

# Datenbanken II

## Übungsblatt 7 – WiSe 2016/17

24. Die Relation  $E$  hat  $n_E = 5000$  Tupel auf  $b_E = 2000$  Blöcken, die Relation  $D$  hat  $n_D = 50$  Tupel auf  $b_D = 10$  Blöcken. Es stehen  $M = 6$  Blöcke im Puffer zur Verfügung. Es wird angenommen, dass es nur wenige Duplikate in den Join-Attributen gibt.  
Berechnen Sie die Block-Zugriffe für einen Block Nested Loop Join  $E \bowtie D$ ,  $k = 1$  Block für  $D$ .
25. Angabe, bis auf die Join-Strategie, wie in Aufgabe 24.  
Berechnen Sie die Block-Zugriffe für einen Block Nested Loop Join  $E \bowtie D$ ,  $k = 5$  Blöcke für  $D$ .
26. Angabe, bis auf die Join-Strategie, wie in Aufgabe 24.  
Berechnen Sie die Block-Zugriffe für einen Block Nested Loop Join  $D \bowtie E$ ,  $k = 5$  Blöcke für  $E$ .
27. Angabe, bis auf die Join-Strategie, wie in Aufgabe 24.  
Berechnen Sie die Block-Zugriffe für einen Hash Join. Geben Sie außerdem an, welche Relation als Probing bzw. Building Input verwendet wird.
28. Angabe, bis auf die Join-Strategie, wie in Aufgabe 24.  
Berechnen Sie die Block-Zugriffe für einen Merge Join mit bzw. ohne Clustered  $B^+$ -Baum Index auf die Join-Attribute.
29. Angabe, bis auf die Join-Strategie, wie in Aufgabe 24.  
Berechnen Sie die Block-Zugriffe für einen Index Nested Loop Join, wenn  $E$  als Hash Datei organisiert ist und der Suchschlüssel den Join-Attributen entspricht. Die verwendete Hashfunktion bildet die Suchschlüssel auf 256 Hashwerte ab.
30. Der Join  $R \bowtie S$ ,  $R(A)$ ,  $S(A)$ , soll als Hash Join ausgeführt werden, wobei  $R = \{1, 6, 11, 18, 25, 31, 28\}$ ,  $S = \{1, 3, 6, 7, 9, 11, 23, 25, 27, 30, 8, 19, 17\}$ . Es stehen  $M = 3$  Blöcke im Puffer zur Verfügung, jeder Block fasst 1 Tupel.
- Welche Relation wird als Build Input verwendet?
  - Verwenden Sie zum Erstellen der Partitionen eine Hashfunktion der Form  $h(x) = x \bmod c$  und bestimmen Sie einen geeigneten Wert für  $c$ .
  - Führen Sie den Join aus. Für den Hauptspeicher-Hash-Index wird der 5-stellige Binärwert als Hashfunktion und erweiterbares Hashing verwendet. Nehmen Sie an, dass das Verzeichnis nicht im Puffer gespeichert werden muss und leere Buckets keinen Platz brauchen.