

Prof. Dr. Guido Moerkotte

Email: moer@pi3.informatik.uni-mannheim.de

Pit Fender

B6, 29, Raum C0.05

68131 Mannheim

Telefon: (0621) 181-2517

Email: pfender@pi3.informatik.uni-mannheim.de

Anfrageoptimierung
Herbst-/Wintersemester 07

2. Übungsblatt
26. September 2007

Aufgabe 1

Die folgenden Äquivalenzen gelten für 2-wertige Logik. Zeigen oder widerlegen Sie die Äquivalenzen für 3-wertige Logik, d.h. für alle Variablen x, y, z , die die Werte *true*, *false* oder *unknown* annehmen können gilt:

Aufgabe 1 a)

$$x \wedge \neg x = \textit{false}$$

bzw.

$$x \vee \neg x = \textit{true}$$

Lösung

\Rightarrow beide falsch für $x = \textit{unknown}$:

$$\textit{unknown} \wedge \neg \textit{unknown} = \textit{unknown} = \textit{unknown} \vee \neg \textit{unknown}$$

Aufgabe 1 b)

$$(x \wedge y) \wedge z = x \wedge (y \wedge z)$$

bzw.

$$(x \vee y) \vee z = x \vee (y \vee z)$$

Lösung

beide korrekt.

Verwende eine Wahrheitstabelle. Wir brauchen nur die Ausdrücke zu untersuchen, in denen der Wert *unknown* (U) vorkommt, denn alle anderen Ausdrücke entsprechen der 2-wertigen Logik, für die die Gesetze bekannt sind.

x	y	z	$(x \wedge y) \wedge z$	$x \wedge (y \wedge z)$	$(x \vee y) \vee z$	$x \vee (y \vee z)$
T	T	U	U	U	T	T
T	U	T	U	U	T	T
U	T	T	U	U	T	T
T	U	U	U	U	T	T
U	T	U	U	U	T	T
U	U	T	U	U	T	T
F	F	U	F	F	U	U
F	U	F	F	F	U	U
U	F	F	F	F	U	U
F	U	U	F	F	U	U
U	F	U	F	F	U	U
U	U	F	F	F	U	U
U	U	U	U	U	U	U

Aufgabe 1 c)

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

bzw.

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

Lösung

beide korrekt.

Wahrheitstabelle:

x	y	z	$x \vee (y \wedge z)$	$(x \vee y) \wedge (x \vee z)$	$x \wedge (y \vee z)$	$(x \wedge y) \vee (x \wedge z)$
T	T	U	T	T	T	T
T	U	T	T	T	T	T
U	T	T	T	T	U	U
T	U	U	T	T	U	U
U	T	U	U	U	U	U
U	U	T	U	U	U	U
F	F	U	F	F	F	F
F	U	F	F	F	F	F
U	F	F	U	U	F	F
F	U	U	U	U	F	F
U	F	U	U	U	U	U
U	U	F	U	U	U	U
U	U	U	U	U	U	U

Aufgabe 2

Gegeben seien folgende Relationen inklusive Filterfaktoren:

$$|R_1| = 10, |R_2| = 100, |R_3| = 1000, |R_4| = 25 \text{ und}$$

$f_{1,2} = 0, 1$, $f_{2,3} = 0, 2$, $f_{1,3} = 0, 4$ und $f_{3,4} = 0, 1$

Berechnen Sie die Kostenfunktionen aller Joinreihenfolgen (Kreuzprodukte eingeschlossen) für C_{out} , C_{nlj} , C_{hj} und C_{smj} .

Lösung

Join	Card	C_{out}	C_{nlj}	C_{hj}	C_{smj}
$R_1 \bowtie R_2$	100	100	1000	12	697.6
$R_1 \bowtie R_3$	4000	4000	10000	12	9999.0
$R_1 \times R_4$	250	250	250	250	149.31
$R_2 \bowtie R_1$	100	100	1000	120	697.6
$R_2 \bowtie R_3$	20000	20000	100000	120	10630.16
$R_2 \times R_4$	2500	2500	2500	2500	780.48
$R_3 \bowtie R_1$	4000	4000	10000	1200	9999.0
$R_3 \bowtie R_2$	20000	20000	100000	1200	10630.16
$R_3 \bowtie R_4$	2500	2500	25000	1200	10081.88
$R_4 \times R_1$	250	250	250	250	149.31
$R_4 \times R_2$	2500	2500	2500	2500	780.48
$R_4 \bowtie R_3$	2500	2500	25000	30	10081.88
$(R_1 \bowtie R_2) \bowtie R_3$	8000	8100	101000	132	11327.77
$(R_1 \bowtie R_2) \times R_4$	2500	2600	3500	2500	1478.08
$(R_1 \bowtie R_3) \bowtie R_2$	8000	12000	410000	4812	58526.52
$(R_1 \bowtie R_3) \bowtie R_4$	10000	14000	110000	4812	57978.23
$(R_2 \bowtie R_3) \bowtie R_1$	8000	28000	300000	24120	296417.63
$(R_2 \bowtie R_3) \bowtie R_4$	50000	70000	600000	24120	296500.51
$(R_1 \times R_4) \bowtie R_2$	2500	2750	25250	550	2805.14
$(R_1 \times R_4) \bowtie R_3$	10000	10250	250250	550	12106.54
$(R_2 \times R_4) \bowtie R_1$	2500	5000	27500	5500	29032.98
$(R_2 \times R_4) \bowtie R_3$	50000	52500	2502500	5500	38965.54
$R_1 \bowtie (R_2 \bowtie R_3)$	8000	28000	300000	132	296417.63
$R_1 \bowtie (R_2 \times R_4)$	2500	5000	27500	2512	29032.98
$R_1 \bowtie (R_3 \bowtie R_4)$	10000	12500	50000	42	38334.38
$R_2 \bowtie (R_1 \bowtie R_3)$	8000	12000	410000	132	58526.52
$R_2 \bowtie (R_1 \times R_4)$	2500	2750	25250	370	2805.14
$R_2 \bowtie (R_3 \bowtie R_4)$	50000	52500	275000	150	38965.54
$(R_3 \bowtie R_4) \bowtie R_1$	10000	12500	50000	3030	38334.38
$(R_3 \bowtie R_4) \bowtie R_2$	50000	52500	275000	3030	38965.54
$R_3 \bowtie (R_1 \bowtie R_2)$	8000	8100	101000	1212	11327.77
$R_3 \bowtie (R_1 \times R_4)$	10000	10250	250250	1450	12106.54
$R_3 \bowtie (R_2 \times R_4)$	50000	52500	2502500	3700	38965.54
$R_4 \times (R_1 \bowtie R_2)$	2500	2600	3500	2500	1478.08
$R_4 \bowtie (R_1 \bowtie R_3)$	10000	14000	110000	42	57978.23
$R_4 \bowtie (R_2 \bowtie R_3)$	50000	70000	600000	150	296500.51
$((R_1 \bowtie R_2) \bowtie R_3) \bowtie R_4$	20000	28100	301000	9732	115170.14

$((R_1 \bowtie R_2) \times R_4) \bowtie R_3$	20000	22600	2503500	3370	39663.15
$(R_3 \bowtie (R_1 \times R_4)) \bowtie R_2$	20000	30250	1050000	12042	145648.05
$(R_2 \bowtie (R_3 \bowtie R_4)) \bowtie R_1$	20000	72500	775000	60150	819480.79
$(R_1 \bowtie R_2) \bowtie (R_3 \bowtie R_4)$	20000	22600	276000	162	39663.15
$(R_1 \bowtie R_3) \bowtie (R_2 \times R_4)$	20000	26500	10012500	7312	86861.9
$(R_2 \bowtie R_3) \bowtie (R_1 \times R_4)$	20000	40250	5100250	24370	298525.17
$(R_1 \times R_4) \bowtie (R_2 \bowtie R_3)$	20000	40250	5100250	670	298525.17
$(R_2 \times R_4) \bowtie (R_1 \bowtie R_3)$	20000	26500	10012500	5512	86861.9
$R_1 \bowtie (R_2 \bowtie (R_3 \bowtie R_4))$	20000	72500	775000	162	819480.79
$R_2 \bowtie (R_3 \bowtie (R_1 \times R_4))$	20000	30250	1050000	162	145648.05
$(R_3 \bowtie R_4) \bowtie (R_1 \bowtie R_2)$	20000	22600	276000	3042	39663.15
$R_3 \bowtie ((R_1 \bowtie R_2) \times R_4)$	20000	22600	2503500	1570	39663.15
$R_4 \bowtie ((R_1 \bowtie R_2) \bowtie R_3)$	20000	28100	301000	162	115170.14

Aufgabe 3

In der Vorlesung wurde angesprochen, daß das Pushen von Selektionen und die Joinreihenfolge sich gegenseitig beeinflussen. Wie läßt sich das Problem lösen? Zeigen Sie an dem Beispiel aus der Vorlesung, wie Sie das Problem mit einem modifizierten Algorithmus lösen können.

Lösung

Joinreihenfolge und Pushen von Selektionen müssen integriert werden:

1. Erstelle eine Liste aus Tabellen (inklusive der Selektionen referenzierender Nicht-Join-Prädikate):

$$\{\sigma_{p.PName='Larson'}(Professor[p]), Lecture[l], Attend[a], Student[s]\}$$

2. Erstelle eine Liste mit Join-Prädikaten:

$$\{s.SNo = a.ASNo, a.ALNo = l.LNo, l.LPNo = p.PNo\}$$

3. Solange noch Join-Prädikate übrig sind, wähle ein Join-Prädikat aus, daß eine bereits gewählte Tabelle aus (1) referenziert und füge den Join in den Plan ein.

- I. $\sigma_{p.PName='Larson'}(Professor[p])$

- II. $(\dots \bowtie_{p.PNo=l.LPNo} Lecture[l])$

- III. $((\dots) \bowtie_{l.LNo=a.ALNo} Attend[a])$

- IV. $((\dots)) \bowtie_{a.ASNo=s.SNo} Student[s]$

4. Füge die verbleibenden Tabellen aus (1) mittels Kreuzprodukte in den Plan ein

Aufgabe 4

Erstellen Sie ein Programm, das Eingaben der Form

```
from
table_1 t1
table_2 t2
...
table_n tn
where
  ti.aj = tk.al
...
;
```

in einen kanonischen Ausführungsplan umwandelt und anschließend ausführt. Gehen Sie davon aus, daß die Eingabe korrekt vorliegt (ein Eintrag pro Zeile etc.). Wenn Ihnen die Variablenbindung zu komplex ist, akzeptieren Sie Anfragen der Form

```
from
table_1
table_2
where
  ai = aj
...
;
```

Lösung

siehe blatt2.zip