Datenbanksysteme Übungsblatt 1

Sommersemester 2003

AIFB
Institut für
Angewandte Informatik und
Formale Beschreibungsverfahren

Aufgabe 1a *(1/2)*

Änderungsanomalie: Wenn eine Änderung nicht überall ordnungsgemäß durchgeführt wird, kann es zu Inkonsistenzen kommen (update-Anomalie)

Bsp.: Betrachte Tabelle „HOTEL" und Tabelle „BUCHUNG". Nehmen wir an, es liegen drei verschiedene Buchungen für das Hotel Holiday Inn in München vor, d.h. drei Tupel in Tabelle „BUCHUNG". Das Hotel zieht nun um, d.h. die Adresse ändert sich. Wird die neue Adresse nun nicht konsequent von der Tabelle „HOTEL" in die Tabelle „BUCHUNG" übernommen, kommt es zu Inkonsistenzen.

HOTEL					
Hotelname	**Straße**	**PLZ**	**Ort**	**Freie Zimmer**	**Preis p.N**
Zur Post	Kanalstr. 123	20899	Haumburg	56	90
Holiday Inn	City-Ring 56	80755	München	78	120
Hilton	Geisbockstr. 2	Aholming	Elektro	Vertr	8

BUCHUNG							
AuftrNr.	**KundenNr**	**FlugNr.**	**Flugpl.**	**Hotelname**	**Straße**	**Ort**	**Zimmer**
34	5011	2264	1	Holiday Inn	City-Ring 56	MUC	2
38	5089	7621	1	Holiday Inn	City-Ring 56	MUC	4
42	5129	8761	2	Holiday Inn	Moseweg 5	MUC	1

Skript: S. 17 - 20

Aufgabe 1a (2/2)

Einfügeanomalie: Informationen können nur dann in die Tabelle aufgenommen werden, wenn alle Spalten der Tabelle ausgefüllt werden können.

Mit Nullwerten kann dieses Problem umgangen werden.

Bsp.: Betrachte Tabelle „BUCHUNG“. Angenommen ein Kunde möchte ein Zimmer im Holiday Inn reservieren, weiß aber noch nicht, ob er den Flug um 15:00 Uhr oder um 17:30 Uhr nehmen kann. Obwohl die Reservierung des Zimmers für sich genommen sinnvoll wäre, kann sie mit diesem System nicht durchgeführt werden.

BUCHUNG							
AuftrNr.	**KundenNr**	**FlugNr.**	**Flugpl.**	**Hotelname**	**Straße**	**Ort**	**Zimmer**
34	5011	?	1	Holiday Inn	City-Ring 56	MUC	2
38	5089	7621	1	Holiday Inn	City-Ring 56	MUC	4

Skript: S. 17 - 20

Aufgabe 1b

Semantische Integritätsbedingungen (sIB):

- innerhalb von Tabellenspalten:
  - Postleitzahlen dürfen nur zwischen 01000 und 99999 liegen (*stat. Bed.*)
  - Anzahl der freien Zimmer in Hotels bzw. freier Plätze in Flügen muß immer >= 0 sein (*stat. Bed.*)
- innerhalb einer Tabelle:
  - Kundennummern sind alle verschieden (analog Auftrags-Nr., Hotel-Nr., Flug-Nr.) (*stat.Bed.*)
- tabellenübergreifend
  - in Tabelle „BUCHUNG“ dürfen nur Kundennummern vorkommen, die auch in Tabelle „KUNDE“ vorkommen (analog Flug-Nr., Hotel-Nr.) (*dyn. Bed.*)
  - die Anzahl der gebuchten Plätze für einen Flug bei einer Neubuchung darf nicht größer sein, als die Anzahl der freien Plätze für diesen Flug in Tabelle „FLUGVERBINDUNG“ (analog Hotelreservierungen) (*dyn. Bed.*)

Skript: S. 36 - 42

Aufgabe 2a

Beim Tabellenentwurf können drei verschiedene Arten von Anomalien auftreten:

Lösch-Anomalie (deletion anomaly)	beim Löschen gehen mehr Informationen verloren, als erwünscht
Änderungs-Anomalie (update anomaly)	Wenn eine Änderung nicht überall ordnungsgemäß durchgeführt wird, entstehen Inkonsistenzen
Einfüge-Anomalie (insertion anomaly)	Informationen können nur dann in die Tabelle aufgenommen werden, wenn alle Spalten der entsprechenden Tabellenzeile ausgefüllt werden können

Skript: S. 17 - 20

Aufgabe 2b *(1/2)*

Wie kann man Probleme vermeiden:

- bereits beim Tabellenentwurf darauf achten, daß möglichst wenig Redundanzen auftreten (vgl. Aufgabe 1a)
- Tabellen „geeignet“ zerlegen:

Unter einer geeigneten Zerlegung versteht man eine Zerlegung, die sowohl **verlustfrei** ist (d.h. nach Zusammenfügen der zerlegten Tabellen erhält man wieder dieselbe Information, wie vor dem Zerlegen), als auch die oben genannten Anomalien beseitigt.

Beispiel: Bei der Tabelle BUCHUNG kann keine Änderungsanomalie auftreten, aber Einfügeanomalien sind möglich.

BUCHUNG					
AuftrNr.	**KundenNr.**	**FlugNr.**	**Plätze**	**Hotel.Nr**	**Zimmer**
2264	5011	2264	1	17	1

Skript: S. 22 - 27

Aufgabe 2b **(2/2)**

Wenn die Tabelle „BUCHUNG“ geeignet in zwei Tabellen „FLUGRESERVIERUNG“ und „ZIMMERRESERVIERUNG“ zerlegt wird, kann die Einfügeanomalie vermieden werden. Durch die Join-Operation erhält man dann wieder die alte Tabelle .

FLUGRESERVIERUNG		
KundenNr.	FlugNr.	Plätze
5011	2264	1

∗(=Join)

ZIMMERRESERVIERUNG			
AuftrNr.	KundenNr.	Hotel.Nr	Zimmer
2264	5011	17	1

BUCHUNG					
AuftrNr.	KundenNr.	FlugNr.	Plätze	Hotel.Nr	Zimmer
2264	5011	2264	1	17	1

Skript: S. 22 - 27

AIFB 26.05.2003

Aufgabe 3a **(1/2)**

Nichttriviale funktionale Abhängigkeiten

Funktionale Abhängigkeit	Gültig (j/n)
MATNR ➔ STUD-NAME	gilt, anhand der Matrikelnummer kann der Name des Studenten eindeutig bestimmt werden
STUD-NAME ➔ MATNR	gilt nur, wenn der Studentenname eindeutig ist; allgemein nicht
KLAUSURNAME ➔ PRUEFER	gilt nicht, wenn zwei verschiedenen Prüfer dieselbe Prüfung prüfen
KLAUSURNAME ➔ INSTITUT	gilt nicht, wenn es zwei Klausuren gleichen Namens an verschiedenen Instituten gibt
PRUEFER ➔ KLAUSURNAME	gilt nicht, sonst dürfte ein Prüfer nur eine Klausur abhalten

AIFB 26.05.2003

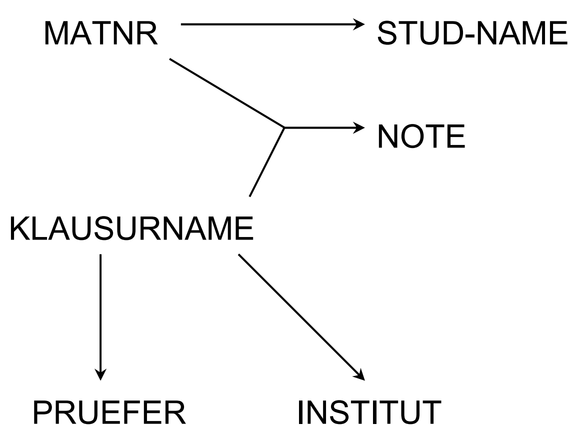
Aufgabe 3a

Funktionale Abhängigkeit	Gültig (j/n)
PRUEFER → INSTITUT	gilt nicht mehr, wenn zwei verschiedene Prüfer mit gleichem Namen an verschiedenen Instituten arbeiten
NOTE → MATNR	gilt nicht
NOTE → STUD-NAME	gilt nicht
MATNR, KLAUSURNAME → NOTE	gilt

Aufgabe 3b

Funktionale Abhängigkeit
MATNR → STUD-NAME
KLAUSURNAME → PRUEFER
KLAUSURNAME → INSTITUT
MATNR, KLAUSURNAME → NOTE

Abhängigkeitsgraph



Aufgabe 4a *(1/2)*

Bildung von $\{b\}^+$ mit Hilfe des Algorithmus APLUS

A'	**A⁺**	**F'**
b	b	ab → c , b → ce , c → d , a → d
b	bce	ab → c , c → d , a → d
bce	bcde	ab → c , a → d
bcde	bcde	

Ergebnis: $\{b\}^+ = \{bcde\}$

Skript: S. 73

Aufgabe 4b *(1/2)*

Bildung von $\{ab\}^+$ mit Hilfe des Algorithmus APLUS

A'	**A⁺**	**F'**
ab	ab	ab → c , b → ce , c → d , a → d
Ab	abc	b → ce , c → d , a → d
abce	abce	c → d , a → d
abce	abcde	a → d
abcde	abcde	

Ergebnis: $\{ab\}^+ = \{abcde\}$

Skript: S. 73

Aufgabe 5a

Zur Bestimmung der abgeschlossenen Hülle F^+ nutzen wir Lemma 1.3:

Zu r: (U | F), $F \subseteq \Phi(U)$ ist $F^+ = \{ A \rightarrow B \mid A \subseteq U, B \subseteq A^+\}$

Bilde von den linken Seiten jeweils A^+ :

$\{ab\}^+ = U = \{abcde\}$
$\{c\}^+ = U = \{abcde\}$
$\{ac\}^+ = U = \{abcde\}$
$\{d\}^+ = \{ad\}$
$\{e\}^+ = \{be\}$

$F^+ = \{ab \rightarrow c, ab \rightarrow d, ab \rightarrow e, c \rightarrow a, c \rightarrow b, c \rightarrow d, c \rightarrow e, ac \rightarrow b,$
$ac \rightarrow d, ac \rightarrow e, d \rightarrow a, e \rightarrow b\}$

Skript: S. 69

Aufgabe 5b *(1/2)*

Zur Bestimmung der minimalen Überdeckung G entfernen wir überflüssige funktionalen Abhängigkeiten mit Lemma 1.1:

Voraussetzung

r: (U | F), $F \subseteq \Phi(U)$; A, B, C, D $\subseteq$ U

Dann gilt:

(1) $B \subseteq A \Rightarrow A \rightarrow B$ (Reflexivität/Projektivität)

(2) $A \rightarrow B \Rightarrow AC \rightarrow BC$ (Erweiterungsregel)

(3) $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C$ (Transitivität)

(4) $A \rightarrow B, A \rightarrow C \Rightarrow A \rightarrow BC$ (Vereinigungsregel)

(5) $A \rightarrow B, BC \rightarrow D \Rightarrow AC \rightarrow D$ (Pseudotransitivität)

(6) $A \rightarrow B, C \subseteq B \Rightarrow A \rightarrow C$ (Zerlegungsregel)

Skript: S. 56

Aufgabe 5b

Folgende funktionalen Abhängigkeiten entfallen:

$c \rightarrow b \Rightarrow ac \rightarrow ab$ (Erweiterungsregel), streiche $ac \rightarrow b$
$c \rightarrow d \Rightarrow ac \rightarrow ad$ (Erweiterungsregel), streiche $ac \rightarrow d$
$c \rightarrow e \Rightarrow ac \rightarrow ae$ (Erweiterungsregel), streiche $ac \rightarrow e$
$c \rightarrow d,\ d \rightarrow a \Rightarrow c \rightarrow a$ (Transitivität), streiche $c \rightarrow a$
$ab \rightarrow c,\ c \rightarrow d \Rightarrow ab \rightarrow d$ (Transitivität), streiche $ab \rightarrow d$
$ab \rightarrow c,\ c \rightarrow e \Rightarrow ab \rightarrow e$ (Transitivität), streiche $ab \rightarrow e$
$c \rightarrow e,\ e \rightarrow b \Rightarrow c \rightarrow b$ (Transitivität), streiche $c \rightarrow b$

Die minimale Überdeckung G ist

$G = \{ ab \rightarrow c,\ c \rightarrow d,\ c \rightarrow e,\ d \rightarrow a,\ e \rightarrow b \}$

Skript: S. 65

Aufgabe 6

1NF

Eine Relation ist in der **Ersten Normalform (1NF)**, wenn die Attributwerte atomar sind.

2NF

Eine 1NF-Relation r ist in der **Zweiten Normalform (2NF)**, wenn jedes Nichtschlüsselattribut von jedem Schlüssel voll funktional abhängig ist

3NF

Eine 1NF-Relation ist in der **Dritten Normalform** (3NF), wenn kein Nichtschlüsselattribut von (irgend)einem Schlüssel transitiv abhängig ist

BCNF

Eine 1NF-Relation ist in der **Boyce-Codd-Normalform** (BCNF), wenn alle elementaren fA‘s gehen von Schlüsseln ausgehen

Skript: S. 81 - 96

Aufgabe 6 (2/3)

r1: Schlüssel a,b, Nichtschlüsselattribute: c,d
Alle elementaren fAs gehen von Schlüsseln aus
➔ BCNF ➔ 3NF ➔ 2NF.

r2: Schlüssel abc, Nichtschlüsselattribut: d
d ist nicht voll funktional abhängig von abc, da ab → d
und ab $\subseteq$ abc
$\neg$ 2NF ➔ $\neg$ 3NF ➔ $\neg$BCNF

r3: Schlüssel a,b,c, keine Nichtschlüsselattribute
Alle elementaren fAs gehen von Schlüsseln aus
➔ BCNF ➔ 3NF ➔ 2NF.

r4: Schlüssel e, Nichtschlüsselattribute: a,b,c,d
e → b (Tranisitivität) ➔ $\neg$ 3NF ➔ $\neg$BCNF
r4 ist in 2NF, da jedes Nichtschlüsselattribut voll funktional abhängig von Schlüssel e ist.

Aufgabe 6 (2/2)

r5: Schlüssel abd, Nichtschlüsselattribute: c,e,f
c ist nicht voll funktional abhängig von abd,
da ab → c und ab $\subseteq$ abd
$\neg$ 2NF ➔ $\neg$ 3NF ➔ $\neg$BCNF

r6: Schlüssel adf, ade, Nichtschlüsselattribute b,c
b ist nicht voll funktional abhängig von ade, da a → b und a $\subseteq$ ade
$\neg$ 2NF ➔ $\neg$ 3NF ➔ $\neg$BCNF

r7: Schlüssel a, f, Nichtschlüsselattribute b,c,d,e
Alle elementaren fAs gehen von Schlüsseln aus
➔ BCNF ➔ 3NF ➔ 2NF.

Aufgabe 7a

	a	b	c	d	e	f	g	h	i	k
r_1	a_1	a_2	a_3	b_{14}	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{10}
r_2	b_{21}	a_2	b_{23}	a_4	b_{25}	b_{26}	b_{27}	b_{28}	b_{29}	b_{20}
r_3	a_1	b_{32}	b_{33}	b_{34}	a_5	a_6	a_7	a_8	b_{39}	b_{30}
r_4	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}	b_{48}	a_9	a_0
$a \rightarrow bc$										
r_1	a_1	a_2	a_3	b_{14}	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{10}
r_2	b_{21}	a_2	b_{23}	a_4	b_{25}	b_{26}	b_{27}	b_{28}	b_{29}	b_{20}
r_3	a_1	a_2	a_3	b_{34}	a_5	a_6	a_7	a_8	b_{39}	b_{30}
r_4	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}	b_{48}	a_9	a_0
$b \rightarrow d$										
r_1	a_1	a_2	a_3	a_4	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{10}
r_2	b_{21}	a_2	b_{23}	a_4	b_{25}	b_{26}	b_{27}	b_{28}	b_{29}	b_{20}
r_3	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	b_{39}	b_{30}
r_4	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}	b_{48}	a_9	a_0

Ersetzte b_{32} und b_{33} durch a_2 und a_3

Ersetzte b_{14} und b_{34} durch a_4

Die Relation e → afgh ändert die Matrix nicht mehr ➔ Keine verlustfreie Zerlegung

Skript: S. 112 - 114

Aufgabe 7b

	a	b	c	d	e	f	g	h	i	k
r_1	a_1	a_2	a_3	a_4	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{10}
r_2	b_{21}	a_2	b_{23}	a_4	b_{25}	b_{26}	b_{27}	b_{28}	b_{29}	b_{20}
r_3	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	b_{39}	b_{30}
r_4	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}	b_{48}	a_9	a_0
r_5	b_{51}	b_{52}	b_{53}	b_{54}	a_5	b_{56}	b_{57}	b_{58}	a_9	b_{50}
$e \rightarrow afgh,\ i \rightarrow k$										
r_1	a_1	a_2	a_3	a_4	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{10}
r_2	b_{21}	a_2	b_{23}	a_4	b_{25}	b_{26}	b_{27}	b_{28}	b_{29}	b_{20}
r_3	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	b_{39}	b_{30}
r_4	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}	b_{48}	a_9	a_0
r_5	a_1	b_{52}	b_{53}	b_{54}	a_5	a_6	a_7	a_8	a_9	a_0
$a \rightarrow bc,\ b \rightarrow d$										
r_1	a_1	a_2	a_3	a_4	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{10}
r_2	b_{21}	a_2	b_{23}	a_4	b_{25}	b_{26}	b_{27}	b_{28}	b_{29}	b_{20}
r_3	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	b_{39}	b_{30}
r_4	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}	b_{48}	a_9	a_0
r_5	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_0

Ergänzung der Zerlegung:

r5 (ei| Ø)

Bei der Ergänzung dürfen keinesfalls neue fA eingeführt werden!

∃ in MAT eine Zeile der Form (a1, a2, …, an)

Zerlegung verlustfrei!

Aufgabe 8

Satz von Delobel

Geg.: r: (U | F), und D: ({ r_1:(A_1 | F_1), r_2:(A_2 | F_2) })
eine Zerlegung von r

Wir setzen: $A_1 \cap A_2 = B$,
$A_1 = AB$, $A_2 = BC$,
mit $A \cap B = A \cap C = B \cap C = \varnothing$

D ist verlustfrei $\Leftrightarrow$ ($B \rightarrow A \in F^+$ oder $B \rightarrow C \in F^+$)

Hier gilt: $U_1 \cap U_2$ = {ah} = B

A = {cdg} ; C = {bef}

D ist verlustfrei, wenn ah → cdg $\in F^+$ oder ah → bef $\in F^+$)

Berechnung von $\{ah\}^{+}$:

$\{ah\}^+$ = {abhgefcd} = U , also gilt ah → cdg $\in F^+$ **und** ah → bef $\in F^+$

➔ Die Zerlegung ist verlustfrei

Skript: S. 105

Aufgabe 9 *(1/2)*

Schlüssel: abg, bdg, bfg, Nichtschlüsselattribute: c, e

e ist nicht voll funktional abhängig von bdg, also folgt ¬ 2NF ➔ ¬ 3NF ➔ ¬BCNF

Synthesealgorithmus:

Schritt 1: Mache alle fA's einfach: F'= {ab → c, ab → d, ab<u>c</u> → e, d → a, d → e, d → f, f → d}

Schritt 2: Entferne alle überflüssigen Attribute der linken Seite:

Ergebnis: G ={ ab → c, ab → d, <u>ab → e</u>, d → a, d → e, d → f, f → d}

Schritt 3: Finde nichtredundante Überdeckung in G

Ergebnis: H ={ab → c, ab → d, d → a, d → a, d → f, f → d}

Schritt 4: fA's mit gleicher linker Seite werden zu einer Klasse zusammengefaßt:

Ergebnis: H1 = {ab → c, ab → d }

H2 = {d → a, d → a , d → f}

H3 = {f → d}

Skript: S. 117 - 122

Aufgabe 9 *(2/2)*

Schritt 5: Konstruiere Relationen

Ergebnis: $\mathfrak{R}$- = {r1: (a,b,c,d | H1), r2: (a,d,e,f | H2), r3: (d,f | H3)}

Schritt 6: Schlüssel K: abg, bfg

Prüfe, ob irgendeine Attributmenge attr(Hi) bereits den Schlüssel der Relation, gebildet aus U und F enthält.

$\mathfrak{R} := \mathfrak{R} - \cup$ r0 : (a,b,g | Ø }

Skript: S. 117 - 122

Aufgabe 10a

Hier gilt: $U_1 \cap U_2$ = {bcd} = B

A = {a} ; C = {e}

Nach Dolobel git: D ist verlustfrei, wenn bcd $\rightarrow$ a $\in F^+$ oder bcd $\rightarrow$ e $\in F^+$)

Es gilt: d $\rightarrow$ e $\in$ F, daraus folgt : bcd $\rightarrow$ e $\in F^+$

➔ Die Zerlegung ist verlustfrei

Aufgabe 10b

	a	b	c	d	e
r_1	a_1	a_2	a_3	a_4	$b_{1,5}$
r_2	$b_{2,1}$	a_2	a_3	a_4	a_5
d → e	**a**	**b**	**c**	**d**	**e**
r_1	a_1	a_2	a_3	a_4	$\mathbf{a_5}$
r_2	$b_{2,1}$	a_2	a_3	a_4	a_5

∃ in der Matrix eine Zeile der Form (a1, a2, …, an) ⇔ Zerlegung verlustfrei!

Aufgabe 11a/b *(1/2)*

Definition: (Zerlegung, fA-erhaltend, verlustfrei)

Geg: Relation r: (U | F);

Datenbank D:($\{r_i: (A_i \mid F_i) \mid i=1, \ldots, n\} \mid \emptyset$)

(1) D ist eine **Zerlegung** von r

:⇔ $\bigcup_i A_i = U$ und $\bigcup_i F_i \subseteq F^+$

Sei D eine Zerlegung von r:

(2) D ist „**fA-erhaltend**" :⇔ $(\bigcup_i F_i^+)^+ = F^+$

Anmerkung:

wegen $(\bigcup_i F_i^+)^+ = (\bigcup_i F_i)^+$ gilt insbesondere:

$\bigcup_i F_i = F \Rightarrow$ D fA-erhaltend

(3) D ist „**verlustfrei**" (auch: „verbundtreu")

:⇔ $r.A_1 * r.A_2 * \ldots r.A_n = r$

Anmerkung: $r.A_i$ ist vom Typ (U_i| F_i)

Aufgabe 11a / b

a) $U_1 \cup U_2 \cup U_3 = U$ und $F_i \subseteq F \Rightarrow$ Zerlegung
$F1 \cup F2 \cup F3 = F \Rightarrow (\bigcup_i F_i)^+ = (\bigcup_i F_i^+)^+ = F^+ \Rightarrow$ fA-erhaltend

b) zu zeigen:
r = r1 * r2 * r3

wegen a) genügt es r1 * r2 * r3 $\subseteq$ r zu zeigen:

sei x = abcde $\in$ r1 * r2 * r3
$\Rightarrow \exists$ x1 = abcde' $\in$ r, $\exists$ x2 = a''bc''d''e $\in$ r, $\exists$ x3 = ab'''c'''d'''e $\in$ r
wobei e', a'', c'', d'', b''', c''', d'''nicht genauer bestimmt sind.
Betrachtet man die funktionalen Abhängigkeiten in r so erhält man

1) b''' = b (wegen $a \to b$)
2) c''' = c, d''' = d (wegen $b \to c$, $b \to d$)

also x3= ab'''c'''d'''e = abcde = x $\Rightarrow$ x $\in$ r

Aufgabe 11c

	a	b	c	d	e
r_1	a_1	a_2	a_3	a_4	$b_{1,5}$
r_2	$b_{2,1}$	a_2	$b_{2,3}$	$b_{2,4}$	a_5
r_3	a_1	$b_{3,2}$	$b_{3,3}$	$b_{3,4}$	a_5

$a \to b$	a	b	c	d	e
r_1	a_1	a_2	a_3	a_4	$b_{1,5}$
r_2	$b_{2,1}$	a_2	$b_{2,3}$	$b_{2,4}$	a_5
r_3	a_1	a_2	$b_{3,3}$	$b_{3,4}$	a_5

$b \to c$, $b \to d$	a	b	c	d	e
r_1	a_1	a_2	a_3	a_4	$b_{1,5}$
r_2	$b_{2,1}$	a_2	a_3	a_4	a_5
r_3	a_1	a_2	a_3	a_4	a_5

$\exists$ in der Matrix eine Zeile der Form (a1, a2, …, an) $\Leftrightarrow$ Zerlegung verlustfrei!

Aufgabe 11d

Schritt 1: Mache alle fA's einfach: F'= $\{a \rightarrow b, b \rightarrow c, b \rightarrow d, e \rightarrow b\}$

Schritt 2: Entferne alle überflüssigen Attribute der linken Seite:

Ergebnis: G $=\{a \rightarrow b, b \rightarrow c, b \rightarrow d, e \rightarrow b\}$

Schritt 3: Finde nichtredundante Überdeckung in G

Ergebnis: H $=\{a \rightarrow b, b \rightarrow c, b \rightarrow d, e \rightarrow b\}$

Schritt 4: fA's mit gleicher linker Seite werden zu einer Klasse zusammengefaßt:

Ergebnis: H1 = $(a \rightarrow b)$, H2 = $(b \rightarrow c, b \rightarrow d)$, H3 = $(e \rightarrow b)$

Schritt 5: Konstruiere Relationen

Ergebnis: $\mathfrak{R}^- = \{$R1 $(ab \mid a \rightarrow b)$, R2 $(bcd \mid b \rightarrow c, b \rightarrow d)$, R3 $(be \mid e \rightarrow b)\}$

Schritt 6: $\mathfrak{R} := \mathfrak{R}^- \cup$ r0 : $(a,e \mid \varnothing)$

Bei dieser Zerlegung sind die resultierenden Relationen zusätzlich in BCNF

Skript: S. 117 - 122

Aufgabe 12

a) Schlüssel: a , NSA: bcd

Alle (der) Schlüssel sind (ist) einelementig, somit sind automatisch alle Nichtschlüsselattribute von den (dem) Schlüssel(n) voll funktional abhängig, somit ist die Relation in der 2NF.

Es existiert folgende transitive Abhängigkeit $a \rightarrow bc \rightarrow d$, somit ist die Relation nicht in der 3NF und nicht in der BCNF

- $b \rightarrow c, bc \rightarrow d \Rightarrow b \rightarrow c, b \rightarrow bc, bc \rightarrow d \Rightarrow b \rightarrow d$, also kann man $b \rightarrow d$ es bleibt $a \rightarrow bc$, $b \rightarrow c$, $bc \rightarrow d$ übrig und keine funktionale Abhängigkeit ist überflüssig

$\Rightarrow$ G $= \{a \rightarrow b, a \rightarrow c, b \rightarrow c, bc \rightarrow d\}$ ist eine minimale Überdeckung

Aufgabe 13

a) Schlüssel: jedes Attribut ist ein Schlüssel NSA: Ø

Die Relation befindet sich in BCNF

b) Schlüssel: abce, acde NSA: ø

Die Relation befindet sich in der 3 NF, aber wegen d $\rightarrow$ b nicht in der BCNF (d ist kein Schlüssel)

c) Schlüssel: abe NSA: c,d

Das Nichtschlüsselattribut c ist nicht voll funktional abhängig vom Schlüssel abe, damit ist die Relation nicht in 2 NF. Folglich befindet sich die Relation auch nicht in 3NF und BCNF

d) Schlüssel: ab NSA: c,d

Jedes Nichtschlüsselattribut ist vom Schlüssel voll funktional abhängig, d.h. die Relation befindet sich in der 2 NF. Wegen der transitiven Abhängigkeit ab $\rightarrow$ c $\rightarrow$ d befindet sich die Relation **nicht** in der 3NF

e) Schlüssel: ab, bc NSA: d

Das Nichtschlüsselattribut d ist nicht voll funktional abhängig vom Schlüssel bc, damit ist die Relation nicht in 2 NF. Folglich befindet sich der Relationstyp auch nicht in 3NF und BCNF