

Kennzahl: _____

Herbst

Kennwort: _____

2006**66113**Arbeitsplatz-Nr.: _____

Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen
- Prüfungsaufgaben -

Fach: **Informatik (vertieft studiert)**Einzelprüfung: **Rechnerarchitektur, Datenbanken, Betriebssysteme**Anzahl der gestellten Themen (Aufgaben): **2**Anzahl der Druckseiten dieser Vorlage: **11**

Thema Nr. 1

1. Normalformen

Gegeben sei das folgende Relationenschema:

Unterricht: {*LehrerPersnr*, *LehrerName*, *Klasse*, *SchülerName*, *SchülerVorname*,
SchülerGeburtsdatum, *Raum*, *Fach*}

Gehen Sie dabei davon aus, dass

- alle Schüler einer Klasse immer am gleichen Unterricht teilnehmen, und
- die Kombination aus Name, Vorname und Geburtsdatum für alle Schüler derselben Klasse verschieden ist.

- a) Zeigen Sie, dass das Relationenschema *Unterricht* nicht die dritte Normalform erfüllt!
- b) Bringen Sie *Unterricht* mit dem Synthesealgorithmus in 3NF!
- c) Ist Ihre Zerlegung abhängigkeiterhaltend?
- d) Prüfen Sie, ob Ihre Zerlegung auch bereits die BCNF erfüllt!

Fortsetzung nächste Seite!

2. SQL

Betrachten Sie das folgende Relationenschema:

<i>Klausuren:</i>	$\{\underline{KlausurNr}, Klasse, Datum, Fach\}$
<i>Schüler:</i>	$\{\underline{SchülerNr}, Name, Vorname, Klasse\}$
<i>KlausurNoten:</i>	$\{\underline{SchülerNr}, \underline{KlausurNr}, Note\}$
<i>MündlicheNoten:</i>	$\{\underline{SchülerNr}, Fach, \underline{Datum}, Note\}$

Erstellen Sie für die folgenden Anfragen SQL-Anweisungen! Sie dürfen sich zur Vereinfachung beliebige Sichten (Views) definieren.

- Berechnen Sie die mündliche Durchschnittsnote des Schülers »Harry Potter« im Fach »Arithmantik«!
- Welche Schüler der Klasse 3a haben die Klausur dieser Klasse in »Verteidigung gegen die dunklen Künste« am 1.3.2005 nicht mitgeschrieben? Geben Sie die Schülernummer, Name und Vorname aus!
- Erstellen Sie ein Zeugnis für »Hermine Granger«! Geben Sie dazu alle Fächer, in denen Noten für sie vorliegen, mit der berechneten Gesamtnote aus! Alle mündlichen Noten zusammen werden dabei wie eine Klausur gewertet.
- Welche Schüler der Klasse 4c haben in der Klausur dieser Klasse in »Zauberkunst« am 12.10.2005 eine überdurchschnittliche Note erzielt? Geben Sie die Schülernummern aufsteigend sortiert aus!

3. Relationale Algebra

Gegeben sei wieder das folgende Schema:

<i>Klausuren:</i>	$\{\underline{KlausurNr}, Klasse, Datum, Fach\}$
<i>Schüler:</i>	$\{\underline{SchülerNr}, Name, Vorname, Klasse\}$
<i>KlausurNoten:</i>	$\{\underline{SchülerNr}, \underline{KlausurNr}, Note\}$
<i>MündlicheNoten:</i>	$\{\underline{SchülerNr}, Fach, \underline{Datum}, Note\}$

- Geben Sie einen Ausdruck der relationalen Algebra an, um die Menge der Klausurnoten (und die zugehörigen Fächer) von »Ron Weasley« seit dem 20.12.2005 zu berechnen!
- Nennen Sie Methoden zur Optimierung von Ausdrücken der Relationenalgebra und beschreiben Sie diese kurz (1-2 Sätze)!
- Optimieren Sie folgenden Ausdruck:

$$\pi_{\{S\}} \left(\sigma_{(F='Arithmantik') \wedge (K1=K2)} \left(Klausuren(K1, A, D, F) \times KlausurNoten(S, K2, N) \right) \right)$$

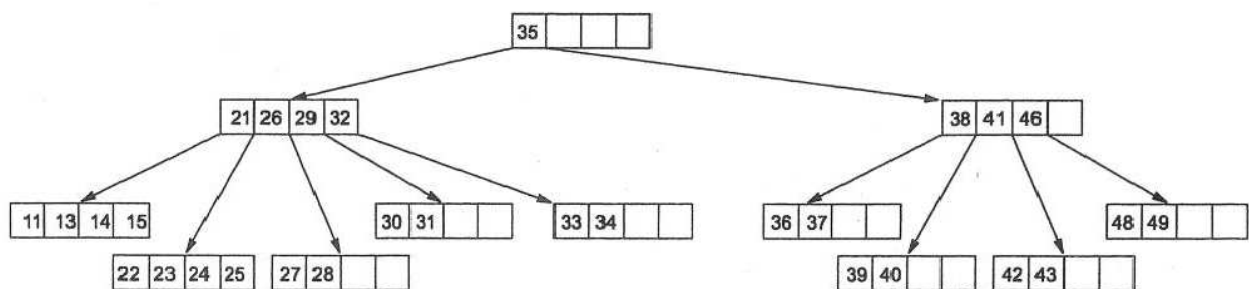
Fortsetzung nächste Seite!

4. Indexstrukturen

Der B-Baum ist als Indexstruktur in Datenbank-Systemen weit verbreitet. Führen Sie für folgenden B-Baum ($k = 2$) *nacheinander* folgende Operationen durch:

- Einfügen von 16
- Löschen von 46

Zeichnen Sie auch sämtliche Zwischenschritte! Teilbäume, in denen sich während des Einfügens bzw. Löschens nichts ändert, müssen nicht vollständig gezeichnet werden.



5. ISO/OSI-Schichten-Modell

- a) Nennen Sie die Schichten des ISO/OSI-Modells!
- b) Auf welcher Schicht findet die eigentliche Datenübertragung statt?

6. Maschinennahe Programmierung

Betrachten Sie folgende stark vereinfachte Maschine:

- Es gibt 8 Register RO-R7 mit je 2 Byte Breite.
- Einziges Datenformat ist eine 2 Byte lange Ganzzahl.
- Folgende Befehle sind verfügbar:

Befehl	Wirkung
MOV R_m, R_n	$R_n := R_m$
MOV int, R_n	$R_n := int$
ADD R_m, R_n, R_o	$R_o := R_m + R_n$
SUB R_m, R_n, R_o	$R_o := R_m - R_n$
JMP $marke$	unbedingter Befehlssprung zu Adresse $marke$
JZ $R_n, marke$	Befehlssprung zu $marke$, falls $R_n=0$

Dabei soll gelten:

- $m, n, o \in \{0, 1, \dots, 7\}$
 - *marke* ist eine Speicheradresse
 - *int* ist eine Ganzzahl.
- Jede Zeile eines Programms enthält genau einen Befehl.
 - Jede Zeile kann am Anfang eine Marke gefolgt von einem Doppelpunkt enthalten, um als Sprungziel zu dienen.
 - Beispiel für ein Programm:

```
anfang: MOV 10, R0
        MOV 1, R1
        SUB R0, R1, R0
        JZ R0, anfang
```

Setzen Sie die folgenden in Java-Syntax angegebenen Anweisungen in ein Programm der gegebenen Maschine um! Es wird dabei davon ausgegangen, dass der Wert der Variablen *i* bzw. *j* vor und nach der Abarbeitung des jeweiligen Befehls im Register R0 bzw. R1 zur Verfügung steht. Außerdem sind *i* und *j* maximal 2 Byte lange Ganzzahlen und es gilt $i, j > 0$.

- a) `i++;`
- b) `i*=j;`

7. Synchronisation

Fünf Philosophen sitzen auf jeweils einem Stuhl um einen runden Tisch. In der Mitte des Tisches stehen zwei Schüsseln mit Reis. Auf dem Tisch liegen fünf Stäbchen. Die Philosophen wiederholen ständig folgenden Ablauf:

- Denken
- Essen

Wenn ein Philosoph hungrig wird, muss er die Stäbchen zur Linken und zur Rechten in die Hand nehmen, um essen zu können. Man kann nur mit zwei Stäbchen essen. Allerdings kann man die Stäbchen nur nacheinander nehmen, nicht aber beide gleichzeitig. Ein Stäbchen, das gerade von einem Nachbarn gehalten wird, kann nicht genommen werden. Um essen zu können, braucht der Philosoph eine der beiden Schüsseln mit Reis. Während ein Philosoph isst, legt er die Stäbchen nicht aus der Hand, und kein anderer Philosoph kann aus seiner Schüssel essen. Nach dem Essen werden die Stäbchen und die Reisschüssel auf den Tisch gelegt und stehen wieder zur Verfügung.

- a) Vervollständigen Sie folgende Codeschablone, die den Ablauf für den Philosophen *i* zeigt! Der Zugriff auf die Variable *i* ist im Code möglich. Gehen Sie dabei davon aus, dass zur Synchronisation der Stäbchen ein Array von fünf binären Semaphoren zur Verfügung steht. Philosoph *i* benötigt zum Essen folgende Stäbchen (Semaphore):

```
staebchen[i] und staebchen[(i+1) mod 5]
```

Für die Reisschalen steht ein einzelnes mit 2 initialisiertes Zählsemaphor `schale` zur Verfügung.

An den mit ... gekennzeichneten Stellen können beliebig viele Codezeilen eingefügt werden.

```
while (true) {  
  
    ...  
  
    essen();  
  
    ...  
  
    denken();  
}
```

- b) Könnte es beim Essen der Philosophen bei ungeschickter Programmierung zu einem *Deadlock* kommen? Begründen Sie Ihre Antwort kurz!

Thema Nr. 2

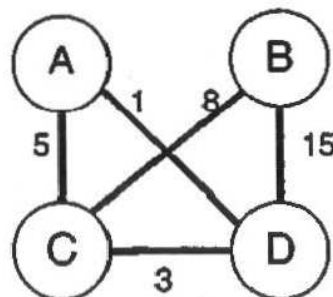
Aufgabe 1: Transportschicht

- a) Nehmen Sie an, Host A (Client) baut eine TCP-Verbindung zu Host B (Server) auf. Host A wählt als Startsequenznummer 111, Host B wählt 234. Es findet der übliche Drei-Wege-Handshake statt, wobei Host A im letzten der drei Handshaking-Pakete 200 Bytes an Daten versendet. Diese Daten werden von B korrekt bestätigt, B sendet keine Nutzdaten an A zurück. Nach Empfang der Bestätigung sendet Host A 50 Bytes an Host B, diese gehen jedoch verloren. Danach sendet Host A weitere 100 Bytes an Host B. Diese kommen fehlerfrei an und B sendet ein Acknowledgement an Host A zurück, das keine Daten enthält. Zeichnen Sie ein Diagramm des zeitlichen Ablaufs der Übertragungen (inkl. Handshaking)! Tragen Sie auch die TCP-Flags, Sequenznummern und Acknowledgement-Nummern für alle Pakete in das Diagramm ein! Es treten keine weiteren Paketverluste und keine Timeouts auf.
- b) Leiten Sie die Latenz zur Übertragung eines Objektes der Größe O mittels einer TCP-Verbindung über einen Link mit Übertragungsrate R her, bei dem sich die Fenstergröße ausschließlich wie im Congestion-Avoidance-Fall verhält, d.h. das Fenster hat anfangs eine Größe von 1 MSS und die Größe wird linear erhöht! Es darf angenommen werden, dass sich das Objekt vollständig in Segmente der Größe $S = \text{MSS}$ Bits zerlegen lässt. Es sollen keine Übertragungswiederholungen (keine Fehler, kein Verlust) auftreten. Geben Sie explizite Formeln für K , die Anzahl der Fenster, die das Objekt überdecken, und Q , die Anzahl der Wartezeiten für ein unendlich großes Objekt an! Leiten Sie damit P , die Anzahl der Wartezeiten des Servers zur Übertragung eines Objekts der Größe O , her und vereinfachen Sie die Latenzformel nach Einsetzen der Variable P möglichst weit!

Hinweis: $\sum_{i=1}^n i = \frac{n(n+1)}{2}$ $x^2 + px + q = 0 \Rightarrow x_{1,2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q}$
--

Aufgabe 2: Routing

Betrachten Sie folgendes Netzwerk mit 4 Knoten:



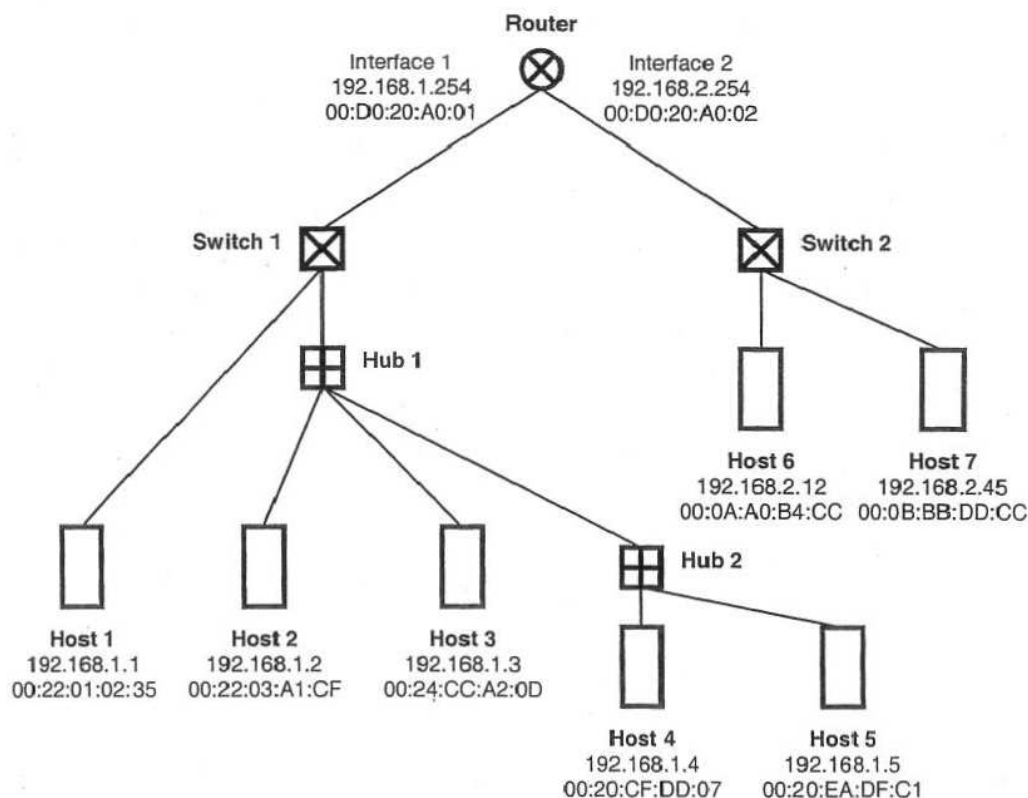
Fortsetzung nächste Seite!

Die Kantenmarkierungen kennzeichnen die jeweiligen Kosten.

- a) Führen Sie den Dijkstra-Algorithmus zur Bestimmung der Pfade mit geringsten Kosten von Knoten A zu allen anderen Knoten des Netzwerkes durch! Verfolgen Sie den Ablauf des Algorithmus für dieses Netz in einer Tabelle! Für jeden Schritt des Algorithmus soll die Tabelle eine Zeile enthalten. In jeder Zeile sollen folgende Angaben vorhanden sein:
- Die Menge der Knoten, für die der günstigste Pfad definitiv bekannt ist
 - Für jeden Knoten außer A:
 - Die Kosten von A zu diesem Knoten auf dem aktuell bekannten günstigsten Pfad
 - Der Vorgängerknoten auf diesem Pfad
- b) Wie kann Knoten A aus den Informationen in der in Aufgabe 2a erstellten Tabelle den günstigsten Pfad zu Knoten B bestimmen?

Aufgabe 3: Sicherungsschicht

Gegeben sei folgendes Netzwerk. Alle Verbindungen zwischen den Geräten sollten mittels Ethernet hergestellt werden. Für jedes Netzwerkinterface ist sowohl die IP-Adresse als auch die MAC-Adresse angegeben. Host 2 möchte nun per TCP mit Host 7 kommunizieren.



- a) Welche Ziel-MAC-Adresse wird beim Absenden des zugehörigen Ethernet-Frames in den Header eingetragen?
- b) Mit welcher Ziel-MAC-Adresse kommt der Ethernet-Frame bei Host 7 an? Welches Gerät trägt diese MAC-Adresse ein?
- c) Welches Protokoll wird verwendet, um diese Ziel-MAC-Adresse in Erfahrung zu bringen? Erläutern Sie kurz die Funktionsweise!

Fortsetzung nächste Seite!

Aufgabe 4: ER-Modellierung und Relationenmodell

4.1 ER-Modellierung

Erstellen Sie das Modell einer fiktiven Tanzschule in E/R-Notation! Wo möglich bzw. sinnvoll, sollen 3-fache Beziehungen und Generalisierung/Spezialisierung verwendet werden. Attribute von Entitäten und Beziehungen sind anzugeben; Schlüsselattribute werden durch Unterstreichen gekennzeichnet. Die Kardinalitäten von Beziehungen und - falls nötig - Rollennamen sollen ins Diagramm aufgenommen werden. Führen Sie Surrogatschlüssel nur ein, falls es nötig ist!

„Tanzschule“

Tanzschüler werden durch eine eindeutige Nummer, Namen, Geschlecht und Geburtsdatum beschrieben. Jeder Tanzschüler hat genau einen anderen Tanzschüler als „Tanzpartner“.

Tänze zeichnen sich durch eine eindeutige Kurzbezeichnung, einen Namen und einen gewissen Stil aus. Ein Tanzschüler hat bestimmte Tänze gelernt, andere nicht.

Ein **Tanzkurs** hat genau einen Tanzlehrer und mindestens einen Teilnehmer. Es werden in jedem Tanzkurs nur ausgewählte Tänze unterrichtet. Ein Tanzkurs hat einen Namen und ein Beginndatum. Er wird durch eine Nummer identifiziert. Ein Tanzschüler kann auch an mehreren Tanzkursen gleichzeitig teilnehmen oder pausieren. Nicht jeder Tanz wird in einem Tanzkurs unterrichtet.

Jeder **Tanzlehrer** kann nur ganz bestimmte Tänze unterrichten. Ein Tanzlehrer hat einen Vor- und Nachnamen und eine Zulassung. Er wird durch seinen Nachnamen identifiziert. Er kann keinen, einen oder mehrere Tanzkurse unterrichten. Für jeden Tanz im Repertoire muss es einen Tanzlehrer geben.

Dass an einem Kurs nur als Paar teilgenommen werden kann, ist nicht zu modellieren.

4.2 Relationenmodell

Ausgehend von der ER-Darstellung ist ein Relationenschema in dritter Normalform (3. NF) zu entwerfen! **Primärschlüssel werden dabei durch Unterstreichen, Fremdschlüssel durch Nennung der referenzierten Relation in eckigen Klammern hinter dem Attributnamen kenntlich gemacht, z.B.:**

```
Haus (Straße, OrtId[Ort])  
Ort  (OrtId, PLZ, Name)
```

Das Attribut OrtId der Relation Haus verweist als Fremdschlüssel auf das Attribut OrtId der Relation Ort.

Aufgabe 5: SQL

Bitte beachten Sie:

Primärschlüssel werden dabei durch Unterstreichen, Fremdschlüssel durch Nennung der referenzier-ten Relation in eckigen Klammern hinter dem Attributnamen kenntlich gemacht, z.B.:

```
Haus (Straße, OrtId[Ort])  
Ort  (OrtId, PLZ, Name)
```

Das Attribut **OrtId** der Relation **Haus** verweist als Fremdschlüssel auf das Attribut **OrtId** der Relation **Ort**.

Szenario: „Schule modelliert!“

Die Schulverwaltungsdatenbank eines Gymnasiums wird gerade neu aufgebaut und die Mitglieder des Kollegiums erkunden die neuen Möglichkeiten der Datenbank.

```
Raum (ZID, AnzahlPlaetze)  
Schueler (SID, Name, Vorname, Geburtsdatum, Klasse)  
Lehrer (LID, Name, Besoldungsstufe, Fachkombination)  
Unterricht (LID[Lehrer], Klasse, ZID[Raum], Tag, Uhrzeit)
```

Es existieren folgende Fremdschlüsselbeziehungen: Das Attribut **ZID** der Relation **Unterricht** bezieht sich auf das Attribut **ZID** der Relation **Raum**. Das Attribut **LID** der Relation **Unterricht** bezieht sich auf das Attribut **LID** der Relation **Lehrer**.

Die Attribute **ZID** der Relation **Raum**, **SID** der Relation **Schueler** und **LID** der Relation **Lehrer** sind Ganzzahlen. Bei dem Attribut **Tag** der Relation **Unterricht** handelt es sich um ein Datum. Das Attribut **Uhrzeit** wird als vierstellige Zeichenfolge im Format hh:mm angegeben. Das Attribut **AnzahlPlaetze** der Relation **Raum** ist eine dreistellige Ganzzahl. Das Attribut **Geburtsdatum** der Relation **Schueler** wird als Datum gespeichert. Die Datentypen der restlichen Attribute sind dem Kontext zu entnehmen.

Das Attribut **Klasse** der Relation **Unterricht** und das Attribut **Klasse** der Relation **Schue-ler** haben denselben Wertebereich.

Geben Sie SQL-Anweisungen für folgende Problemstellungen an:

- I) Erzeugung des beschriebenen Relationenschemas mit Tabellen, Primary Key Constraints und Foreign Key Constraints.
- II) Der Schüler „Hans Müller“ hat die Schule verlassen und ist aus der Tabelle **Schueler** zu löschen.
- III) Die Klasse „7b“ soll von der Lehrerin „Frau Schmidt“ (LID: 15) montags um 09:30 Uhr im Raum „02.124“ (ZID: 124) unterrichtet werden. „Frau Schmidt“ und Raum „02.124“ sind in den entsprechenden Tabellen bereits eingetragen.
- IV) Geben Sie den Namen der Lehrkraft, welche den meisten Unterricht in der Klasse „11a“ hält, aus!

Fortsetzung nächste Seite!

- V) In welcher Klasse befinden sich die meisten Schüler?
- VI) Geben Sie die Liste aller Räume mit genügend Plätzen für die Klasse „5c“ aus!

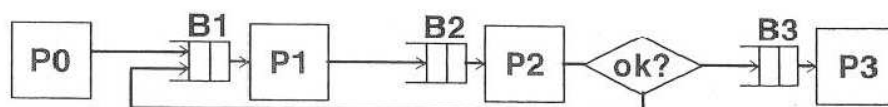
Aufgabe 6: Speicherverwaltung

- a) Skizzieren Sie (grafisch) die Abbildung einer logischen Adresse in eine physikalische Adresse in einem System mit Segmentierung!
- b) Was muss das Betriebssystem tun, wenn aufgrund von Hauptspeichermangel ein Segment ausgelagert werden soll? Beschreiben Sie den Ablauf und welche Änderungen in welchen der in Teilaufgabe a beschriebenen Datenstrukturen vorgenommen werden müssen!
- c) Was passiert, wenn der Prozess nach dem Auslagern erneut auf die Daten des Segments zugreift?
Beantworten Sie in Ihrer Beschreibung vor allem die folgenden Fragen:
- Welche Einheit des Systems erkennt, dass das Segment nicht vorhanden ist?
 - Woran wird das erkannt?
 - Welche Aktivitäten finden daraufhin im Betriebssystem oder in der Anwendung statt?
 - Welche Datenstrukturen werden in diesem Zusammenhang wie modifiziert?
 - Welche Prozesszustände nimmt der betroffene Prozess in welcher Phase ein?
- d) Ein Segment soll von zwei Prozessen gemeinsam genutzt werden. Was muss das Betriebssystem hierfür tun?
- e) Nennen Sie drei wesentliche Unterschiede zwischen Segmentierung und Seitenadressierung!

Aufgabe 7: Prozesse und Prozesskoordination

Gegeben sei folgendes Szenario: Ein Prozess P0 erzeugt Daten, die er zur Vorverarbeitung einem Prozess P1 übergibt. Das Ergebnis der Vorverarbeitung übergibt P1 dann an Prozess P2, der eine Weiterverarbeitung vornimmt. Das Ergebnis der Weiterverarbeitung wird entweder an Prozess P3 zur Ausgabe übergeben oder muss - falls es bestimmten Kriterien noch nicht genügt - nochmals an Prozess P1 zur erneuten Vorverarbeitung übergeben werden.

Die Prozesse P1 bis P3 verfügen jeweils über einen FIFO-Puffer (B1 - B3) mit n Speicherplätzen über den sie ihre Eingaben entgegen nehmen. P1 hat nur einen solchen Puffer (B1), über den sowohl P0 als auch P2 die Daten bei ihm anliefern. Die Prozesse geben ihre Daten grundsätzlich in Form von Datenpaketen weiter, ein Puffer-Speicherplatz kann ein solches Datenpaket aufnehmen.



Fortsetzung nächste Seite!

- a) Beschreiben Sie, an welchen Stellen Koordinierungsbedarf in diesem Szenario besteht und mit welchen Koordinierungsmechanismen man die Zugriffe jeweils koordinieren kann!
- b) Skizzieren Sie nun in einer programmiersprachlichen Form die Operationen `put` und `get`:
- `void put (message m)`
nimmt eine Nachricht entgegen und kopiert sie in den Puffer. Falls gerade kein Platz in dem Puffer frei ist, blockiert die Operation so lange, bis wieder ein Speicherplatz für die Nachricht frei wird.
- `message get ()`
entnimmt die nächste Nachricht aus dem Puffer. Falls der Puffer leer ist, blockiert die Operation so lange, bis wieder eine Nachricht vorliegt.
- c) Ist Ihre Lösung verklemmungsgefährdet? Begründen Sie Ihre Antwort, indem Sie beschreiben, wann eine Verklemmung auftreten kann bzw. warum Verklemmungen nicht auftreten werden!