UNIVERSITÄT DES SAARLANDES
Fachbereich 14 – Informatik
Prof. Dr.-Ing. G. Weikum
Dipl.-Inform. R. Schenkel

Datenbanksysteme WS 1999/2000

Zweite Teilklausur 17.01.2000

Schreiben Sie bitte auf **jedes** Blatt, das Sie abgeben, deutlich Ihren Namen, Ihre Matrikelnummer und Ihre Übungsgruppe. Sie können zur Bearbeitung der Klausur beliebige Unterlagen benutzen. Die Klausur besteht aus drei Teilaufgaben, benutzen Sie bitte für jede Teilaufgabe ein eigenes Blatt. Kreuzen Sie auf dem beiliegenden Deckblatt an, zu welchen Aufgaben Sie Lösungen abgeben, und geben Sie das Deckblatt mit ab. Sie haben zur Bearbeitung aller Aufgaben insgesamt 90 Minuten Zeit.

Anwendungsszenario

Für alle Teilaufgaben der Klausur ist eine simplifizierte Lizenzdatenbank einer Softwarefirma mit den beiden Relationen `Produkte` und `Lizenzen` vorgegeben. Die Relation `Produkte` enthält Informationen über alle Softwareprodukte der Firma. Ihr Primärschlüssel ist zusammengesetzt aus `Bezeichnung` und `Sprache`. Außerdem enthält sie das Freigabedatum jedes Produktes. In der Relation `Lizenzen` werden alle verkauften Lizenzen verwaltet, ihr Primärschlüssel ist `Lizenznr`.

Eine mögliche Beispielausprägung könnte wie folgt aussehen:

Produkte

Bezeichnung	Sprache	Freigabedatum	Endkundenpreis
Windows 2000 Professional	Deutsch	24. Februar 2000	DM 499,99
Windows 2000 Professional	Englisch	17. Februar 2000	$ 149.99
Office 2000 Enterprise Edition	Deutsch	24. Juni 1999	DM 1.299,99
Internet Explorer 2000	Portugiesisch	1. April 2000	DM 0,00
Windows NT 4.0 Workstation	Deutsch	8. Januar 1996	DM 449,00

Lizenzen

Lizenznr	Bezeichnung	Sprache	Kaufdatum	Endkundennummer
D-1123	Office 2000 Enterprise Edition	Deutsch	25. Juni 1999	1742
D-1704	Windows NT 4.0 Workstation	Deutsch	19. Mai 1996	1861
D-2001	Office 2000 Enterprise Edition	Deutsch	7. August 1999	1861

Es gebe 10.000 verschiedene Tupel in der Relation Produkte, und zwar 200 verschiedene Bezeichnungen in jeweils 50 verschiedenen Sprachen.

In der Relation Lizenzen gebe es 10.000.000 Lizenzen, und zwar – zur Vereinfachung – genau 1.000 für jedes der insgesamt 10.000 Softwareprodukte.

Alle Tupel beider Relationen seien konstant 300 Bytes lang. Alle Werte des Attributs `Bezeichnung` sollen die konstante Länge 150 Bytes haben, alle Werte des Attributs `Sprache` die konstante Länge 50 Bytes. Alle TIDs (RowIDs) sowie Sohnzeiger in B*-Bäumen sollen 6 Bytes lang sein.

Die Seitengröße der Datenbank sei 4096 Bytes mit einem Header von 96 Bytes Verwaltungsinformation, und alle Seiten seien zu 75 Prozent gefüllt.

Aufgabe 1: Indexgrößen (1+1=2 Punkte)

Berechnen Sie die Anzahl der Blattseiten, den Fanout und die Baumhöhe für jede der beiden folgenden mittelbaren, unclustered B*-Baum-Indexstrukturen. Tragen Sie Ihre Ergebnisse in die Tabelle auf dem Deckblatt ein. (Die notwendigen Rechnungen, die Sie zu diesen Ergebnissen geführt haben, müssen Sie natürlich ebenfalls abgeben.)

a) Ein Index *LB* über das Attribut `Bezeichnung` von Lizenzen, und
b) ein Join-Index *J* über die Konkatenation der Attribute `Bezeichnung` und `Sprache` beider Relationen.

Zusätzlich sind ein Index *LS* über das Attribut `Sprache` der Relation Lizenzen, ein Index *LBS* über die Konkatenation der Attribute `Bezeichnung` und `Sprache` von Lizenzen und ein Index *PBS* über die Konkatenation der Attribute `Bezeichnung` und `Sprache` von Produkte vorhanden. (Sie benötigen diese Indizes nicht für diese Aufgabe, aber eventuell in Aufgabe 2.) Sie können die Anzahl der Blattseiten und die Höhe dieser Indizes der Tabelle entnehmen.

Hinweis: Möglicherweise passen die TIDs, die zu einem Schlüssel gehören, nicht immer in eine einzige Blattseite. Die TIDs werden dann über mehrere Blattseiten verteilt, die durch Zeiger im Seitenheader (also außerhalb der Nutzdaten) verkettet werden. Nehmen Sie zur Vereinfachung an, daß eine Blattseite in diesem Fall exklusiv für die TIDs genutzt wird, die zu einem Schlüssel gehören, auch wenn dadurch viel Platz darin frei bleibt. Die Einträge in der Index-Ebene über den Blättern referenzieren nur die erste Seite einer solchen seitenübergreifenden TID-Liste.

Aufgabe 2: Ausführungspläne (1+2+2=5 Punkte)

Nehmen Sie an, alle fünf in Aufgabe 1 gegebenen Indexstrukturen seien vorhanden. Bestimmen Sie die kostengünstigsten Ausführungspläne (und ihre Kosten) für die folgenden Anfragen. Das verwendete Datenbanksystem unterstützt lediglich Nested Loop, Nested Loop mit Indexzugriff, Sort-Merge, Index-Scan und Relationen-Scan. Der für das Sortieren zur Verfügung stehende Hauptspeicher beträgt 50 MByte, der Datenbank-Seitenpuffer ist so klein, dass Caching-Effekte vernachlässigbar sind.

a)
```
SELECT * FROM Lizenzen WHERE Bezeichnung=“Windows NT 4.0 Workstation“
AND Sprache=“Deutsch“
```
b)
```
SELECT * FROM Lizenzen L, Produkte P
WHERE L.Bezeichnung=P.Bezeichnung AND L.Sprache=P.Sprache
```
c)
```
SELECT Endkundennummer,Lizenznummer,L.Bezeichnung
FROM Lizenzen L, Produkte P
WHERE L.Bezeichnung = P.Bezeichnung AND L.Sprache=P.Sprache
AND P.Freigabedatum>=“1. Dezember 1999“
```

Der Anteil der ab dem 1. Dezember 1999 freigegebenen Softwareprodukte sei dabei 0,1%.

Begründen Sie jeweils die Wahl des kostengünstigsten Ausführungsplans. Berechnen Sie dazu ggf. die Kosten alternativer Pläne oder geben Sie dafür untere Schranken an.

Aufgabe 3: Algebraische Optimierung (3 Punkte)

Nehmen Sie an, es gelte die folgende Integritätsbedingung, die sicherstellt, dass Softwareprodukte erst ab ihrer Freigabe verkauft werden:

```
CREATE ASSERTION KaufFreigabeConstraint CHECK
(Lizenzen.Kaufdatum>=ALL(SELECT P.Freigabedatum FROM Produkte P WHERE
P.Bezeichnung=Lizenzen.Bezeichnung AND P.Sprache=Lizenzen.Sprache)
```

Zusätzlich gebe es einen Index über das Attribut `Kaufdatum` der Relation `Lizenzen`. Betrachten Sie nun erneut die folgende SQL-Anfrage aus Aufgabe 2c:

```
SELECT Endkundennummer,Lizenznummer,Bezeichnung FROM Lizenzen L,
Produkte P WHERE P.Freigabedatum>=“1. Dezember 1999“ AND
L.Bezeichnung=P.Bezeichnung AND L.Sprache=P.Sprache
```

bzw. den daraus erzeugten Algebraausdruck

```
Π[Endkundennummer,Lizenznummer,Bezeichnung](σ[Freigabedatum>="1. Dezember
1999"](Produkte |x| Lizenzen))
```

a) Formulieren Sie die Integritätsbedingung als Ausdruck der Relationenalgebra.

b) Optimieren Sie die als Algebraausdruck dargestellte Anfrage mit Hilfe der in der folgenden Liste enthaltenen algebraischen Äquivalenzregeln, so dass der Join auf möglichst kleinen Operanden arbeitet und die vorhandenen Indizes ausgenutzt werden. Nutzen Sie dabei auch die Integritätsbedingung aus. Geben Sie für jeden Transformationsschritt an, welche Äquivalenzregel Sie anwenden.

Anhang: Liste von relationenalgebraischen Äquivalenzregeln

Kommutativitätsregeln:

(R1) $\Pi[R1]\ \sigma[P]\ (R) = \sigma[P]\ \Pi[R1]\ (R)$, falls P nur R1-Attribute enthält

(R2) $R \mid x \mid S = S \mid x \mid P$

Idempotenzregeln:

(R4) $\Pi[R1](\ \Pi[R2]\ (R)\ ) = \Pi[R1]\ (R)$, falls $R1 \subseteq R2$

(R5) $\sigma[P1](\ \sigma[P2]\ (R)\ ) = \sigma[P1 \wedge P2]$

Distributivitätsregeln:

(R8) $\sigma[P1](\ R \mid x \mid S) = \sigma[P1](R) \mid x \mid S$, falls P1 nur R-Attribute enthält

(R9) $\Pi[R1,S1](\ R \mid x \mid [P]\ S) = \Pi[R1](R) \mid x \mid [P]\ \Pi[S1](S)$, falls P-Attribute $\subseteq R1 \cup S1$

Weitere Regel:

(R12) $R \mid x \mid R = R$

2. Teilklausur Datenbanksysteme WS 1999/2000

Deckblatt – unbedingt mit abgeben!

Name: ..... Matrikelnummer: .....

Kreuzen Sie bitte jede Aufgabe an, für die Sie eine Lösung abgeben:

Aufgabe 1	Aufgabe 2	Aufgabe 3

Wertetabelle für Aufgabe 1

Index	**Anzahl Schlüssel**	**Anzahl Blätter**	**Fanout**	**Höhe**
LS	50	20.050	53	2
LB				
LBS	10.000	30.000	14	5
PBS	10.000	715	14	4
J				