ein anderes. Außerdem haben sie im Kleiderschrank noch eine alte Mütze, ein altes Paar Handschuhe und einen großen Schal gefunden.

Fortsetzung nächste Seite!

Wenn nun ein Kind nach draußen will, braucht es noch zwei von den Sachen aus dem Schrank:

- entweder Schal und Mütze
- oder Mütze und Handschuhe
- oder Handschuhe und Schal

a) Geben Sie eine Lösung dieses Problems an unter Verwendung von

- P/V-Semaphoren
- P/V<sub>multiple</sub>-Semaphoren
- einem Petrinetz

b) Diskutieren Sie, ob sich das von Ihnen angegebene Prozeßsystem verklemmen kann!

(Welches allgemeine Verfahren verwenden Sie dabei zur Verhinderung von Verklemmungen)

### Aufgabe 3 : Banker-Algorithmus

Gegeben sei ein System aus drei Prozessen  $P_1, P_2, P_3$  und Betriebsmitteln  $B_1, B_2, B_3, B_4$ . Die Anzahl der vorhandenen Betriebsmittel ist festgelegt durch den Vektor:

$$V = (6, 7, 6, 4).$$

Die maximal benötigten Betriebsmittel der drei Prozesse sind gegeben durch die Matrix:

$$B = \begin{bmatrix} 1 & 5 & 3 & 2 \\ 2 & 3 & 3 & 1 \\ 4 & 5 & 2 & 4 \end{bmatrix}$$

Die folgenden Matrizen ergeben jeweils einen Status der zugeteilten Betriebsmittel:

$$Q_1 = \begin{bmatrix} 0 & 1 & 3 & 0 \\ 1 & 3 & 1 & 0 \\ 3 & 2 & 0 & 1 \end{bmatrix}$$

$$Q_2 = \begin{bmatrix} 1 & 4 & 1 & 0 \\ 1 & 3 & 3 & 1 \\ 4 & 0 & 2 & 4 \end{bmatrix}$$

$$Q_3 = \begin{bmatrix} 1 & 0 & 1 & 2 \\ 2 & 3 & 1 & 1 \\ 1 & 3 & 1 & 1 \end{bmatrix}$$

- a) Sind die Zustände  $Q_1, Q_2$  und  $Q_3$  realisierbar?
- b) Existieren für  $Q_1, Q_2$  und  $Q_3$  sichere Sequenzen?
- c) Es sei nun  $V = (6, 7, 6, 5)$ , d.h. von Betriebsmittel  $B_4$  steht ein Exemplar mehr zur Verfügung. Wie ändern sich dadurch die Antworten auf die Teilaufgaben a) und b)?

Fortsetzung nächste Seite!

**Aufgaben 4: (Datenbanken)****4.1 Transaktionen**

Welche vier grundlegenden Eigenschaften kennzeichnen eine Transaktion und was bedeuten diese Eigenschaften?

**4.2 SQL**

Gegeben seien die folgenden Relationen:

PERSONAL (PNR NAME GEHALT BERUF ANR)  
ABTEILUNG (ANR NAME ORT)

mit PNR Personalnummer und ANR Abteilungsnummer.

Folgende Operationen sind in SQL zu formulieren:

- a) Gib mir die Namen der Angestellten, die in der Abteilung "Software" arbeiten!
- b) Bestimme das Durchschnittsgehalt der Angestellten aus Abteilung "4711"!
- c) Gib mir das höchste Gehalt aus jeder Berufsgruppe!
- d) Referentielle Integritätsbedingungen: Jeder Angestellte muß in einer Abteilung arbeiten! (keine genaue Syntax)

**4.3 Codasyl**

- a) Geben Sie für obige beiden Relationen das Bachman-Diagramm für den Codasyl-Set an!
- b) Erstellen Sie ein Ausprägungsdiagramm für folgende Einträge: "In der Abteilung 4711 arbeiten die Angestellten Müller und Meier."

**4.4 Normalformenlehre**

Führen Sie nachfolgende Relation in die dritte Normalform über:

STUDENT (MATRNR NAME GEB ADR FBR FBNAME DEKAN)

9516570 HUBER 010148 XX 11 ING-WISS WALTER  
9517058 MAIER 210849 YY 11 ING-WISS WALTER  
9110457 BAUER 130548 XX 3 MEDIZIN HILBERG

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1993**

Prüfungsteilnehmer	Prüfungstermin	Einzelprüfungsnummer
Kennzahl: _____	<b>HERBST</b> <b>1993</b>	<b>66111</b>
Kennwort: _____		
Arbeitsplatz-Nr.: _____		

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**  
- Prüfungsaufgaben -

**Fach:** Informatik (vertieft studiert)  
**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt  
**Anzahl der gestellten Themen (Aufgaben):** 1  
**Anzahl der Druckseiten dieser Vorlage:** 4

Sämtliche Teilaufgaben sind zu bearbeiten!

**Aufgabe 1:**

Betrachten Sie für die folgenden Teilaufgaben eine von einem Polizisten geregelte Straßenkreuzung, für die gelten soll:

- Es kreuzen sich zwei Einbahnstraßen. Die Autos wollen die Kreuzung in Nord-Süd Richtung und Ost-West Richtung überqueren (Abbiegen ist nicht möglich).
- Es kann sich immer nur ein Auto in der Kreuzung befinden.  
Vor der Kreuzung kann sich ein Stau bilden (nach der Kreuzung nicht).
- Der Polizist regelt den Verkehr fair und möglichst optimal. Er kann die Durchfahrt (wie an Kreuzungen üblich) in Nord-Süd oder Ost-West Richtung freigeben und die andere Richtung jeweils sperren.  
Er wechselt die Richtung spätestens nach 20 sec, schaltet aber dann schon eher um, wenn in der freien Richtung keine Autos mehr kommen, in der gesperrten Richtung aber welche warten.

Fortsetzung nächste Seite!

- Beim Umschalten der Richtung hebt er zuerst den Arm, worauf keiner mehr in die Kreuzung einfahren darf. Erst wenn sich kein Auto mehr in der Kreuzung befindet, schaltet er um.

Bemerkung für die folgenden Teilaufgaben: Man kann nicht damit rechnen, daß sich Autofahrer immer an die Verkehrsregeln halten. Aus Sicherheitsgründen sollte deshalb das Modul "Straßenkreuzung" (soweit möglich) die Operationenfolge der durchfahrenden Autos an der Kreuzung festlegen.

### Aufgabe 1.1:

Entwerfen Sie eine formale Spezifikation eines Moduls "Straßenkreuzung" für das beschriebene Problem! Das Modul soll dabei die folgenden Operationen und Prozesse beinhalten.

a) die Operationen des Moduls Kreuzung:

*an\_kreuzung\_heranhfahren* (auto: wagen, richt: richtung)

*in\_kreuzung\_einfahren* (richt: richtung)

*aus\_kreuzung\_herausfahren* ( )

*über\_kreuzung\_fahren* (auto: wagen, richt: richtung) /\* diese Operation ruft die drei anderen Operationen in der richtigen Reihenfolge auf \*/

b) den Prozeß

*polizist*

c) Zusätzlich enthält das Modul die Operation Zeittakt, die jede Sekunde von einem Uhrmodul aufgerufen wird.

*zeit\_takt* ( )

EFFECTS: zeit = 'zeit + 1

Anmerkung zur Notation: In Prädikaten, die den Effekt einer Operation beschreiben, wird der Variablenname als Bezeichner des Wertes verwendet, den die Variable nach Ausführung der Operation hat, und der Variablenname mit vorgestelltem Apostroph für den Wert unmittelbar vor Ausführung der Operation.

d) Es seien die folgenden Typen bekannt:

TYPES richtung = (ost\_west, nord\_süd)

wagen = autonummer;

queue /\* oder sequence, mit den üblichen Operationen

first, tail und add, remove \*/

END\_TYPES

**Aufgabe 1.2:**

Implementieren Sie für das eingangs beschriebene Problem ein Modul "Straßenkreuzung" mit den in Aufgabe 1.1 genannten Operationen und Prozessen! Ihre Implementierung soll insbesondere die auftretenden Synchronisationsprobleme lösen.

Sie können selbst festlegen, welches Synchronisationsmittel Sie verwenden wollen (z.B. Semaphore mit den Funktionen  $p$  und  $v$ ). Begründen Sie Ihre Wahl!

(Sie können nach Bedarf auch zusätzliche Hilfsfunktionen und/oder Prozesse einführen.)

**Aufgabe 2: (Warteschlangen)**

In der Mensa existiere nur eine Kasse. Da sich an der einen Kasse regelmäßig längere Wartezeiten ergeben, werden zur Beschleunigung der Abfertigung die 3 folgenden Lösungen diskutiert:

- a) Es wird eine zweite Kasse installiert. Die Studenten stellen sich in einer Warteschlange an und der jeweils erste wartende Student benutzt die erste freiwerdende Kasse.
- b) Durch organisatorische Maßnahmen kann erreicht werden, daß die Abfertigung an der einen Kasse doppelt so schnell erfolgt.
- c) Es werden zwei getrennte Warteschlangen mit zwei Kassen mit der ursprünglichen Abfertigungsgeschwindigkeit installiert. Die Studenten verteilen sich gleichmäßig auf die beiden Warteschlangen.

Sowohl die Ankunftsintervalle der Studenten (Ankunftsrate  $\lambda$ ) als auch die Bedienzeiten an der Kasse (Bedienrate  $\mu$ ) sollen exponentiell verteilt sein.

**Aufgabe 2.1:**

Entwerfen Sie zu jedem der gemachten Vorschläge ein Warteschlangenmodell!

**Aufgabe 2.2:**

Bestimmen Sie für die einzelnen Vorschläge die mittlere Verweilzeit im System!

Welches ist somit das beste System (mit der kürzesten Verweilzeit im System)?

Hinweis: Für ein System aus  $k$  Bedienstationen mit exponentiell verteilter Ankunfts- und Bedienrate ( $M/M/k$ ) gilt für den Erwartungswert der Anzahl von Aufträgen im System:

$$E[N] = k\rho + p_0 \frac{(\rho(k\rho)^k)}{k!(1-\rho)^2} \quad \text{mit } p_0 = \frac{1}{\frac{(k\rho)^k}{k!(1-\rho)} + \sum_{i=0}^{k-1} \frac{(k\rho)^i}{i!}} \quad \text{und } \rho = \frac{\lambda}{k\mu}$$

und es gilt der Satz von Little:

mittlere Verweilzeit =  $E[N]$  / mittlere Anzahl ankommender Aufträge pro Zeiteinheit.

**Aufgabe 2.3:**

Wie verhalten sich die 3 Vorschläge bezüglich der Ausfallsicherheit?



**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Frühjahr 1994**

---

**Prüfungsteilnehmer**

**Prüfungstermin**

**Einzelprüfungsnummer**

---

**Kennzahl:** \_\_\_\_\_

**FRÜHJAHR**

**66111**

**Kennwort:** \_\_\_\_\_

**1994**

**Arbeitsplatz-Nr.:** \_\_\_\_\_

---

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 5

bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

Aufgabe 1 :

Erläutern Sie den Begriff des Deadlock anhand eines Beispiels

Aufgabe 2 :

Gegeben sei ein Speicherbereich mit 1000 freien Speicherplätzen, welcher bei Adresse 10000 beginnt. Es werden freie Speicherbereiche wie folgt angefordert und zurückgegeben:

- a.) Zeitpunkt 1: Anforderung 500 Plätze
- b.) Zeitpunkt 2: Anforderung 300 Plätze
- c.) Zeitpunkt 3: Anforderung 100 Plätze
- d.) Zeitpunkt 4: Rückgabe von a.)
- e.) Zeitpunkt 5: Anforderung 70 Plätze
- f.) Zeitpunkt 6: Anforderung 430 Plätze
- g.) Zeitpunkt 7: Anforderung 100 Plätze

Wie sieht die Speicherbelegung zu jedem Zeitpunkt aus, falls die Speichervergabe nach der

- a. First-Fit-Methode
- b. Best-Fit-Methode

erfolgt ? Sind alle Speicheraanforderungen jeweils zu erfüllen ? Es ist davon auszugehen, daß freie Speicherbereiche in einer Liste verkettet werden und ein zurückgegebener Block jeweils am Anfang dieser Liste eingekettet wird.

Aufgabe 3 :

Erläutern Sie den Begriff des Von-Neumann-Rechners

**Aufgabe 4 :**

Gegeben seien die folgenden Attribute für Bücher, Entleiher und Autoren:

Signatur	Signatur eines Buches
Titel	Buchtitel
Gebiet	Fachgebiet des Buches
Art	Art des Buches. z.B. Einführungsbuch o.ä.
ErschOrt	Erscheinungsort
Matr	Matrikelnummer eines Entleihers
StudName	Name des Entleihers
StudGeb	Geburtsdatum des Entleihers
Sem	Semester des Entleihers
StudWohnort	Wohnort des Entleihers
StudFachr	Fachrichtung des Entleihers
Aut.Nr	Nummer eines Autors
Aut.Name	Name eines Autors
AutWohnort	Wohnort eines Autors
AutBuchHonorar	Honorar, welches ein Autor für ein Buch bekommt

Studenten können mehrere Bücher entliehen haben. Ein Autor kann mehrere Bücher geschrieben haben. Ein Buch kann mehrere Autoren haben, in diesem Fall bekommt jeder Autor ein eigenes autor- und buchspezifisches Honorar.

Welche funktionalen Abhängigkeiten gelten unter den obigen Attributen ?

**Aufgabe 5 :**

Man betrachte das relationale Schema

R(Signatur, Titel, Gebiet, Art, ErschOrt, Matr, StudName, StudGeb, Sem,  
StudWohnort, StudFachr, AutNr, AutName, AutWohnort, AutBuchHonorar)

und die funktionalen Abhängigkeiten aus Aufgabe 4.

Ist die Dekomposition

R1(Mat, StudName, StudGeb, Sem, StudWohnort, StudFachr).

R2(Signatur, Titel, Gebiet, Art, ErschOrt).

R3(AutNr, AutName, AutWohnort).

R4(Signatur, AutNr, AutBuchHonorar)

verlustfrei, d.h. hat sie die lossless-join-Eigenschaft ? Man begründe die Antwort, z.B. durch Verwendung eines geeigneten Algorithmus.

**Aufgabe 6 :**

Man betrachte das relationale Schema

R(Signatur, Titel, Gebiet, Art, ErschOrt, Matr, StudName, StudGeb, Sem,  
StudWohnort, StudFachr, AutNr, AutName, AutWohnort, AutBuchHonorar)

und die funktionalen Abhängigkeiten aus Aufgabe 4.

Ist die Dekomposition

R1(Mat, StudName, StudGeb, Sem, StudWohnort, StudFachr),

R2(Signatur, Titel, Gebiet, Art, ErschOrt),

R3(AutNr, AutName, AutWohnort),

R4(Signatur, AutBuchHonorar)

abhängigkeitserhaltend ? Man begründe die Antwort. z.B. durch Verwendung eines geeigneten Algorithmus.

**Aufgabe 7 :**

Man betrachte das relationale Schema

R(Signatur, Titel, Gebiet, Art, ErschOrt, Matr, StudName, StudGeb, Sem,  
StudWohnort, StudFachr, AutNr, AutName, AutWohnort, AutBuchHonorar)

und die funktionalen Abhängigkeiten aus Aufgabe 4.

Man gebe eine abhängigkeitserhaltende und verlustfreie (lossless join) Dekomposition von R in 3. Normalform an. Man verwende einen geeigneten Algorithmus.

**Aufgabe 8 :**

Man betrachte das relationale Schema

R(Signatur, Titel, Gebiet, Art, ErschOrt, Matr, StudName, StudGeb, Sem,  
StudWohnort, StudFachr, AutNr, AutName, AutWohnort, AutBuchHonorar)

und die funktionalen Abhängigkeiten aus Aufgabe 4.

Man gebe eine Dekomposition von R in Boyce-Codd-Normalform an. Man verwende einen geeigneten Algorithmus. Ist die Dekomposition abhängigkeitserhaltend und verlustfrei (lossless join) ?

**Aufgabe 9 :**

Für die Attribute aus Aufgabe 4 gebe man SQL CREATE Anweisungen für ein sinnvolles Schema an. z.B. für die Dekomposition aus Aufgabe 7 oder aus Aufgabe 8.

Aufgabe 10 :

Unter Verwendung des Schemas aus Aufgabe 9 gebe man SQL-Anweisungen und Ausdrücke in der relationalen Algebra für die folgenden Anfragen an:

- a. Gesucht ist der Name des Studenten mit der Matrikelnummer 123456.
- b. Gesucht sind die Signaturen der Bücher, die der Student mit der Matrikelnummer 654321 ausgeliehen hat.
- c. Gesucht sind die Signaturen der Bücher, die der Student namens Albrecht ausgeliehen hat.
- d. Gesucht sind die Autornamen der Bücher, die der Student namens 'Albrecht' ausgeliehen hat.

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1994**

---

**Prüfungsteilnehmer**

**Prüfungstermin**

**Einzelprüfungsnummer**

---

**Kennzahl:** \_\_\_\_\_

**Herbst**

**66111**

**Kennwort:** \_\_\_\_\_

**1994**

**Arbeitsplatz-Nr.:** \_\_\_\_\_

---

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechn.architekt

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 2

Bitte wenden!



Sämtliche Aufgaben sind zu bearbeiten!

### Aufgabe 1: Verklemmungen:

- 1.1) Wie lauten die 4 notwendigen Bedingungen für Verklemmungen?
- 1.2) Wie können Verklemmungen behoben werden (Kurze Erläuterung)?
- 1.3) Gegeben sei ein Prozeßsystem mit drei Prozessen.

Prozeß 1 erzeugt Betriebsmittel vom Typ 2.

Die Betriebsmittel vom Typ 1 und 3 sind wiederverwendbar.

Anfangs sind an Betriebsmitteln vorhanden:

BM1: 5

BM2: 3

BM3: 3

Die drei Prozesse haben folgende Operationen bereits ausgeführt:

Anforderung  $P_2$ : {(BM1, 1)};

Anforderung  $P_3$ : {(BM3, 3)};

Zuteilung ( $P_2$ );

Zuteilung ( $P_3$ );

Anforderung  $P_1$ : {(BM1, 2), (BM3, 1)};

Anforderung  $P_2$ : {(BM3, 1)};

Anforderung  $P_3$ : {(BM2, 4)};

Stellen Sie den augenblicklichen Zustand als Betriebsmittelgraph dar!

- 1.4) Liegt eine Verklemmung vor? Kurze Begründung!

### Aufgabe 2: Arbeitsspeicherverwaltung

- 2.1) Beschreiben Sie den Seitenaustauschalgorithmus SC (second chance):
- 2.2) Was enthält ein Segmentdeskriptor bei einem System, das nur Segmentierung und keine Seitenadressierung kennt und keinen Cache hat?
- 2.3) Was versteht man unter Seitenflattern (Thrashing) und was sind die Folgen?
- 2.4) Welche zusätzlichen Prozeßzustände sind notwendig, um das Problem Thrashing zu lösen. Zeichnen Sie das erweiterte Prozeß-Zustandsdiagramm.

### Aufgabe 3: Periphere Geräte

Beschreiben Sie kurz die Funktionsweise sowie Vor- und Nachteile der folgenden Anschlußschemata für periphere Geräte?

- 3.1) Integrierte Ausführung
- 3.2) Abgesetzte Ausführung
- 3.3) Selbständige Ausführung

### Aufgabe 4: Prozeßsystem / Synchronisation

- 4.1) Implementieren Sie das Leser-Schreiber-Problem als Prozeßsystem  $P$ , mit folgenden Eigenschaften:  
Es gibt eine beschränkte Zahl von  $k$  Lesern, die alle gleichzeitig lesen dürfen. Es gibt eine beliebige Zahl von Schreibern. Ein Schreiber braucht exklusiven Zugriff zu den Daten. Sie können frei wählen, welches Synchronisationssystem Sie verwenden wollen, ebenso, ob die Schreiber Vorrang haben sollen oder nicht. (bitte aber angeben)  
Ihre Lösung darf sich nicht verklemmen und darf kein aktives Warten (busy wait) aufweisen.
- 4.2) Beschreiben Sie das von Ihnen verwendete Synchronisationssystem.

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Frühjahr 1995**

Kennzahl: \_\_\_\_\_

**Frühjahr**

**66111**

Kennwort: \_\_\_\_\_

**1995**

Arbeitsplatz-Nr.: \_\_\_\_\_

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechnerarch.

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 3

Bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

### 1. Aufgabe:

- a) Drücken Sie durch eine Formel aus, wieviel Zeit benötigt wird, um ein Programm auf einem Rechner in Abhängigkeit der folgenden Größen abzuarbeiten:
- /      mittlere Zykluslänge  
CPI    Anzahl der Zyklen pro Maschineninstruktion (Cycles Per Instruction)  
IC      mittlere Befehlszahl (Instruction Count).
- b) Welche Möglichkeiten gibt es für den Rechnerarchitekten einen Rechner zu entwickeln, der das Programm möglichst schnell abarbeitet? Welche dieser Möglichkeiten sind eher technologie-, welche eher architekturabhängig? Welche dieser Möglichkeiten lassen sich nur schlecht gleichzeitig verwirklichen?
- c) Für einen Mix aus Benchmarks wurden folgende Anteile verschiedener Instruktionsarten ermittelt:

Instruktion	Anteil	Rechner 1	Rechner 2
Integer-Operation	51 %	2	1
Load/Store	30 %	4	1
Branch	17 %	2	1
Jump	2 %	2	1

Die beiden letzten Spalten geben die Anzahl der Zyklen pro Instruktion für Rechner 1 bzw. Rechner 2 an. Die Taktfrequenz von Rechner 1 beträgt 20 MHz und die Taktfrequenz von Rechner 2 beträgt 8 MHz. Welcher Rechner ist schneller?

### 2. Aufgabe:

Betrachten Sie eine typische 6-stufige Befehlspipeline!

- a) Welche Funktion haben die einzelnen Pipelinestufen?
- b) Wie groß ist die Beschleunigung (Speed-up), die man erhält, wenn der Instruktionsstrom aus  $n$  Maschineninstruktionen besteht und keine Pipeline-Hemmnisse auftreten?
- c) Welche Beschleunigung erhält man für einen Instruktionsstrom der Länge  $n$  im Idealfall, wenn die Prozessorarchitektur  $m$  derartige Pipelines enthält, die nebenläufig betrieben werden können (Superskalaprozessor)?
- d) Nennen Sie (wenigstens) zwei Eigenschaften von Instruktionsströmen, die verhindern, daß dieser Idealfall eintreten kann! Erläutern Sie die Auswirkung dieser Eigenschaften anhand von Pipeline-Phasendiagrammen!
- e) Skizzieren Sie eine 4-stufige(arithmetische) Pipeline für die Gleitpunkt-Addition, und erläutern Sie die Funktion der einzelnen Stufen!

Fortsetzung nächste Seite!

**3. Aufgabe:**

- a) Erläutern Sie die Begriffe CISC und RISC! Stellen Sie dazu jeweils charakteristische Eigenschaften gegenüber!
- b) Um die Befehlspipeline in einem Prozessor nicht zu bremsen, arbeiten manche RISC Prozessoren mit sogenannten *delayed branches*, d.h. der hinter einem Sprungbefehl stehende Befehl (im sogenannten *delay slot*) wird noch in jedem Fall ausgeführt.

Geben Sie zwei Möglichkeiten zur Füllung des *delay slots* an und begründen Sie, wann sie verwendet werden können und wann dadurch eine Leistungssteigerung erreicht wird!

- c) Nennen Sie neben Sprüngen eine weitere Ursache, die zu Verzögerungen in einer Befehlspipeline führen kann und geben Sie eine Möglichkeit zur Behebung an!
- d) Welche Auswirkungen haben RISC-Architekturen auf Compiler?

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Frühjahr 1996**

---

<b>Prüfungsteilnehmer</b>	<b>Prüfungstermin</b>	<b>Einzelprüfungsnummer</b>
---------------------------	-----------------------	-----------------------------

---

**Kennzahl:** \_\_\_\_\_

**Frühjahr**

**66111**

**Kennwort:** \_\_\_\_\_

**1996**

**Arbeitsplatz-Nr.:** \_\_\_\_\_

---

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**

**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechnerarch.

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 4

Bitte wenden!

**Sämtliche Teilaufgaben sind zu bearbeiten!**

**Aufgabe 1**

Eine Prioritätswarteschlange kann als Heap implementiert werden.

a) Geben Sie die Definition eines Heaps an, und vereinbaren Sie eine entsprechende Datenstruktur!

b) Beschreiben Sie die Operationen

- access (Zugriff auf das Maximum)
- remove (entfernen des Maximums)
- insert (einfügen eines Elements)
- construct (Aufbau eines Heaps aus n Werten)

c) Geben Sie die Komplexität der angegebenen Operationen an, und begründen Sie Ihre Angaben!

**Aufgabe 2**

Eine Universitätsdatenbank soll folgende Daten verwalten.

- Studenten (Name, Matrikelnummer, Geburtsdatum, Adressen, Semesteranzahl, Studiengang, Fakultät, belegte Lehrveranstaltungen und deren Art)
- Lehrveranstaltungen (Anfangszeit, Semester, Art ( Vorlesung, Seminar, Übung, Praktikum) Name, Anzahl der Hörer, Nummer im Vorlesungsverzeichnis, Anzahl Semesterwochenstunden, Dozent, Raum)

a) Entwerfen Sie ein ER-Schema für diese Applikation! Berücksichtigen Sie dabei, daß eine Vorlesung in 2 Unterrichtseinheiten aufgeteilt werden kann (z.B. Mo 10.00 und Do 14.00). Begründen Sie Ihren Entwurf!

b) Spezifizieren Sie für die Entity-Typen Attribute, und zeichnen Sie die Schlüsselattribute aus!

c) Geben Sie ein relationales Datenbankschema an!

Fortsetzung nächste Seite!



## Aufgabe 3

- a) Beschreiben Sie den Aufbau des UNIX-Filesystems!
- b) Welche Aufgabe kommt den Inodes zu?
- c) Was ist ein Link ?
- d) Welche Tabellen werden durchlaufen, wenn durch einen Dateideskriptor auf ein File zugegriffen wird?

## Aufgabe 4

Beschreiben Sie die Auswirkungen, die sich durch Anwenden des Pipeline-Prinzips

- a) bei der Befehlsentschlüsselung
- b) bei der Arithmetik von Vektorrechnern ergeben!

## Aufgabe 5

Charakterisieren Sie

- a) eine Batch Transformation
- b) eine ereignisgesteuerte Benutzeroberfläche
- c) eine Client-Server-Lösung!

## Aufgabe 6

Folgende Tabelle gibt die IC-Fahrzeiten an:

Abfahrt	Ankunft	Zeit
KA	BAD	18 min
KA	MA	25 min
MA	MZ	46 min
MA	F	48 min
MZ	KO	49 min
MZ	F	51 min
F	MZ	51 min
F	KS	1:58 h
F	FD	1:05 h
KO	DO	1:36 h
FD	G	2:19 h
DO	H	1:32 h
DO	HB	1:47 h
H	HB	1:12 h
H	HH	1:14 h
HB	HH	53 min

Fortsetzung nächste Seite!

Bestimmen Sie einen Weg von Karlsruhe nach Hamburg

- (a) mit Tiefensuche
- (b) mit Breitensuche
- (c) mit Hill Climbing Suche
- (d) mit branch and bound Suche.

Vergleichen Sie die Fahrzeiten!

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1996**

---

Prüfungsteilnehmer

Prüfungstermin

Einzelprüfungsnummer

---

Kennzahl: \_\_\_\_\_

**Herbst**

**66111**

Kennwort: \_\_\_\_\_

**1996**

Arbeitsplatz-Nr.: \_\_\_\_\_

---

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**  
**- Prüfungsaufgaben -**

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Betriebs/Datenbanksyst., Rechnerarch.

**Anzahl der gestellten Themen (Aufgaben):** 1

**Anzahl der Druckseiten dieser Vorlage:** 4

Bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

**Aufgabe 1:**

- 1a) Rechensysteme werden nach bestimmten Kriterien klassifiziert, um sie qualitativ und quantitativ analysieren und miteinander vergleichen zu können. Die Klassifikation kann dabei nach Leistung, Anwendungsanforderungen oder Anwendungsgebieten erfolgen. Nennen Sie zwei weit verbreitete Anwendungsgebiete und erläutern Sie deren Besonderheiten im Hinblick auf die Architektur von Rechensystemen.
- 1b) Eine häufig verwendete Taxonomie ist die sogenannte *Flynn'sche Klassifikation*. Erklären Sie diese Taxonomie und nennen Sie die Merkmale der beiden Unterklassen der *Princeton-* und *Harvard-Architektur*. Welche Arten der Parallelität gibt es bei Parallelrechnern?
- 1c) Der Datenaustausch in MIMD-Rechner kann auf zwei Arten erfolgen. Beschreiben Sie die beiden Methoden.

**Aufgabe 2:**

- 2a) Ein Computer besitzt eine drei-stufige Speicherhierarchie aus Cache, Hauptspeicher und Plattenspeicher. Ein Cache-Zugriff erfordert 1 Taktzyklus, ein Zugriff auf den Hauptspeicher 4 Taktzyklen und ein Zugriff auf den Plattenspeicher 1200 Taktzyklen; 0.5 % der Zugriffe erfolgen auf den Plattenspeicher, 30 % auf den Hauptspeicher und die restlichen 69.5 % auf den Cache.  
Wie viele Zyklen werden im Mittel für einen Speicherzugriff benötigt? Wieviele Zugriffe benötigt derselbe Computer im Mittel, wenn er keinen Cache besitzt und stattdessen diese Zugriffe vom Hauptspeicher erledigt werden können?  
Wie wirkt sich die Einführung eines Second-Level-Caches aus, der eine Zugriffszeit von 2 Taktzyklen besitzt und auf den 60 % der Speicherzugriffe erfolgen, die im ursprünglichen System vom Hauptspeicher erledigt wurden; die Wahrscheinlichkeiten für Zugriffe auf den Plattenspeicher und den First-Level-Cache bleiben unverändert.
- 2b) Ein Computer besitzt eine Taktfrequenz von 120 MHz und benötigt für eine Integer-Instruktion im Mittel 6 Takte. Wieviele  $\mu$ s werden im Mittel für eine Instruktion benötigt und wieviele MIPS (million instructions per second) werden im Durchschnitt ausgeführt? Derselbe Rechner benötigt für einen Befehl mit Fließkommaverarbeitung durchschnittlich 18 Takte. Wieviele  $\mu$ s werden im Mittel für eine Fließkommaverarbeitung benötigt und wieviele FLOPS (floating point operations per second) werden im Mittel geleistet?

**Aufgabe 3:**

Ein SISD-Rechner berechnet ein Problem, bestehend aus 9 gleichartigen Teilproblemen und benötigt für jeden Teil 3 h. Der Anteil der parallelisierbaren Operationen beträgt  $\frac{2}{3}$ .  
Wie lange benötigt ein MIMD-Rechner mit 10 Prozessoren für das Gesamtproblem, falls die Kommunikation vernachlässigbar ist und jeder Prozessor die gleiche Rechenleistung wie der SISD-Rechner aufweist? Wie groß ist der Speedup und die Effizienz? Wie groß sind die Durchsätze dieser Rechner?

Fortsetzung nächste Seite!

**Aufgabe 4:****Anfragen auf relationalen Datenbanken**

Gegeben sei die folgende, in dritter Normalform vorliegende relationale Datenbank zur Modellierung des Gebrauchtwagenparks eines Autohändlers.

Modelle	mnr	hnr	typ	neupreis	ps
	1	1	Corsa	18.000	60
	2	1	Vectra	30.000	90
	3	1	Omega	40.000	110
	4	1	Astra	28.000	90
	5	2	Golf	30.000	90

Hersteller	hnr	hersteller
	1	Opel
	2	VW

Fahrzeuge	mnr	fgr	baujahr	preis
	1	H674	1992	13.000
	1	C634	1990	12.000
	2	D459	1992	22.000
	3	C634	1989	22.000
	5	H789	1993	24.000

Die Relation *Modelle* beinhaltet alle Fahrzeugtypen, die der Händler im Gebrauchtwagenprogramm führt. Die Modelle sind über das Attribut 'mnr' über alle Hersteller hinweg eindeutig nummeriert. 'mnr' ist daher Primärschlüssel der Relation *Modelle*. Über das Attribut 'hnr' wird von *Modelle* auf die Relation *Hersteller* verwiesen. In der Relation *Fahrzeuge* werden alle tatsächlich beim Händler am Lager befindlichen Fahrzeuge geführt. Über 'mnr' wird von *Fahrzeuge* auf *Modelle* verwiesen. Bei gegebener Modellnummer ist die vergebene Fahrgestellnummer ('fgr') eindeutig. Darum bilden 'mnr' und 'fgr' zusammen den Primärschlüssel der Relation *Fahrzeuge*. Formulieren Sie folgende Anfragen in jeweils drei der vier Anfragesprachen: relationale Algebra, relationer Tupelkalkül, SQL und Quel.

- Bestimmen Sie alle Modelle mit mehr als 60 PS.
- Bestimmen Sie die Typen aller Modelle des Herstellers VW.
- Bestimmen Sie die Nummern aller Modelle des Herstellers Opel, von denen tatsächlich Fahrzeuge auf Lager sind.

Formulieren Sie folgende Anfragen nur in SQL.

- a) Bestimmen Sie die Namen der Hersteller, für deren sämtliche Modelle mindestens ein Fahrzeug im aktuellen Bestand vorhanden ist.
- b) Bestimmen Sie den Durchschnittspreis (Attribut 'preis') der am Lager vorhandenen Opel-Fahrzeuge.
- c) Bestimmen Sie die jeweils durchschnittlichen Neupreise (Attribut 'neupreis') aller geführten Modelle der verschiedenen Hersteller.

Am 14.10.96  
Dr. Kroll

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Frühjahr 1997**



Kennzahl: \_\_\_\_\_

**Frühjahr**

**66111**

Kennwort: \_\_\_\_\_

**1997**

Arbeitsplatz-Nr.: \_\_\_\_\_

Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen

- Prüfungsaufgaben -

Fach: Informatik (vertieft studiert)

Einzelprüfung: Betriebs/Datenbanksyst., Rechnerarch.

Anzahl der gestellten Themen (Aufgaben): 1

Anzahl der Druckseiten dieser Vorlage: 3

Bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

Aufgabe 1 Maschinenprogrammierung

- 1a) Nennen und erläutern Sie vier verschiedene Adressierungsmodi der maschinennahen Programmierung.
- 1b) Erklären Sie anhand der folgenden Speicherbelegung die Adressierungsmodi 'Register-direkt', 'Register-indirekt' und 'Register-Speicher-indirekt'.

Speicheradresse	...	18	...	117	...	269	...	293	...	456	...
Speicherinhalt		293		269		456		117		18	

Es soll der Befehl

LOAD to Register1, Register2;

mit verschiedenen Adressierungsmodi ausgeführt werden. Der Inhalt von Register2 dient dazu, den Wert zu bestimmen, der in Register1 geladen wird. In Register2 befindet sich der Wert 293. Welcher Wert steht - abhängig vom jeweiligen Adressierungsmodus - nach Ausführung des Befehls in Register1?

- 1c) Beschreiben Sie die Schritte bei der Ausführung eines Unterprogrammaufrufs und -rücksprungs. Es soll das Stapelprinzip verwendet werden, wobei der Stapel zu kleineren Adressen wächst, der Stackpointer auf das nächste freie Element zeigt und das Abspeichern eines Elements auf dem Stack den Stackpointer um 4 verändert. Nennen Sie wenigstens zwei wesentliche Vorteile dieser Unterprogrammverwaltung.

Fortsetzung nächste Seite!

Aufgabe 2 Speichersystem

- 2a) Was versteht man unter Speicherzugriffszeit und unter Speicherzykluszeit? Wann sind beide gleich?
- 2b) Was versteht man unter der räumlichen und der zeitlichen Referenzlokalität? Erläutern Sie, wie diese zustande kommen! Geben Sie wenigstens drei Beispiele an, die zeigen, wie diese Eigenschaften von der Rechnerarchitektur ausgenützt werden können!
- 2c) Was versteht man unter einer Speicherhierarchie? Geben Sie ein Beispiel an! Wie lautet die Formel für die mittlere Speicherzugriffszeit bei einer zweistufigen Hierarchie?
- 2d) Welche Aufgaben hat die MMU (Memory Manage Unit)?  
Welches sind ihre wichtigsten Hardwarekomponenten?

Aufgabe 3 Ein-/Ausgabe

- 3a) Was versteht man unter einer speicherabgebildeten und unter einer separaten Ein-/Ausgabe? Geben Sie jeweils mindestens zwei charakterisierende Eigenschaften an! Nennen Sie für jede der beiden Organisationsformen wenigstens einen Vorteil!
- 3b) Was versteht man unter 'programmierter Ein-/Ausgabe'?  
Welche Alternativen dazu gibt es?  
Erläutern Sie den Unterschied zwischen 'Polling' und interruptgesteuerter Ein-/Ausgabe!
- 3c) Weshalb lassen sich Peripheriegeräte u.a. nicht direkt mit der CPU und dem Hauptspeicher verbinden? Nennen Sie mindestens 4 Möglichkeiten, eine Verbindung herzustellen!

**66111**

**Betriebssysteme / Datenbanksysteme / Rechnerarchitektur  
(vertieft)**

**Herbst 1997**

---

Prüfungsteilnehmer

Prüfungstermin

Einzelprüfungsnummer

---

Kennzahl: \_\_\_\_\_

**Herbst**

**66111**

Kennwort: \_\_\_\_\_

**1997**

Arbeitsplatz-Nr.: \_\_\_\_\_

---

Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen

- Prüfungsaufgaben -

Fach: Informatik (vertieft studiert)

Einzelprüfung: Betriebs/Datenbanksyst., Rechnerarch.

Anzahl der gestellten Themen (Aufgaben): 1

Anzahl der Druckseiten dieser Vorlage: 5


Bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

## 1. Aufgabe (virtuelle Speicherverwaltung)

- 1.1 Erklären Sie den Unterschied zwischen einer Maschinenadresse und einer virtuellen Adresse. Beschreiben Sie den Aufbau beider Adresstypen unter der Annahme von Segment-Seitenadressierung.
- 1.2 Erklären Sie die Schritte zur Abbildung der virtuellen Adresse auf die zugehörige Maschinenadresse; insbesondere beschreiben Sie dabei auch den Aufbau der dazu notwendigen Datenstrukturen.
- 1.3 Beschreiben Sie die Aktionen, die im Betriebssystem ablaufen, wenn der Prozessor auf ein Datum zugreifen will, dessen Seite gerade aus dem Arbeitsspeicher ausgelagert ist (Hinweis: Seitefehl-Alarm).
- 1.4 Betrachtung von Seitenersetzungsstrategien:

- a) Beschreiben Sie die Seitenersetzungsstrategien FIFO, LRU und LFU.
- b) Ein Programm umfasse 5 Seiten, während die Kapazität des Arbeitsspeichers 2 Seiten betrage. Der Arbeitsspeicher sei zu Anfang leer. Zeigen Sie anhand der folgenden Seitenzugriffsreihenfolge jeweils das Verhalten von FIFO, LRU, LFU:

3.	6.	9.	Zugriff
1	1	1	2
1	4	2	3
3	5	5	4
Seitennummer			
			
Zeit			

Wieviele Seitefehl-Alarme werden für jede der Ersetzungsstrategien ausgelöst?

- c) Wieviele Seitefehl-Alarme würde eine optimale Ersetzungsstrategie auslösen? Begründen Sie Ihre Antwort, indem Sie Ihre Strategie kurz beschreiben. Warum ist diese Strategie in der Praxis nicht anwendbar?
- d) Das Working-Set-Modell kann als formale Basis für Seitenersetzungsstrategien dienen. Definieren Sie den Begriff Working-Set, und geben Sie für die obige Zugriffsreihenfolge (siehe b) den Inhalt des Working-Sets jeweils nach dem 3., 6. und 9. Speicherzugriff an (unter der Annahme, daß  $\Delta = 3$  ist).
- 1.5 Welches Zugriffsverhalten haben die folgenden Programmkonstrukte, und wie wirkt sich dieses jeweils auf die Segment-Seitenadressierung aus?

Goto, Iteration, Rekursion, Modularität

Fortsetzung nächste Seite!

## 2. Aufgabe (Prozeß-Synchronisation)

- 2.1 Charakterisieren Sie den Begriff der Verklemmung anhand eines Betriebsmittelzuteilungsgraphen.
- a) Welche Methoden kennen Sie zur Verklemmungsauflösung?
  - b) Welche Methoden kennen Sie zur Verklemmungsvermeidung?
- 2.2 Implementieren Sie ein allgemeines, ganzzahliges Semaphor S mittels binärer Semaphore und einer integer-Variablen zusammen mit der P- und V-Operation für S (benutzen Sie dafür eine Pascal- oder C-ähnliche Programmiersprache).
- 2.3 Entwickeln Sie mittels Semaphore Lösungen für folgende Varianten des Leser-Schreiber-Problems! Welche der nachfolgenden Varianten können zu unendlichen Warten führen? Begründen Sie kurz Ihre Antwort!
- a) Jeweils nur ein Leser oder ein Schreiber ist zu einem Zeitpunkt erlaubt.
  - b) Jeweils nur ein Schreiber oder viele Leser sind zu einem Zeitpunkt erlaubt.
  - c) Jeweils nur ein Schreiber oder viele Leser sind zu einem Zeitpunkt erlaubt; Schreiber haben jeweils höhere Priorität.
  - d) Jeweils nur ein Schreiber oder viele Leser sind zu einem Zeitpunkt erlaubt; Schreiber haben jeweils höhere Priorität, jedoch nach Beendigung eines Schreibers werden die wartenden Leser berücksichtigt.
- 2.4 Erklären Sie, wie Semaphore und zugehörige Operationen im Betriebssystemkern realisiert werden können. Welche Voraussetzungen muß das Betriebssystem dabei erfüllen? Auf der Basis dieser Voraussetzungen skizzieren Sie einen Algorithmus zur Realisierung der P- und V-Operationen!

### 3. Aufgabe (Datenbankentwurf mit Entity-Relationship-Diagrammen)

Für ein Unternehmen soll eine Fertigungsdatenbank aufgebaut werden. Der Erhebungsprozeß liefere folgenden Informationsbedarf:

Entity-Mengen:

- ABTEILUNG mit den Attributen ANR, ANAME, AORT, MNR
- PERSONAL mit den Attributen PNR, NAME, BERUF
- MASCHINEN mit den Attributen MANR, FABRIKAT, TYP, BEZ, LEISTUNG
- TEILE mit den Attributen TNR, BEZ, GEWICHT, FARBE, PREIS

Relationship-Mengen:

- ABT-PERS zwischen ABTEILUNG und PERSONAL
- SETZT-EIN zwischen ABTEILUNG und MASCHINEN
- KANN-BEDIENEN zwischen PERSONAL und MASCHINEN
- GEEIGNET-FÜR-DIE-HERSTELLUNG-VON zwischen MASCHINEN und TEILE
- PRODUKTION zwischen PERSONAL, TEILE und MASCHINEN mit den Attributen DATUM und MENGE.

Dabei sollen folgende grundlegenden Integritätsbedingungen (in Form von erweiterten Kardinalitätsrestriktionen) gelten:

- Zu einer Abteilung gehört immer mindestens ein Beschäftigter.
- Eine Person ist immer nur genau einer Abteilung zugeordnet.
- Eine Maschine kann, wenn überhaupt, nur von einer Abteilung eingesetzt werden.
- Alle anderen (Teil-)Beziehungen sind nicht weiter eingeschränkt.

3.1 Zeichnen Sie zu dem obigen Szenario das zugehörige Entity-Relationship-Diagramm.

3.2 Legen Sie die Schlüsselkandidaten fest, und zeichnen Sie diese in das ER-Diagramm ein! Ergänzen Sie die o.g. Integritätsbedingungen um weitere erweiterte Kardinalitätsrestriktionen zur genauen Festlegung der Semantik der Miniwelt, und tragen Sie diese Informationen ebenfalls in das ER-Diagramm ein.

3.3 Führen Sie die gefundene Informationsstruktur über in Datenstrukturen nach dem Relationenmodell! Geben Sie im Relationenschema die Primär- und Fremdschlüssel an.



#### 4. Aufgabe (Benutzung von SQL)

Gegeben sei folgende Datenbank, die das Ausleihwesen einer Bibliothek unterstützt:

LESER: L (LSNR, NAME, VORNAME, WOHNORT, GEB.DAT)  
BUCH: B (ISBN, TITEL, SEITENZAHL, VERLAG, ERSCHEINUNGSJAHR, ANZAHL\_EXEMPLARE)  
VERLAG: V (VERLAG, VERLAGSORT, ...)  
EXEMPLAR: E (ISBN, EXPNR, INVENTARNR, STANDORT)  
AUSLEIHE: A (LSNR, ISBN, EXPNR, DATUM)

Formulieren Sie folgende Anfragen mit SQL:

- 4.1 Welche Leser haben Bücher ausgeliehen, die an ihrem Wohnort verlegt wurden?
- 4.2 Von welchen Buchtiteln sind alle Exemplare ausgeliehen?
- 4.3 In die LESER-Relation werden drei neue Leser(innen) gespeichert, deren Daten wie folgt lauten:

LSNR	NAME	VORNAME	WOHNORT	GEB.DAT
4711	Müller	Hugo	KL	11121955
4712	Maier	Maria	PS	05091900
4713	Meyer	Heike	unbekannt	unbekannt

#### 5. Aufgabe (Rechnernetze)

Charakterisieren Sie das Verhalten des Tokenring gegenüber dem Ethernet bei hoher bzw. niedriger Last. Für welche Einsatzbereiche und Situationen würden Sie den Tokenring gegenüber dem Ethernet vorziehen? In welchen Fällen würden Sie das Ethernet vorziehen? Begründen Sie und diskutieren Sie jeweils Ihre Wahl.