

## Serie 2

### Anfrageoptimierung 2007/08

1. Geben Sie analog zur Aufgabe 1 aus Serie 1 eine Implementation des Hash-Joins  $R1 \text{ hjoin}_{A=B} R2$  an, bei der der Hash-Zugriff auf  $R2$  beschränkt ist. Eine Funktion `open-hash-index-scan (IndexId,Wert) → ScanID` ist zu benutzen.

2. Die folgenden Äquivalenzen gelten für eine zweiwertige Logik. Zeigen oder widerlegen Sie sie für eine dreiwertige Logik.

a)  $a \vee \neg a = \text{true}$

b)  $a \wedge \neg a = \text{false}$

c)  $(a \vee b) \wedge c = (a \wedge c) \vee (b \wedge c)$

d)  $(a \wedge b) \vee c = (a \vee c) \wedge (b \vee c)$

e)  $\neg(a \wedge b) = \neg a \vee \neg b$

f)  $\neg(a \vee b) = \neg a \wedge \neg b$

3. Wenden Sie den in der Vorlesung angegebenen Algorithmus zur logischen Optimierung auf das folgende Beispiel an:

```
Select distinct s.SName
```

```
From Students s, Attend a, Lecture l
```

```
Where s.SNO=A.ANo and a.ALNo=l.LNo and l.Ltitle="DB1" and s.Age >25
```

4. Wenden Sie den in der Vorlesung angegebenen Algorithmus zur logischen Optimierung exakt auf das folgende Beispiel an. Wie muss der Algorithmus verbessert werden, damit bessere Ergebnisse erzielt werden können?

```
Select distinct l.LTitle
```

```
From Lecture l, Student s, Attend a
```

```
Where s.SNo=a.ASno and a.ALNo=l.LNo and s.SName="Meier"
```