

Übung zur Vorlesung *Datenbanksysteme I* im SS 20

Gabriel Haas (gabriel.haas@uni-jena.de)

dbis1.github.io/courses/ss20/db1/

Blatt Nr. 10

Abgabe auf moodle.uni-jena.de. Die Bedingungen für den Notenbonus finden Sie auf der Vorlesungswebsite.

Hausaufgabe 1

Gegeben sei die folgende SQL-Anfrage:

```
select p.Name, v.Titel, vi.anzahlHoerer
from Vorlesungen v,
  ( select vi.VorlNr, count(*) as anzahlHoerer
    from hoeren h join Vorlesungen vi on h.VorlNr = vi.VorlNr
    where vi.sws > 2
    group by vi.VorlNr
    having count(*) > 1) vi,
  Professoren p
where v.VorlNr = vi.VorlNr and p.PersNr = v.gelesenVon
```

Geben Sie die kanonische Übersetzung dieser Anfrage in die relationale Algebra an. Verwenden Sie zur Darstellung des relationalen Algebraausdrucks die Baumdarstellung.

Optimieren Sie Ihren relationalen Algebraausdruck logisch. Gehen Sie dabei von **realistischen** Kardinalitäten (für eine sehr große Universität) für die relevanten Relationen aus. Verwenden Sie hierfür die aus der Vorlesung bekannten Optimierungstechniken.

Hausaufgabe 2

Gegeben sind die beiden Relationenausprägungen:

R	
	A
...	0
...	5
...	7
...	8
...	8
...	10
⋮	⋮

S	
B	
5	...
6	...
7	...
8	...
8	...
11	...
⋮	⋮

Werten Sie den Join $R \bowtie_{R.A=S.B} S$ mithilfe des Nested-Loop- sowie des Sort/Merge-Algorithmus aus. Machen Sie deutlich, in welcher Reihenfolge die Tupel der beiden Relationen verglichen werden und kennzeichnen Sie die Tupel, die in die Ergebnismenge übernommen werden. Vervollständigen Sie hierzu die beiden folgenden Tabellen:

		S.B					
		5	6	7	8	8	11
R.A	0						
	5						
	7						
	8						
	8						
	10						

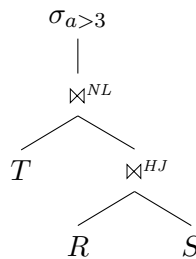
Nested-Loop-Join

		S.B					
		5	6	7	8	8	11
R.A	0						
	5	2✓					
	7						
	8						
	8						
	10						

Sort/Merge-Join

Hausaufgabe 3

Gegeben Sei der folgende physische Anfrageplan:



Hier steht \bowtie^{NL} für den Nested-Loop-Join und \bowtie^{HJ} für den Hash-Join. Geben Sie an, wie oft die *next*-Funktion von *R* aufgerufen wird, wenn ein Datenbanksystem diesen Anfrageplan mit dem Iteratorkonzept ausführt. Gehen Sie von folgenden Kardinalitäten aus: $|R| = 10$, $|S| = 20$ und $|T| = 5$. Gehen Sie davon aus, dass beim Nested-Loop-Join die linke Eingabe zuerst geöffnet wird und dass beim Hash-Join aus der linken Eingabe eine Hashtabelle erzeugt wird.