

Serie 4 Anfrageoptimierung 2007/08

1. Zeigen Sie, dass für die Anzahl $f(n)$ der bushy join trees für Chain Queries (kein kartesisches Produkt)

gilt: $f(n) = 2^{n-1} C(n-1)$, wobei C die Catalanzahl angibt.

2. Gegeben sei folgendes abstraktes relationales Schema:

$R_1(a, c, d, e)$

$R_2(a, b)$

$R_3(b, c)$

$R_4(d)$

$R_5(e)$

Geben Sie für die folgenden Anfragen den Query Graphen an. Klassifizieren Sie den Query Graphen entsprechend der Notation aus der Vorlesung. Wie groß ist jeweils der Suchraum für diese Klasse von Anfragen, wenn ein left-deep bzw. bushy Join-Tree ohne Kreuzprodukte berücksichtigt wird? Existieren effiziente Algorithmen für die Bestimmung der Join-Reihenfolge der Anfragetypen?

Aufgabe 2 a)

```
SELECT DISTINCT *  
FROM R1, R2, R3, R4  
WHERE R1.c = R3.c AND R2.b = R3.b AND R1.d = R4.d;
```

Aufgabe 2 b)

```
SELECT DISTINCT *  
FROM R1, R2, R3, R5  
WHERE R1.a = R2.a AND R1.c = R3.c AND R1.e = R5.e AND R2.b = R3.b;
```

Aufgabe 2 c)

```
SELECT DISTINCT *  
FROM R1, R2, R3, R4, R5  
WHERE R1.a = R2.a AND R1.c = R3.c AND R1.d = R4.d AND R1.e = R5.e;
```

3. Erstellen Sie eine Tabelle, in der Sie Join-Ordering-Algorithmen (Greedy Heuristiken) klassifizieren können. D.h. die Tabelle sollte eine Spalte für den Namen des Algorithmus und je eine Spalte für jedes Merkmal eines Join-Ordering-Problems enthalten.