Aufgabenblatt 4

— Anfrageausführung —

Ausgabe am 09.06.2008
Abgabe bis 23.06.2008, 13.00 Uhr

Aufgabe 1: Nested-Loop Join (9 P)

Seien S und R zwei Relationen mit $B(S) = B(R) = 10.000$. Der verfügbare Hauptspeicher umfasst $M = 1.001$ Blöcke.

(a) Berechnen Sie die Anzahl an I/O-Operationen für die Berechnung von $R \bowtie S$, wenn zur Berechnung der Nested-Loop Join-Algorithmus verwendet wird. **(3 P)**

(b) Wie groß muss M mindestens sein, wenn $R \bowtie S$ unter Verwendung des Nested-Loop Join-Algorithmus mit nicht mehr als 100.000 I/O-Operationen berechnet werden soll? **(3 P)**

(c) Wie groß muss M mindestens sein, wenn $R \bowtie S$ unter Verwendung des Nested-Loop Join-Algorithmus mit nicht mehr als 25.000 I/O-Operationen berechnet werden soll? **(3 P)**

Aufgabe 2: Sort Join (12 P)

Betrachten Sie erneut das Beispiel auf Folie 49 mit $B(R) = 1000$, $B(S) = 500$ und $M = 101$. Es wurde erwähnt, dass zusätzliche I/O-Operationen anfallen, wenn alle Tupel aus R und S, die einen bestimmten Y-Wert besitzen, nicht mehr gleichzeitig im Hauptspeicher gehalten werden können.

Ermitteln Sie die Anzahl an I/O-Operationen für die Bestimmung von $R \bowtie$ S unter Verwendung des (einfachen) Sort Join-Algorithmus, wenn

(a) es nur zwei unterschiedliche Y-Werte gibt, wobei jeder Wert in der Hälfte der Tupel aus R und der Hälfte der Tupel aus S auftritt. **(4 P)**

(b) es nur fünf unterschiedliche Y-Werte gibt, wobei jeder Wert in einem Fünftel der Tupel aus R und einem Fünftel der Tupel aus S auftritt. **(4 P)**

(c) es nur zehn unterschiedliche Y-Werte gibt, wobei jeder Wert in einem Zehntel der Tupel aus R und einem Zehntel der Tupel aus S auftritt. **(4 P)**

Aufgabe 3: Hash Join (12 P)

Nehmen Sie an, dass für das Bewegen des Schreib-Lesekopfes auf einen beliebigen Block durchschnittlich 100 ms und für das Lesen bzw. Schreiben eines Blocks 0.5 ms benötigt werden. Somit beträgt die Zeit für das Übertragen (in beide Richtungen) eines beliebigen Blocks 100.5 ms und für das Übertragen von k aufeinanderfolgenden Blöcken $100 + 0.5k$ ms.

Sei $B(R) = 1000$, $B(S) = 500$ und $M = 101$. Es soll ein Hash-basierter Join-Algorithmus für die Berechnung von $R \bowtie S$ verwendet werden, wobei die Bucketanzahl so gering wie möglich sein soll. Nehmen Sie hierbei an, dass die Tupel gleichmäßig auf die Buckets verteilt sind.

(a) Wie lange dauert die Berechnung von $R \bowtie S$, wenn ein zweiphasiger Hash Join-Algorithmus verwendet wird? **(4 P)**

(b) Wie lange dauert die Berechnung von $R \bowtie S$, wenn ein hybrider Hash Join-Algorithmus verwendet wird? **(4 P)**

(c) Wie lange dauert die Berechnung von $R \bowtie S$, wenn ein Sort Join-Algorithmus verwendet wird und sortierte Teillisten auf aufeinanderfolgenden Blöcken der Festplatte gespeichert werden. **(4 P)**

Aufgabe 4: Cluster und Nicht-Cluster Indexe (9 P)

Sei R eine Relation mit $B(R) = 10.000$ und $T(R) = 500.000$. Sei $\mathcal{I}$ ein Index auf dem Attribut a der Relation R und sei $V(R, a) = k$ für ein beliebiges, aber festes $k \in \mathbb{N}$.

Berechnen Sie die Anzahl der I/O-Operationen für die Bestimmung von $\sigma_{a=0}(R)$ als Funktion von k, unter folgenden Nebenbedingungen. Vernachlässigen Sie dabei die I/O-Kosten, die sich durch den Indexzugriff ergeben.

(a) $\mathcal{I}$ ist ein Cluster-Index. **(3 P)**

(b) $\mathcal{I}$ ist ein Nicht-Cluster Index. **(3 P)**

(c) R ist eine *clustered relation* und $\mathcal{I}$ wird nicht verwendet. **(3 P)**

Aufgabe 5: Index-Scan versus Full Table-Scan (5 P)

Betrachten Sie erneut die Relation R aus Aufgabe 4. Nehmen Sie nun an, dass R eine *clustered relation* ist und der Index $\mathcal{I}$ auf dem Attribut a ein Nicht-Cluster Index ist. Dann kann es für die Berechnung von $\sigma_{a=0}(R)$ abhängig vom Wert des Parameters k (aus Aufgabe 4) günstig sein entweder einen Full Table-Scan oder einen Index-Scan durchzuführen.

Berechnen Sie die Werte von k, bei denen die Nutzung des Index-Scan vorzuziehen ist, weil dadurch weniger I/O-Operationen im Vergleich zum Full Table-Scan anfallen.