



Bitte hier unbedingt Matrikelnummer und Adresse eintragen, sonst keine Bearbeitung möglich.

Seven empty boxes for entering the matriculation number.

Postanschrift: FernUniversität D - 58084 Hagen

Name, Vorname

Straße, Nr.

PLZ, Wohnort

FERNUNIVERSITÄT in Hagen

EINGANG

INF

Bitte zurück an:

FERNUNIVERSITÄT in Hagen

D-58084 Hagen

Fakultät für Mathematik und Informatik

Kurs: 01672 „Datenbanken II“

Leistungsnachweisklausur am 03. September 2016

Klausurort:

- Checkboxes for locations: Berlin, Bochum, Frankfurt, Hamburg, Karlsruhe, Köln, München, Bregenz, Wien, Zürich, Dortmund.

Status:

- Checkboxes for status: Vollzeitstudent, Teilzeitstudent, Zweithörer, Akademie-Studierende/r.

Table with columns for Aufgabe, 1, 2, 3, 4, 5, Gesamt and rows for bearbeitet, erreichte Punktzahl.

Datum: _____

Korrektur: _____

Aufgabe 1: Serialisierbarkeit**(12 Punkte)**

Gegeben sind die im Folgenden beschriebenen Abarbeitungs-Schedules der Transaktionen T1, T2 und T3:¹

- a) S₁: r1(C); r3(B); r1(A); r2(B); w1(A); w2(B); r3(A); r2(C);
 w3(A); w2(C); r3(C); w3(C);
- b) S₂: r2(A); r3(C); w2(A); w3(C); r3(A); r2(B);
 r1(C); w2(B); r3(B); w1(C); w3(A); w3(B); r1(B);

Sind die Transaktionsabläufe serialisierbar? Zeichnen Sie jeweils den zugehörigen Abhängigkeitsgraphen. Falls Serialisierbarkeit nicht gegeben ist, benennen Sie die in Konflikt stehenden Operationen, falls Serialisierbarkeit gegeben ist, geben Sie eine serielle Schedule an (Reihenfolge der Transaktionen). (jeweils 4 Punkte)

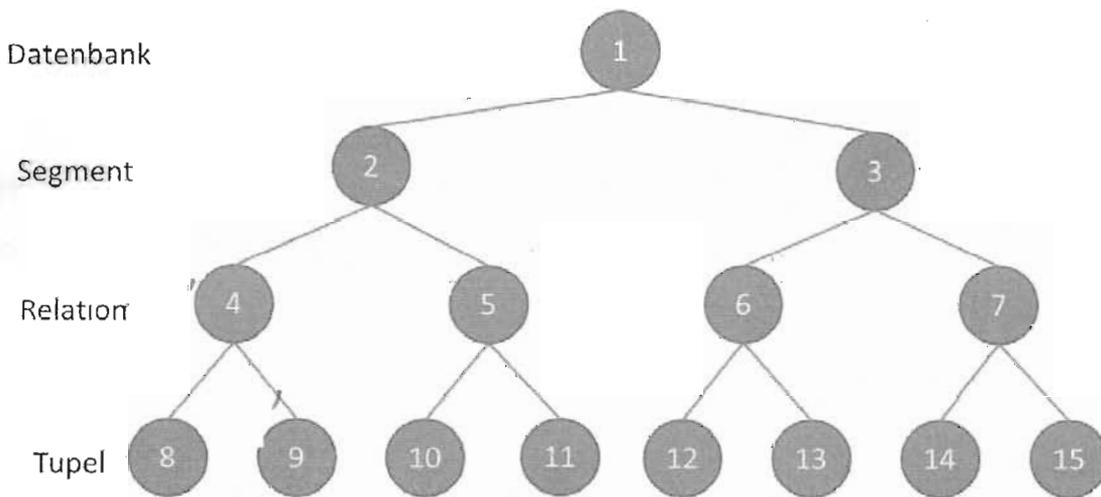
- c) Falls eine der beiden Schedules nicht serialisierbar ist, versuchen Sie, die Schedule serialisierbar zu machen, indem Sie genau eine Operation in der Schedule verschieben. (4 Punkte)

¹ r1(C) bedeutet: Transaktion T1 liest das Objekt C; w2(B) bedeutet: Transaktion T2 schreibt das Objekt B

Aufgabe 2: Hierarchische Sperren**(17 Punkte)**

Gegeben sei eine Hierarchie aus 4 Ebenen:

- Datenbank
- Segment
- Relation
- Tupel

Wir betrachten in dieser Hierarchie die folgenden Knoten von 1 – 16.⁷⁵

- a) Wir betrachten lediglich Schreib- und Lese-Sperren (W und R). Welche weiteren Sperren werden für das hierarchische Sperrverfahren benötigt? Erstellen Sie eine Kompatibilitätsmatrix für die Sperren. (5 Punkte)

Beschreiben Sie, wie die Transaktionen T1, T2 und T3 vorgehen müssen, um die folgenden Schreib- oder Lesesperren zu bekommen. Die Sperranforderungen sind als Sequenz zu sehen, d.h., Transaktion T1 könnte z.B. in d) in Konflikt mit Transaktion T2 kommen aufgrund der Sperren, die T2 in c) bekommen hat.

- b) T1 möchte das Tupel 8 mit einer Schreibsperre (W) versehen.
 c) T2 möchte das Tupel 10 zum Lesen mit einer (R)-Sperre besetzen.
 d) T1 möchte die Relation 5 zum Schreiben sperren.
 e) T3 möchte das Tupel 9 zum Schreiben sperren.
 f) T1 möchte die ganze Relation 6 zum Schreiben sperren.
 g) T2 möchte das Tupel 12 zum Lesen sperren.

(jeweils 2 Punkte pro Teilaufgabe)

Notieren Sie die jeweiligen Schritte in Form von Tripeln: $(T_i, K_j, \text{lock_type})$. Dies bedeutet, dass die Transaktion T_i auf dem Knoten K_j eine Sperre vom Typ „lock_type“ setzt. Falls eine in den Aufgaben genannte Sperre nicht gewährt werden kann, notieren Sie die Schritte bis zu dem Punkt, an dem das Vergabeverfahren den Konflikt bemerkt.

Aufgabe 4: Sperren**(21 Punkte)**

a) (6 Punkte)

Gegeben sind die Transaktionen T_1 , T_2 und T_3 . Im Folgenden sind einige Sperrabfolgen dieser Transaktionen verzeichnet. $L_1(A)$ heißt, dass Transaktion T_1 eine Sperre auf A setzt (Lock), $U_1(A)$ heißt, dass Transaktion T_1 die Sperre auf A wieder freigibt (Unlock). Wir betrachten der Einfachheit halber nur exklusive Sperren, verlangen aber die Einhaltung des Zweiphasen-Sperrprotokolls.

Alle diese Sperrabfolgen beinhalten nicht korrekte Sperranforderungen (d.h. es werden Sperren gesetzt, die nicht gesetzt werden dürften). Analysieren Sie die Sperrfolgen und zeigen Sie auf, wo Konflikte existieren.

1. $L_2(A); L_3(B); L_3(D); L_2(B); U_3(D); L_2(C); L_1(D); U_2(A); U_1(D); U_3(B); U_2(C); U_2(B)$.
2. $L_2(A); L_2(B); L_3(C); U_2(B); L_1(D); U_2(A); L_3(E); L_1(A); U_3(C); L_1(C); L_1(E); U_1(C); U_3(E); U_1(D); U_1(A); U_1(E)$.
3. $L_3(B); L_2(A); L_3(C); L_1(D); L_2(B); L_1(E); U_1(D); L_2(C); L_3(A); U_3(B); U_2(A); U_1(E); U_3(C); U_2(B); U_2(C); U_3(A)$.

b) (15 Punkte)

Nun betrachten wir den Fall, dass ein Sperr-Manager die Vergabe und die Freigabe von Sperren überwacht:

- Wenn eine Sperre auf ein Objekt angefordert wird und dieses Objekt ist bereits mit einer Sperre belegt, so wird die anfordernde Transaktion auf inaktiv gesetzt und als letztes Element in die Warteschlange für dieses Objekt eingetragen. Das heißt, alle Folgeaktionen dieser Transaktion können nicht mehr ausgeführt werden.
- Wenn eine Sperre freigegeben wird, so prüft der Sperrmanager zunächst, ob es ein Element in der Warteschlange gibt. Wenn das der Fall ist, dann wird diese Transaktion aufgeweckt, sie bekommt die Sperre, und kann anschließend alle noch nicht durchgeführten Aktionen ausführen, die gemäß ursprünglicher Folge vor der aktuellen Sperrfreigabe liegen.
- Beispiel:
 - o ...; $L_1(C); L_2(C); L_1(A); L_2(D); U_2(B); U_1(A); U_2(D); U_1(C); \dots$
 - o T_2 wird bei $L_2(C)$ inaktiv gesetzt und kann damit auch seine Operationen $L_2(D)$, $U_2(B)$ und $U_2(D)$ nicht ausführen.
 - o Wenn T_1 allerdings die Sperre auf C freigibt ($U_1(C)$), dann kommt T_2 zum Zuge und kann auch gleich die drei Operationen durchführen – es sei denn, beim Setzen einer Sperre kommt es wieder zu einem Konflikt. Operationen von T_2 hinter $U_1(C)$ werden dann wieder in der vorgegebenen Reihenfolge durchgeführt.
- Beachten Sie: Wenn eine Transaktion inaktiv ist, kann sie natürlich auch keine Sperren freigeben.

Untersuchen Sie die drei Sperrfolgen und geben Sie an, ob sich die unerlaubten Fälle durch den Sperrmanager auflösen lassen. Wenn ja, geben Sie an, welche Reihenfolge der Aktionen dabei herauskommt. Falls der Sperrmanager nicht in der Lage ist, den Konflikt aufzulösen, skizzieren Sie, warum das nicht geht.

Aufgabe 5: Transaktionsfehler**(28 Punkte)**

Eine Datenbank soll dem Benutzer den Eindruck vermitteln, seine Transaktion laufe unabhängig von anderen Transaktionen, also quasi im Einbenutzerbetrieb. Dies wird durch den Begriff der Serialisierbarkeit ausgedrückt.

- a) Geben Sie die Definition für den Begriff der Serialisierbarkeit an (4 Punkte).

Mögliche Typen von Fehlern, die bei fehlender Synchronisierung der Transaktionen auftreten können, sind Lost Update, inkonsistente Sicht bzw. inkonsistente Datenbank sowie Phantome.

- b) Beschreiben Sie diese Fehlertypen und geben Sie jeweils ein Beispiel an (12 Punkte).
- c) Geben Sie für jeden dieser Fehlertypen einen Mechanismus an, mit dem der Fehler vermieden werden kann. Bleiben Sie dabei im Kontext der Sperrverfahren. Zeigen Sie anhand Ihres Beispiels, dass der Fehler dann nicht auftritt (12 Punkte).

Anmerkung zu c): Versuchen Sie, für jeden Fehlertyp das einfachste Verfahren zu beschreiben, das das Auftreten dieses Fehlertyps verhindert. Sie könnten zum Beispiel sagen: Jede Transaktion sperrt die gesamte Datenbank exklusiv. Dann würden diese Fehler sicherlich nicht auftreten, aber diese Aussage würde trotzdem als nicht richtig bewertet werden.