

# $\begin{array}{c} {\rm Modul} \\ {\bf Grundlagen} \ {\bf von} \ {\bf Datenbanken} \end{array}$

Wintersemester 2020/2021

Probeklausur ??.??.2021, ?? Uhr, digital

## Musterlösung

1	2	3	4	5	6	7	8	9	10	Gesamt
18	12	12	14	12	6	8	12	11	15	120
	1 18	1         2           18         12				1 2 3 4 5 6	1 2 3 4 5 6 7	1 2 3 4 5 6 7 8	1 2 3 4 5 6 7 8 9	1 2 3 4 5 6 7 8 9 10 18 19 19 14 19 6 8 19 11 15

#### Rechtsmittelbelehrung:

Gegen die Bewertung dieser Prüfungsleistung kann innerhalb eines Monats nach ihrer Bekanntgabe Widerspruch erhoben werden. In diesem Zeitraum kann die Bewertung der Klausur eingesehen werden. Der Widerspruch ist schriftlich oder zur Niederschrift beim Vorsitzenden des B.Sc./M.Sc.-Prüfungsausschusses einzulegen. Es wird darauf hingewiesen, dass ein erfolgloses Widerspruchsverfahren kostenpflichtig ist.

#### Bearbeitungshinweise:

- Die Bearbeitungsdauer der Klausur beträgt insgesamt 120 Minuten.
- Es werden maximal 120 Punkte vergeben. Dabei entsprechen die den einzelnen Aufgaben zugeordneten Punkte den jeweils vorgesehenen Bearbeitungszeiten (in Minuten). Zum Bestehen der Klausur sind 50% der erreichbaren Punkte hinreichend.
- Es sind alle Hilfsmittel zugelassen, die keine Interaktion mit anderen Personen erfordern.

Viel Erfolg!

Matrikel-Nr:	Name:	
ausschließlicher Nutzu	n, dass ich die Klausur selbstständig und ng der erlaubten Hilfsmittel bearbeitet u elehrung zur Kenntnis genommen habe.	
Unterschrift.		

Blättern Sie erst weiter, wenn die Bearbeitungszeit offiziell begonnen hat.

## Aufgabe 1: Informationsmodellierung mit dem Entity-Relationship-Modell

[18 P.]

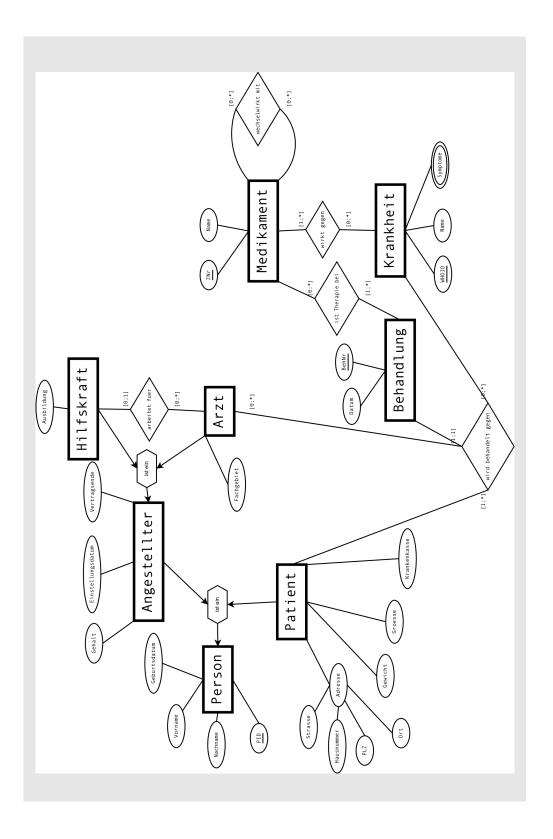
Erfassen Sie die im Folgenden beschriebenen Informationsstrukturen (Ausschnitt einer Anwendung zur Verwaltung der Patienten und deren Behandlungen in einem Ärztehaus) in einem ER-Diagramm. Beziehen Sie sich dabei genau auf die gegebene Beschreibung, ohne weiteres Wissen zu möglicherweise ähnlichen Anwendungsbereichen einfließen zu lassen. Markieren Sie in Ihrem Entwurf Primärschlüssel durch Unterstreichung und konkretisieren Sie die Abbildungstypen durch Kardinalitätsrestriktionen (Notation: [min,max]). Verwenden Sie unbedingt die aus der Vorlesung bekannte Notation. Andere Notationen werden nicht gewertet! Ausnahme: Die Zuordnung bei 1:n Abbildungstypen bleibt Ihnen überlassen, muss aber eindeutig als solche markiert werden (z.B. durch ein ausformuliertes Beispiel). Benutzen Sie möglichst wenige Entitäten (Ausnahme Vererbung).

Etwaige (Existenz-)Abhängigkeiten sollen nicht modelliert werden.

Es werden Angestellte und Patienten sowie deren Behandlungen betrachtet. Die Angestellten untergliedern sich in Hilfskräfte, welche eine Ausbildung haben, und Ärzte, welche ein Fachgebiet haben. Beiden Angestellten-Gruppen gemein ist, dass für sie ein Gehalt, ein Einstellungsdatum sowie ein Vertragsende festgelegt ist. Angestellte und Patienten haben personenspezifische Eigenschaften: eine eindeutige Identifikationsnummer (PID), einen Familiennamen, einen Vornamen und ein Geburtsdatum. Jede Hilfskraft kann (maximal) einem Arzt zugeordnet sein. Einem Arzt können wiederum mehrere Hilfskräfte zugeordnet werden. Für Patienten werden neben den bereits genannten personenspezifischen Eigenschaften noch die Größe, das Gewicht, die Adresse, welche sich aus einer Straße, einer Hausnummer, einer PLZ und einem Ort zusammensetzt, und der Name der Krankenkasse gespeichert.

Eine Krankheit hat einen eindeutigen Identifikationscode (WHOID), einen Namen und eine Reihe von Symptomen. Ein Medikament hat eine eindeutige Zulassungsnummer (ZNr) und einen Namen. Ein Medikament kann gegen eine oder mehrere Krankheiten einsetzbar sein. Zwar kann es für jede Krankheit mehrere geignete Medikamente geben, es ist aber auch möglich, dass eine Krankheit nicht durch Medikamente behandelbar ist. Zu beachten ist, dass ein Medikament Wechselwirkungen mit anderen Medikamenten haben kann, aber nicht haben muss.

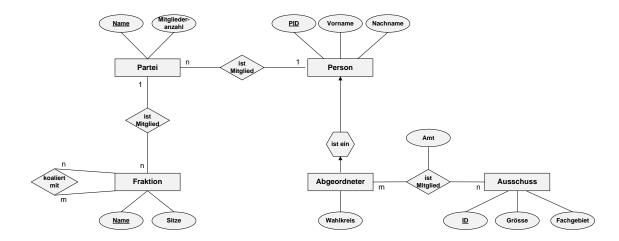
Eine Behandlung ist durch eine eindeutige Behandlungsnummer (BehNr) gekennzeichnet. Zudem besitzt sie ein Datum an dem sie stattfand. Eine Behandlung betrifft immer genau einen Patienten, einen Arzt und eine Krankheit. Ein Patient kann mehrmals (aber mindestens einmal) behandelt werden, ein Arzt kann mehrmals behandeln und eine Krankheit kann ebenfalls mehrmals behandelt werden. In einer Behandlung können beliebig viele (aber mindestens ein) Medikament angewendet werden. Ein Medikament kann in beliebig vielen Behandlungen Anwendung finden.



## Aufgabe 2: Abbildung eines ER-Diagramms auf das relationale Datenmodell

#### [12 P.]

Gegegeben sei folgendes ER-Diagramm:



Entwickeln Sie aus dem dargestellten ER-Diagramm ein entsprechendes relationales Datenbankschema anhand der in der Vorlesung erläuterten Abbildungsregeln. Stellen Sie sicher, dass Ihr Datenbankschema die minimale Anzahl von Relationen aufweist. Stellen Sie das resultierende DB-Schema dar, in dem Sie die notwendigen Relationenschemata in der Form

 $Relationenname(Attributname_1, Attributname_2, ...., Attributname_n)$ 

anführen und dabei jeweils den Primärschlüssel unterstreichen. Gegebenenfalls enthaltene Fremdschlüssel sind zu 'unterstricheln' und durch die aus den Übungen bekannte Pfeilnotation zu spezifizieren:

$$Attr_i \rightarrow Rel_b.Attr_j$$
.

Hinweis: Die dargestellten (1:n)-Beziehungen haben folgende Semantik:

- (i) Eine Person kann nur in einer Partei Mitglied sein; eine Partei kann mehrere Mitglieder haben.
- (ii) Eine Partei ist Teil höchstens einer Fraktion; einer Fraktion können mehrere Parteien angehören.

Hinweis: Die dargestellte (m:n)-Beziehung 'istMitglied' zwischen den Entity-Typen Abgeordneter und Ausschuss hat folgende Semantik:

(iv) Ein Abgeordneter kann in einem Ausschuss maximal ein Amt belegen.

Matrikel-Nr: Name:

```
\begin{aligned} & \operatorname{Person}(\underline{\operatorname{PID}}, \operatorname{Vorname}, \operatorname{Nachname}, \operatorname{Partei} \to \operatorname{Partei}.\operatorname{Name}) \\ & \operatorname{Abgeordneter}(\underline{\operatorname{PID}} \to \operatorname{Person}.\operatorname{PID}, \operatorname{Wahlkreis}) \\ & \operatorname{Partei}(\underline{\operatorname{Name}}, \operatorname{Mitgliederzahl}, \operatorname{Fraktion} \to \operatorname{Fraktion}.\operatorname{Name}) \\ & \operatorname{Fraktion}(\underline{\operatorname{Name}}, \operatorname{Sitze}) \\ & \operatorname{Ausschuss}(\underline{\operatorname{ID}}, \operatorname{Groesse}, \operatorname{Fachgebiet}) \\ & \operatorname{ist\_Mitglied}( \\ & \operatorname{Abgeordneter} \to \operatorname{Abgeordneter}.\operatorname{PID}, \operatorname{Ausschuss} \to \operatorname{Ausschuss}.\operatorname{ID}, \operatorname{Amt}) \\ & \operatorname{koaliert\_mit}( \ \underline{\operatorname{Partner1}} \to \operatorname{Fraktion}.\operatorname{Name}, \ \underline{\operatorname{Partner2}} \to \operatorname{Fraktion}.\operatorname{Name}) \end{aligned}
```

### Aufgabe 3: Relationenalgebra und SQL

[12 P.]

Gegeben seien die folgenden Relationenschemata:

```
Person(\underline{PNr}, Name, EMail, Alter, \underline{Werber} \rightarrow \underline{Person.PNr})
Artikel(\underline{ANr}, Bezeichnung, Menge, Startpreis, \underline{Verk\"{a}ufer} \rightarrow \underline{Person.PNr})
Gebot(\underline{Bieter} \rightarrow \underline{Person.PNr}, \underline{Artikel} \rightarrow \underline{Artikel.ANr}, Zeit, Betrag)
```

Benutzen Sie zur Lösung der folgenden Aufgaben ausschließlich die in der Vorlesung vorgestellten Operatoren der Relationenalgebra!

[3 P.] a) Geben Sie einen Relationenalgebra-Ausdruck an, der alle Artikel (ANr, Bezeichnung), auf welche eine Person mit dem Namen 'Müller' geboten hat, ausgibt.

```
\pi_{ANr,Bezeichnung}((\sigma_{Name='M\"{u}ller'}(Person)) \bowtie_{PNr=Bieter} Gebot \bowtie_{Artikel=ANr} Artikel)
```

[3 P.] b) Geben Sie einen Relationenalgebra-Ausdruck an, der alle Personen (PNr), welche noch keinen Artikel mit der Bezeichnung 'Buch' verkauft haben, ausgibt.

```
\pi_{PNr}(Person) - \pi_{PNr}(Person \bowtie_{PNr=Verk\"{a}ufer} (\sigma_{Bezeichnung='Buch'}(Artikel)))
oder
\pi_{PNr}(Person) - \pi_{Verk\"{a}ufer}(\sigma_{Bezeichnung='Buch'}(Artikel))
```

[3 P.] c) Geben Sie eine SQL-Anweisung an, welche die gleiche Ergebnismenge liefert wie der folgende Relationenalgebra-Ausdruck:

```
\pi_{Name}((Person \bowtie_{PNr=Bieter} Gebot \bowtie_{Artikel=ANr} Artikel) \bowtie_{Werber=PNr \land Verk \"{a}ufer=PNr} (\pi_{PNr}(Person)))
```

```
SELECT DISTINCT p1.Name
FROM Person p1, Gebot g,
Artikel a, Person p2

WHERE g.Bieter = p1.PNr
AND g.Artikel = a.ANr
AND a.Verkäufer = p2.PNr
AND p1.Werber = p2.PNr;
```

Matrikel-Nr: Name:

d) Der kleinste in der Datenbank gespeicherte Startpreis eines Artikels [3 P.] beträgt  $10 \in$  und der größte  $159 \in$ . Bestimmen Sie die Selektivität der Selektion  $\sigma_{Startpreis \geq 46}$  unter der Annahme einer Gleichverteilung:

$$sel(Startpreis \ge 46) = \frac{159 - 45}{159 - 9} = \frac{114}{150} = \frac{38}{50} = 0.76$$

In der Literatur wird vereinfacht häufig auch folgende Berechnung verwendet:

$$sel(Startpreis \ge 46) = \frac{\text{maxSel-minSel}}{\text{maxDB-minDB}} = \frac{159-46}{159-10} = \frac{113}{149} = 0.758$$

#### Aufgabe 4: SQL

[14 P.]

Gegeben seien die Relationenschemata aus Aufgabe 3:

```
Person(\underline{PNr}, Name, EMail, Alter, \underline{Werber} \rightarrow \underline{Person.PNr})
Artikel(\underline{ANr}, Bezeichnung, Menge, Startpreis, \underline{Verk\"{a}ufer} \rightarrow \underline{Person.PNr})
Gebot(\underline{Bieter} \rightarrow \underline{Person.PNr}, \underline{Artikel} \rightarrow \underline{Artikel.ANr}, Zeit, Betrag)
```

Formulieren Sie für die in den nachfolgenden Teilaufgaben angeführten, natürlichsprachlich formulierten Anfragen entsprechende SQL-Anweisungen. Verwenden Sie den in der Vorlesung verwendeten SQL-Standard. Das SQL-Schlüsselwort JOIN darf dabei nicht verwendet werden.

[2 P.] a) Die PNr, die EMail und alle gebotenen Beträge der Personen namens 'Harry' nach PNr absteigend sortiert.

```
SELECT p.PNr, p.EMail, g.Betrag
FROM Person p, Gebot g
WHERE g.Bieter = p.PNr
AND p.Name = 'Harry'
ORDER BY p.PNr DESC;
```

[3 P.] b) Die PNr, der Name sowie der in allen Geboten abgegebene Gesamtbetrag jeder Person, die älter als 30 Jahre alt ist.

```
SELECT p.PNr, p.Name, SUM(g.Betrag) as Gesamtbetrag
FROM Person p, Gebot g
WHERE p.Alter > 30
AND g.Bieter = p.PNr
GROUP BY (p.PNr, p.Name);
```

Matrikel-Nr: Name:

c) Die Namen und EMails aller Personen, die noch nie geboten haben. [3 P.]

```
SELECT (DISTINCT) p.Name, p.EMail
FROM Person p
WHERE NOT EXISTS(
SELECT *
FROM Gebot g
WHERE g.Bieter=p.PNr);

oder

SELECT (DISTINCT) p.Name, p.EMail
FROM Person p
WHERE p.PNr NOT IN(
SELECT g.Bieter
FROM Gebot g);
```

d) Die ANr, die Bezeichnung und den Startpreis jedes Artikels, dessen [3 P.] Verkäufer von einer Person mit dem Namen 'Bond' geworben wurde.

```
SELECT a.ANr, a.Bezeichnung, a.Startpreis
FROM Person p1, Person p2, Artikel a
WHERE a.Verkäufer = p1.PNr
AND p1.Werber = p2.PNr
AND p2.Name = 'Bond';
```

e) Die PNr und das durchschnittliche Alter der von ihm geworbenen Personen für jeden Werber, der maximal 10 Personen geworben hat.

```
SELECT p.Werber, AVG(p.Alter) as Durchschnittsalter
FROM Person p
GROUP BY p.Werber
HAVING(COUNT(*) <=10);
```

## Aufgabe 5: Änderungsoperationen auf Sichten

#### [12 P.]

Ausgehend von der Basisrelation

Musiker (PNr, Name, Alter, Genre, Vermögen, Status)

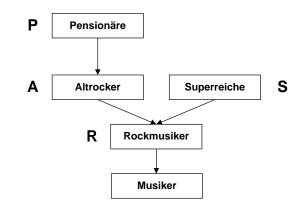
seien folgende Sichten definiert:

CREATE VIEW Rockmusiker AS SELECT \* FROM Musiker WHERE Genre = 'Rock';

CREATE VIEW Altrocker
AS SELECT \* FROM Rockmusiker
WHERE Alter > 50;

CREATE VIEW Pensionäre
AS SELECT \* FROM Altrocker
WHERE Status = 'Ruhestand';

CREATE VIEW Superreiche
AS SELECT \* FROM Rockmusiker
WHERE Vermögen > 100M;



Da diese Sichten Änderungsoperationen erlauben, **könnten** sie alle mit dem Zusatz WITH CASCADED CHECK OPTION definiert werden. In der Tabelle auf Seite 11 unten sind verschiedene Konfigurationen der Sichten mit bzw. ohne Check-Option dargestellt. Ein Spaltenwert 'CASC' bedeutet dabei, dass die Option WITH CASCADED CHECK OPTION für die zugehörige Sicht gesetzt ist; ein Spaltenwert '-' bedeutet, dass die zugehörige Sicht ohne CHECK OPTION definiert wurde.

Betrachten Sie folgende Änderungsoperationen:

```
    a) UPDATE Rockmusiker
    SET Alter = 49
    WHERE Alter = 51
    AND Status = 'aktiv'
    AND Vermögen < 50M;</li>
```

- b) INSERT INTO Superreiche VALUES (12, 'Madonna', 50, 'Pop', 500M, 'aktiv');
- c) UPDATE Rockmusiker SET Status = 'aktiv' WHERE Name = 'Beatles' AND Alter > 60 AND Vermögen > 200M;

d) UPDATE Pensionäre SET Alter = 48 WHERE Vermögen > 200M;

Markieren Sie im unteren Teil der untenstehenden Tabelle, welche dieser Änderungsoperationen von einem Datenbanksystem in den verschiedenen Konfigurationen erlaubt bzw. zurückgewiesen werden. Die Lösungen zur ersten Änderungsoperation sind als Hilfestellung bereits angegeben. Für die Fälle, in denen die Änderung bzw. das Einfügen zulässig ist, geben Sie in der Tabelle außerdem an, in welchen Sichten **auf jeden Fall alle** geänderten/eingefügten Tupel nach Abschluss der Operation sichtbar werden. Verwenden Sie dabei die folgende Notation:

nein Operation wird für die Konfiguration zurückgewiesen ja/Sicht-Liste Operation ist für die Konfiguration zulässig.

Alle geänderten/eingefügten Tupel sind anschließend in den in der Sicht-Liste aufgeführten Sichten sichtbar.

Zur Vereinfachung sind anstelle der kompletten Sichtennamen die folgenden Abkürzungen zu verwenden:

Rockmusiker R Altrocker A Pensionäre P Superreiche S

Rockmusiker	-	-	CASC	-
Altrocker	-	CASC	-	-
Pensionäre	-	-	-	CASC
Superreiche	-	-	CASC	-
a)	ja/R	ja/R	ja/R	ja/R
b)	ja/-	ja/-	nein	ja/-
c)	ja/R,A,S	ja/R,A,S	ja/R,A,S	ja/R,A,S
C)	J == 7 = - 7	0 / / /		

#### Aufgabe 6: Optimierung

[6 P.]

Gegeben seien folgende Relationenschemata:

 $Eigent\"{u}mer(\underline{EID}, Vorname, Nachname, Wohnsitz \rightarrow Immobilie.IID)$ 

 $Handwerksfirma(\underline{HID}, Name, Fachgebiet, Firmensitz \rightarrow Immobilie.IID)$ 

 $Immobilie(\underline{IID},\ Adresse,\ \underline{Baufirma} \rightarrow \underline{Handwerksfirma}.\underline{HID},\ \underline{Eigentuemer} \\ \rightarrow \underline{Eigentuemer}.\underline{EID})$ 

[3 P.]

1. Für die nachfolgende Anfrage soll eine algebraische Optimierung durchgeführt werden. Optimieren Sie die vorgegebene Anfrage anhand der in der Vorlesung eingeführten Regeln und zeichnen Sie den entsprechenden (optimierten) Operatorbaum. (Projektionen sollen dabei jedoch nicht nach unten gezogen werden.)

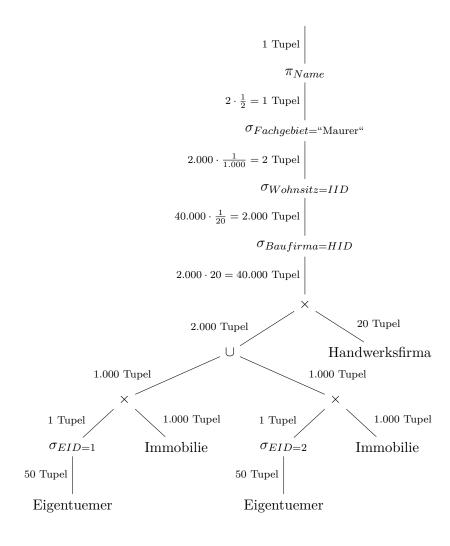
[3 P.]

2. Bewerten Sie den Operatorbaum mit den Kardinalitäten der Zwischenergebnisse. (Die Anzahl der Attribute soll dabei **nicht** betrachtet werden.) Für die zugehörige Datenbank werden folgende Kardinalitäten angenommen:

Card(Eigentuemer) = 50, Card(Immobilie) = 1000und Card(Handwekrsfirma) = 20. Unter den Handwekrsfirmen gibt es nur 2 verschiedene Fachgebiete wobei von denen jede gleich oft auftritt. Matrikel-Nr: Name:

 $\pi_{Name}(\sigma_{Fachgebiet=\text{``Maurer''}}(\sigma_{Wohnsitz=IID}(\sigma_{Baufirma=HID}(((\sigma_{EID=1}(Eigentuemer)\times Immobilie))\cup(\sigma_{EID=2}(Eigentuemer)\times Immobilie))\times Handwerksfirma))))$ 

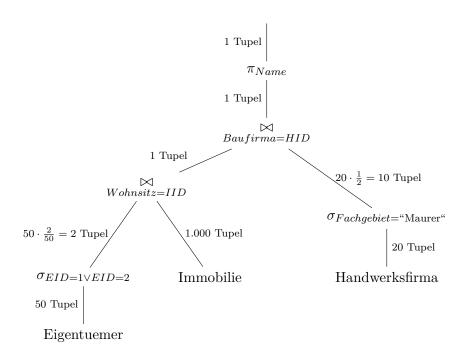
#### nicht optimiert:



#### optimiert:

$$\pi_{Name} \big( \sigma_{Fachgebiet=\text{``Maurer''}} (Handwerksfirma) \underset{HID=Baufirma}{\bowtie} \big)$$

$$\big( \sigma_{EID=1 \lor EID=2} (Eigentuemer) \underset{Wohnsitz=IID}{\bowtie} Immobilie \big) \big)$$



Matrikel-Nr:

Name:

### Aufgabe 7: Transaktionen

[8 P.]

a) Gegeben sind die drei Objekte x, y und z, welche von den Transaktionen  $T_1$ ,  $T_2$  und  $T_3$  gelesen bzw. geschrieben werden. Dabei bezeichnet  $w_i(x)$  den Schreibzugriff der Transaktion  $T_i$  auf das Objekt x und  $r_i(x)$  den Lesezugriff der Transaktion  $T_i$  auf x.

Bestimmen Sie bei den folgenden drei Schedules  $S_1$ ,  $S_2$  und  $S_3$  jeweils, ob diese serialisierbar sind oder nicht. Begründen Sie Ihre Antwort, indem Sie entweder alle äquivalenten, seriellen Transaktionsreihenfolgen (z.B.  $T_1T_2T_3$ ) oder einen auftretenden Konfliktzyklus angeben.

i)  $S_1 = r_1(y) r_2(z) w_3(y) w_2(x) r_3(y) r_1(y) r_3(x)$ 

Hier ergibt sich ein Konfliktzyklus: Mit  $r_1(y) \rightarrow w_3(y)$  folgt  $T_1$  vor  $T_3$ Mit  $w_3(y) \rightarrow r_1(y)$  folgt  $T_3$  vor  $T_1$ Mit  $w_2(x) \rightarrow r_3(x)$  folgt  $T_2$  vor  $T_3$ 

ii)  $S_2 = r_1(z) \ r_3(x) \ r_2(x) \ r_1(y) \ w_2(z) \ r_3(y) \ w_3(y)$ 

mögliche seriellen Abfolgen lauten:  $T_1T_2T_3$  und  $T_1T_3T_2$ 

iii)  $S_3 = w_2(y) r_3(z) r_1(x) r_1(y) w_3(x) w_3(z)$ 

mögliche serielle Abfolge lautet: T<sub>2</sub>T<sub>1</sub>T<sub>3</sub>

b) Anomalien bei unkontrolliertem Mehrbenutzerbetrieb

Betrachten Sie die zwei Transaktionen  $T_1$  und  $T_2$ , deren Operationen ohne Mehrbenutzerkontrolle in der nachfolgend angegebenen Reihenfolge ausgeführt werden. Geben Sie eine Operation von  $T_2$  zum Zeitschritt 3 an (bei Platzmangel bitte unter, **nicht** in die Tabelle schreiben), die für  $T_1$  ein non-repeatable Read verursacht, und begründen Sie kurz die Entstehung dieser Anomalie im vorliegenden Beispiel.

17

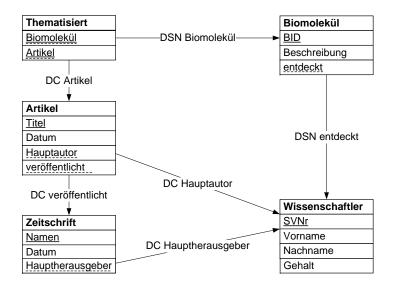
Zeitschritt	$T_1$	$T_2$
1	SELECT SUM(Gehalt)	
	FROM Personal	
	WHERE Abteilung = 'Treadstone';	
2		SELECT AVG(Alter)
		FROM Personal;
3		UPDATE Personal
		SET Gehalt = $Gehalt*2$
		WHERE Abteilung = 'Treadstone';
4	SELECT SUM(Gehalt)	
	FROM Personal	
	WHERE Abteilung = 'Treadstone';	
5	COMMIT;	
6		COMMIT;

 $T_2$  ändert in Zeitschritt 3 Tupel, die von  $T_1$  in Zeitschritt 1 gelesen wurden. Ein erneutes Lesen dieser Tupel, wie es in Zeitschritt 4 geschieht, liefert ein anderes Ergebnis als das erste Lesen. Weitere Lösungen sind möglich, zum Beispiel eine Löschoperation auf der Relation 'Personal', welche Tupel mit dem Wert 'Treadstone' im Attribut 'Abteilung' betrifft.

### Aufgabe 8: Referentielle Aktionen

[12 P.]

Gegeben sei folgender Referenzgraph für eine Biomoleküldatenbank.



Die referentiellen Aktionen der Fremdschlüssel in den Abbildungen sind jeweils abgekürzt mit 'DC' für 'ON DELETE CASCADE', 'DR' für 'ON DELETE RESTRICT' bzw. 'DSN' für 'ON DELETE SET NULL'.

1. Welche Anforderung erfüllt ein (bzgl. der referentiellen Aktionen) si- [1 P.] cheres Schema?

Bei einem sicheren Schema ist das Ergebnis einer Änderungsoperation unabhängig von der Reihenfolge, in der die referentiellen Aktionen ausgeführt werden; es treten also keine reihenfolgeabhängigen Ergebnisse auf.

2. Handelt es sich im vorliegenden Fall um ein sicheres Schema? [1 P.]

 $\otimes$  Ja

O Nein

Die theoretisch einzig mögliche Stelle für eine Reihenfolgenabhängigkeit im Schema existiert beim Löschen eines Wissenschaftlers, da beim Kaskadieren des Löschens auf zwei Wegen bis zum Artikel kaskadiert werden kann; Einmal über die Zeitschrift und einmal auf direktem Wege. **Das vorliegende Schema ist sicher**, da auf beiden Wegen keine blockierenden referentiellen Aktionen wie beispielsweise 'ON DELETE RESTRICT' verwendet werden.

[5 P.] 3. Die Fremdschlüssel der Relation 'Artikel' wurden durch folgende SQL-DDL-Anweisung erzeugt:

```
CREATE TABLE Artikel (
                    VARCHAR(50) PRIMARY KEY,
    Titel
    Datum
                    DATE
                                NOT NULL,
                    INTEGER
                                 NOT NULL,
    Hauptautor
    veröffentlicht
                    VARCHAR(50) NOT NULL,
    CONSTRAINT fkHauptautor FOREIGN KEY (Hauptautor)
        REFERENCES Wissenschaftler (SVNr) ON DELETE CASCADE
    CONSTRAINT fkVeröffentlicht FOREIGN KEY (veröffentlicht)
        REFERENCES Zeitschrift (Name) ON DELETE CASCADE
);
```

Für den Fremdschlüssel 'fk Veröffentlicht' aus der Relation 'Artikel' soll nun die referentielle Aktion von 'ON DELETE CASCADE' auf "ON DELETE RESTRICT" geändert werden:

Geben Sie eine Folge von **SQL-DDL**-Anweisungen an, welche diese Schemaänderung realisiert und dabei die Löschung der Relation selbst vermeidet.

Siehe auch die Folien zur Schemaevolution, z.B. Folie K5/F106:

ALTER TABLE Artikel DROP CONSTRAINT fkVeröffentlicht;

```
ALTER TABLE Artikel ADD CONSTRAINT fkVeröffentlicht FOREIGN KEY (veröffentlicht)
REFERENCES Zeitschrift (Name) ON DELETE RESTRICT;
```

4. Handelt es sich bei dem geänderten Schema um ein sicheres Schema? Falls das Schema nicht sicher ist, beschreiben Sie ein Szenario, in dem das Ergebnis einer Änderungs-/Löschoperation von der Auswertungsreihenfolge der referentiellen Aktionen abhängig ist.

O Ja

⊗ Nein

Das vorliegende Schema ist jetzt nicht mehr sicher, da beim Löschen eines Wissenschaftlers die Löschoperation direkt und über 'Zeitschrift' zu 'Artikel' propagiert wird. Dies führt zu reihenfolgeabhängigen Ergebnissen, wenn ein Wissenschaftler Hauptherausgeber einer Zeitschrift ist, in der nur Artikel veröffentlicht wurden, bei denen der zu löschende Wissenschaftler Hauptautor ist:

- Wird zunächst der direkte Weg beschritten, wird der Artikel gelöscht. Da kein Artikel mehr die Zeitschrift des Wissenschaftlers referenziert, kann nun auch die Zeitschrift gelöscht werden.
- Wird zunächst der Weg über 'Zeitschrift' beschritten, wird das Löschen der Zeitschrift jedoch verweigert, da die zu löschende Zeitschrift noch von einem Artikel referenziert wird.

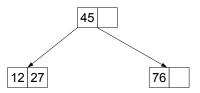
[5 P.]

## Aufgabe 9: B-Bäume

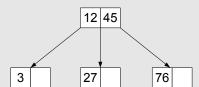
[11 P.]

Anmerkung: Die B-Bäume sind im Folgenden stets gemäß der vereinfachten Darstellungsart aus der Vorlesung abgebildet.

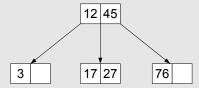
a) Nehmen Sie den (Standard-) Split-Faktor 1 an und fügen Sie in den [5 P.] unten abgebildeten **B-Baum** der Klasse  $\tau(1,h)$  die Datensätze mit den Schlüsselwerten **3, 17, 83, 33 und 79** in dieser Reihenfolge ein. Zeichnen Sie den Baum nach jedem Split-Vorgang und erläutern Sie kurz die durchgeführten Maßnahmen (Splitten, einfaches Einfügen).



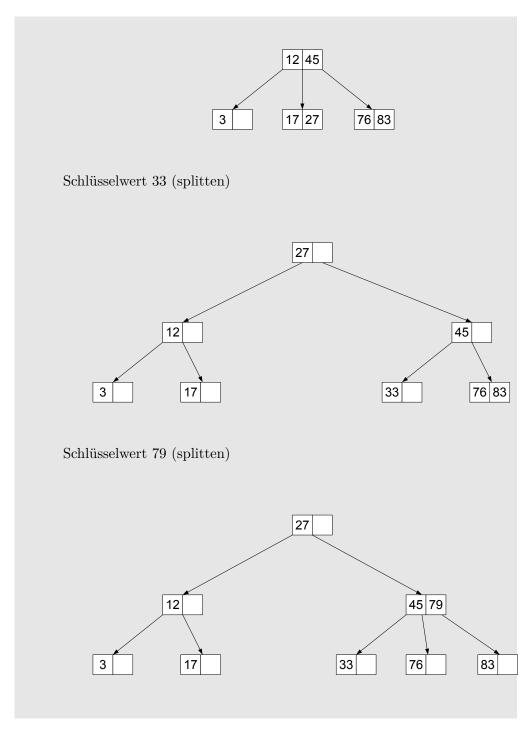
Schlüsselwert 3 (Splitten)



Schlüsselwert 17 (einfaches Einfügen)



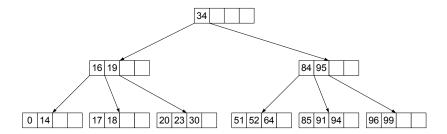
Schlüsselwert 83 (einfaches Einfügen)

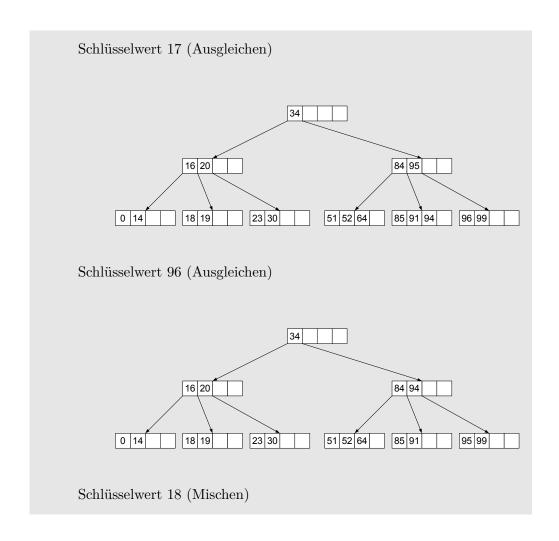


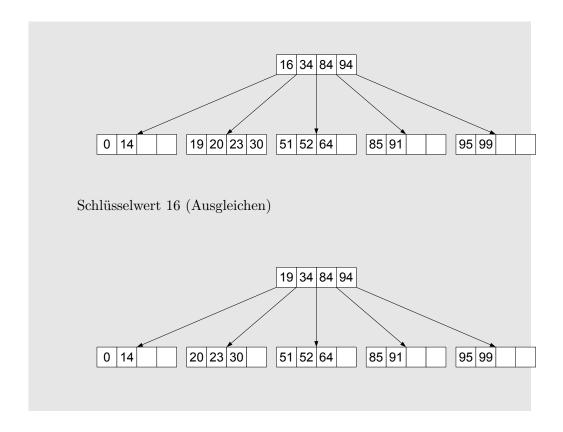
[6 P.]

b) Löschen Sie aus dem unten abgebildeten **B-Baum** der Klasse  $\tau(2,h)$  die Datensätze mit den Schlüsselwerten **17, 96, 18 und 16** (in dieser Reihenfolge). Zeichnen Sie den Baum nach jedem Löschvorgang und geben Sie jeweils kurz an, welche konkrete Maßnahme Sie durchgeführt

haben (Mischen, Ausgleichen, einfaches Löschen). Für Ausgleichs- und Mischoperationen sollen nur direkt benachbarte Geschwisterknoten (bevorzugt der rechte) herangezogen werden.







## Aufgabe 10: Berechnungen in B- und B\*-Bäumen

[15 P.]

a) Gegeben seien ein maximal gefüllter B-Baum der Klasse  $\tau(3,2)$  und [11 P.] ein minimal gefüllter B\*-Baum der Klasse  $\tau(2,3,3)$ .

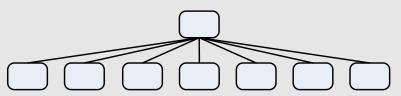
**Anmerkung:** Begründen Sie in den folgenden Teilaufgaben Ihr jeweiliges Ergebnis durch einen Rechenweg.

(1) Wie groß ist die Differenz der Anzahl an Datensätzen der jeweiligen Bäume? [6 P.]

Ein maximal gefüllter B-Baum der Klasse  $\tau(3,2)$  hat 48 Datensätze, ein minimal gefüllter B\*-Baum der Klasse  $\tau(2,3,3)$  dagegen nur 18 Datensätze. Differenz: 30 Datensätze

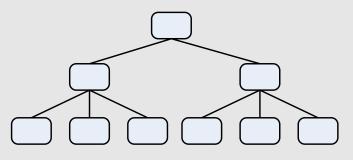
maximal gefüllter B-Baum:

(1 Wurzel + 7 Blattknoten = 8 Knoten mit je 6 Datensätzen)



minimal gefüllter B\*-Baum:

(1 Wurzel (1 Eintrag) + 2 innere Knoten (je 2 Einträge) + 6 Blattknoten = 6 Knoten mit je 3 Datensätzen)



(2) Wenn man sequentiell alle Datensätze aus einem der beiden Bäu- [3 P.] me lesen möchte, wieviele verschiedene Knoten (Seiten) muss man in dem jeweiligen Baum lesen?

Bei dem B-Baum muss man 8 (Wurzel + 7 Blätter) verschiedene Knoten (Seiten) lesen. Beim B\*-Baum sind es ebenfalls 8 Knoten (Seiten) (einmal von Wurzel zu linken äusseren Blatt, danach über den  $sequence\ set$  alle Blätter).

[2 P.] (3) Wenn man wahlfrei einen Datensatz aus einem der beiden Bäume lesen möchte, wieviel Knoten (Seiten) muss man im jeweiligen Baum maximal lesen?

Bei dem B-Baum muss man maximal 2 Knoten (Seiten) lesen, beim B\*-Baum hingegen immer 3 Knoten (Seiten).

[4 P.] b) Gegeben sei ein  $\mathbf{B^*}$ -Baum der Klassse  $\tau(k,k^*,h)$ . Innerhalb des zugehörigen (fiktiven) DBS wird vereinfachend eine Seitengröße von 800B (L=800B) angenommen. Die weiteren Kenngrößen sind dabei wie folgt festgelegt:

$$\begin{array}{rcl} l_M & = & 4B \\ l_D & = & 160B \\ l_K & = & 5B \\ l_P & = & 6B \end{array}$$

Die inneren Baum- und Blattknoten entsprechen jeweils genau einer Seite. Wie groß sind demnach k und  $k^*$ ?

Begründen Sie Ihr Ergebnis durch einen Rechenweg.

Hinweis: Die oben genannten Kenngrößen haben folgende Bedeutung:

$l_M$	die Länge der Seiteninformationen einer Seite
$l_D$	die Länge eines Datensatzes
$l_K$	die Länge eines Schlüsselwertes
$l_P$	die Länge eines Zeigers

$$k = \lfloor \frac{L - l_M - l_P}{2 \cdot (l_K + l_P)} \rfloor = \lfloor \frac{800B - 4B - 6B}{2 \cdot (5B + 6B)} \rfloor = \lfloor \frac{790B}{22B} \rfloor \simeq \lfloor 35, 9 \rfloor = 35$$

$$k^* = \lfloor \frac{L - l_M - 2 \cdot l_P}{2 \cdot (l_K + l_D)} \rfloor = \lfloor \frac{800B - 4B - 12B}{2 \cdot (5B + 160B)} \rfloor = \lfloor \frac{784B}{330B} \rfloor \simeq \lfloor 2, 376 \rfloor = 2$$