

## Datenbanken II B: DBMS-Implementierung — Klausur —

**Name:** \_\_\_\_\_

**Matrikelnummer:** \_\_\_\_\_

**Studiengang:** \_\_\_\_\_

Aufgabe	Punkte	Max. Punkte	Zeit
1 (Platten-Leistung)		4	10 min
2 (Puffer-Manager)		8	15 min
3 (Speicherberechnung)		6	15 min
4 (B <sup>+</sup> -Baum)		7	15 min
5 (Index-Auswahl)		6	15 min
6 (Verbunde)		6	10 min
7 (Oracle Data Dictionary Anfrage)		3	5 min
8 (Backups)		3	5 min
Summe		54	90 min

- Ich bin gesundheitlich in der Lage, diese Prüfung abzulegen.  
(Andernfalls bitte bei Aufsicht melden).
  
- Falls ich zu dieser Prüfung nicht angemeldet sein sollte, stelle ich beim Prüfungsausschuss unwiderruflich den Antrag, nachträglich angemeldet zu werden. Ich versichere, dass ich die Prüfung nicht bereits endgültig nicht bestanden habe.

\_\_\_\_\_  
(Unterschrift)

**Hinweise:**

- Bearbeitungsdauer: 105 Minuten
- Es dürfen drei Din A4-Blätter mit Notizen verwendet werden (Vorder- und Rückseite, also 6 Seiten, auch gedruckt). Weitergehende schriftliche Unterlagen (z.B. Bücher, Skripte), Notebooks, PDAs, etc. dürfen nicht verwendet werden. Mobiltelefone bitte ausschalten (oder mit Aufsicht besprechen).
- Die Klausur hat 15 Seiten. Bitte prüfen Sie die Vollständigkeit.
- Bitte benutzen Sie den vorgegebenen Platz. Wenn Sie auf die Rückseite oder ein Zusatzblatt ausweichen müssen, markieren Sie klar, dass es eine Fortsetzung gibt.
- Schmierpapier bitte deutlich durchstreichen und mit abgeben. Es sollen bitte keine Notizen über die Aufgaben den Raum verlassen.
- Tauschen Sie keinesfalls irgendwelche Dinge mit den Nachbarn aus. Notfalls rufen Sie eine Aufsichtsperson zur Kontrolle.
- Schreiben Sie bitte deutlich lesbar! Verwenden Sie keinen roten Stift. Bleistift ist nicht verboten, aber problematisch, wenn Sie sich nach der Klausureinsicht beschweren wollen.
- Mit 60% der Punkte haben Sie garantiert bestanden, die Grenze wird ggf. gesenkt.
- Sie dürfen vor 11<sup>30</sup> nicht gehen, danach nicht mehr auf Toilette.
- Fragen Sie, wenn Ihnen eine Aufgabe nicht klar ist!

**Aufgabe 1 (Platten-Leistung)****4 Punkte**

- a) Angenommen, Sie müssen eine Datei der Größe 8 GByte lesen, die auf einer üblichen Magnetplatte sequentiell gespeichert ist. Mit welcher Zeit würden Sie in etwa rechnen? Erläutern Sie kurz, wie Sie auf das Ergebnis gekommen sind. (Sie dürfen sinnvolle Werte wählen, mit denen leicht zu rechnen ist, und dann sagen, ob das eher eine sehr schnelle oder langsame Platte wäre.)

---

---

---

- b) Nun soll die gleiche Datei gelesen werden, aber sie ist in Blöcken von jeweils 8 KByte wild über die Platte verstreut. wie lange wird das in etwa dauern? (Auch hier geht es im wesentlichen um die Größenordnung, die Sie noch im Kopf berechnen können.)

---

---

---

- c) Was unterscheidet SSDs von Magnetplatten? Nennen Sie zwei Punkte.

---

---

---

**Aufgabe 2 (Puffer-Manager)****8 Punkte**

Sie sollen einen Puffer-Manager für ein neues DBMS implementieren, und haben sich für das übliche LRU-Verfahren entschieden.

- a) Was sind die Haupt-Datenstrukturen, die man für das LRU-Verfahren benötigt? Beschreiben Sie kurz, wie die Operation  $\text{pin}(f, n)$  abläuft, die Block  $n$  von Datei  $f$  zur Verfügung stellen soll. Machen Sie deutlich, wo welche Datenstrukturen benutzt werden. Komplikationen wie veränderte Blöcke brauchen Sie hier nicht zu betrachten.

---

---

---

---

---

---

- b) Wie kann man effizient mit veränderten Blöcken im Puffer umgehen? Sie könnten z.B. erläutern, wie Oracle das macht.

---

---

---

---

---

---

- c) Was ist das Hauptproblem eines reinen LRU-Verfahrens in Datenbanken (d.h. bei welcher Datenbank-Operation führt es zu ungünstigen Entscheidungen)? Beschreiben Sie mindestens zwei verschiedene Situationen, in denen höhere Schichten des DBMS über Wissen verfügen, das für den Puffer-Manager relevant ist und zu Modifikationen des strikten LRU-Verfahrens führen sollte.

---

---

---

---

---

---

- d) Nennen und erläutern Sie einen Parameter, den man bei Oracle im `CREATE TABLE` Statement angeben kann, und mit dem man die Pufferung beeinflussen kann.

---

---

---

---

---

---

**Aufgabe 3 (Speicherberechnung)****6 Punkte**

Gegeben sei eine Tabelle, in der Zugriffe auf einen Webserver protokolliert werden:

```
CREATE TABLE WEBLOG(  
  ID NUMERIC(8) NOT NULL CHECK(ID > 0) PRIMARY KEY,  
  TIMESTAMP DATE NOT NULL,  
  PAGE_ID NUMERIC(5) NOT NULL CHECK(PAGE_ID > 0)  
    REFERENCES PAGES,  
  IP_ADDR CHAR(15) NOT NULL,  
  REFERER VARCHAR(80) NOT NULL,  
  SESSION_ID NUMERIC(6) NULL CHECK(SESSION_ID > 0))
```

a) Wieviel Speicherplatz benötigt ein Tupel in Oracle unter folgenden Annahmen:

- Für die NUMERIC-Spalten gehen Sie vom schlechtesten Fall aus, der die volle Länge ausnutzt. Für NUMERIC(8) wäre das z.B. der Wert 10000001. Natürlich sind die Werte positiv.
- Ein DATE-Wert in Oracle benötigt 7 Byte (Jahrhundert, Jahr, Monat, Tag, Stunde, Minute, Sekunde).
- Die Spalte REFERER ist öfters leer. Rechnen Sie mit einer durchschnittlichen Länge von 10 Zeichen.
- Sie können davon ausgehen, dass alle String-Typen mit ein Byte pro Zeichen codiert werden.
- Rechnen Sie zuerst für den Fall, dass die SESSION\_ID bei Einfügung null ist (siehe b).
- Rechnen Sie den Row Directory Eintrag mit ein.

Geben Sie wesentliche Zwischenschritte der Rechnung mit an.

---

---

---

---

---

---

- b) Später läuft über die gesammelten Daten eine Analyse-Software, die versucht, einzelne Web-Zugriffe zu Sitzungen zusammenzusetzen. Sie führt ein `UPDATE` auf den Tupeln aus, das die `SESSION_ID` setzt (sie war bei der Einfügung `NULL`). Wie lang sind die Tupel jetzt (wieder inklusive Row Directory Eintrag)?

---

---

- c) Für welchen Parameter, der in Oracle beim `CREATE TABLE` angegeben werden kann, ist dieses verlängernde Update wichtig? Was wäre ein sinnvoller Wert (unter der Annahme, dass die Tupel sequentiell geschrieben werden, und die Analysesoftware mit dem Update deutlich später läuft)? Nennen Sie nicht nur den Namen des Parameters sondern auch seine Bedeutung/Wirkung.

---

---

---

---

- d) Was passiert, wenn man diesen Parameter zu klein wählt?

---

---

**Aufgabe 4 (B<sup>+</sup>-Baum)****7 Punkte**

Es sei jetzt der Index für die Tabelle `WEBLOG` aus der letzten Aufgabe betrachtet, der den Schlüssel `ID` implementiert. Folgende (halbwegs relative) Annahmen seien vorausgesetzt:

- Der Index ist als B<sup>+</sup>-Baum implementiert.
- Ein Eintrag in B<sup>+</sup>-Baum (aus Schlüsselwert und ROWID bzw. Pointer, sowie "Overhead") benötigt 20 Byte. Zur Vereinfachung sei angenommen, dass Einträge in den Blattknoten und Einträge in den Verzweigungsknoten gleich viel Platz benötigen.
- Ein Block von 4 KByte hat nach Abzug von "Block Header" und anderem "Overhead" noch 4000 Byte Platz.

Es sei weiter angenommen, dass die Tabelle zwei Millionen (2.000.000) Tupel hat.

- a) Wie viele Blattknoten kann es unter diesen Voraussetzungen geben (minimal und maximal)?

---

---

---

- b) Was sind mögliche Höhen des B<sup>+</sup>-Baums (minimal und maximal)? Zählen Sie dabei die Blattebene mit. Skizzieren Sie grob die Bäume.

---

---

- c) Wie viele Blockzugriffe braucht man im schlechtesten Fall, bis man das Tupel zu einer ID hat? Begründen Sie Ihre Antwort.

---

---

---

- d) Wie viel Speicherplatz wird der B<sup>+</sup>-Index belegen (wieder im ungünstigsten Fall)? Erläutern Sie Ihre Rechnung kurz.

---

---

---

- e) Wenn die IDs fortlaufend vergeben werden, wäre vielleicht eine andere (von Oracle unterstützte) Indexstruktur besser? Welche? Wie viele Blockzugriffe braucht man da im Normalfall?

---

---

---

**Aufgabe 5 (Index-Auswahl)****6 Punkte**

Gegeben sei die folgende Tabelle:

```
CREATE TABLE R(  
  A NUMERIC(8) NOT NULL PRIMARY KEY,  
  B NUMERIC(4) NOT NULL CHECK(B > 0),  
  C NUMERIC(2) NOT NULL CHECK(C BETWEEN 1 AND 10),  
  D VARCHAR(500) NOT NULL)
```

Die Tabelle ist recht groß (10 Millionen Tupel). Die möglichen Wertebereiche werden im wesentlichen auch ausgenutzt (z.B. sind viele Werte für D tatsächlich recht lange Zeichenketten, und die meisten der 10.000 möglichen Werte für B kommen in den Daten vor). Sie können auch Gleichverteilung annehmen und statistische Unabhängigkeit der Werte für verschiedene Attribute.

Die Tupel sind physisch in der Reihenfolge der A-Werte gespeichert (dies ist ein fortlaufend generierter Schlüsselwert), eventuell mit kleinen Schwankungen an den Blockgrenzen. Die Blöcke sind 4 KB groß und enthalten jeweils ungefähr 10 Tupel. Ein Full Table Scan muss also eine Million Blöcke sequentiell lesen. Eine ROWID benötigt 10 Byte. Sie können davon ausgehen, dass die B<sup>+</sup>-Baum Indexe recht frisch angelegt sind.

Zeichnen Sie für jede der folgenden Anfragen einen Oracle Zugriffsplan. Er kann einen oder mehrere der jeweils angegebenen Indexe verwenden. Ein Full Table Scan ist natürlich auch möglich. Sie dürfen aber nur die angegebenen Indexe verwenden, keine anderen. Beschreiben Sie kurz die Auswertung und warum Sie sich für diese Möglichkeit entschieden haben. Kreuzen Sie die verwendeten Indexe an (dabei sind auch kein Kreuz oder mehrere möglich).

a) 

```
SELECT COUNT(A)  
FROM R  
WHERE C = 7
```

- IND1 über (A)
- IND2 über (B)
- IND3 über (C)
- IND4 über (B, C)
- IND5 über (C, B)

---

---

---

---

---

b) `SELECT AVG(LENGTH(D))`  
`FROM R`  
`WHERE C = 7`

- IND1 über (A)
  - IND2 über (B)
  - IND3 über (C)
  - IND4 über (B, C)
  - IND5 über (C, B)
- 
- 
- 
- 

c) `SELECT A`  
`FROM R`  
`WHERE C = 7 AND B > 9950`

- IND1 über (A)
  - IND2 über (B)
  - IND3 über (C)
  - IND4 über (B, C)
  - IND5 über (C, B)
- 
- 
- 
-

**Aufgabe 6 (Verbunde)****6 Punkte**

Gegeben seien:

- $R(A, B)$ , eine Tabelle mit 100.000 Tupeln in 1000 Blöcken.  
Das Attribut B hat 50.000 verschiedene Werte (gleichverteilt, jeder kommt also ungefähr zwei Mal vor).
- $S(\underline{C}, A \rightarrow R, D)$ , eine größere Tabelle mit einer Million Tupeln in 10.000 Blöcken.  
In A kommen die 100.000 verschiedenen Werte wirklich vor (gleichverteilt), d.h. jeder Wert kommt ca. 10 Mal vor.

Es gibt folgende Indexe (jeder hat die Höhe 3):

- I1 über  $R(A)$
- I2 über  $S(C)$
- I3 über  $S(A)$
- I4 über  $S(A, D)$

Geben Sie für jede der folgenden Anfragen einen möglichst guten Auswertungsplan an (als Oracle QEP). Sie dürfen einen intelligenten Optimierer annehmen.

a) `SELECT S.C  
FROM R, S  
WHERE R.B = 12345 AND S.A = R.A`

---

b) SELECT S.D, R.B  
FROM R, S  
WHERE R.A BETWEEN 10000 AND 10010  
AND S.A = R.A

**Aufgabe 7 (Oracle Data Dictionary Anfrage)****3 Punkte**

Gegeben sei eine Tabelle MYTAB. Sie möchten wissen, in welchen Datenbank-Dateien jeweils wie viele Tupel dieser Tabelle gespeichert sind. Ausgegeben werden soll also jeweils der Dateiname und die Anzahl Tupel. Sie wissen, dass

```
DBMS_ROWID.ROWID_RELATIVE_FNO(R.ROWID)
```

die Nummer der Datendatei eines Tupels R liefert ("relativ" zum Tablespace, d.h. die Nummer ist eindeutig nur innerhalb eines Tablespaces, verschiedene Tablespaces können Dateien mit gleicher "Relative File Number" haben). Entsprechend gibt es auch Funktionen ROWID\_OBJECT ("data object id"), ROWID\_BLOCK\_NUMBER und ROWID\_ROW\_NUMBER. Folgende Data Dictionary Tabellen sind für die Anfrage möglicherweise relevant:

- TABS(TABLE\_NAME, TABLESPACE\_NAME, PCT\_FREE, PCT\_USED, INITIAL\_EXTENT, NEXT\_EXTENT, MIN\_EXTENTS, MAX\_EXTENTS, PCT\_INCREASE, FREELISTS, NUM\_ROWS, BLOCKS, EMPTY\_BLOCKS, CHAIN\_CNT, AVG\_ROW\_LEN, AVG\_SPACE, AVG\_SPACE\_FREELIST\_BLOCKS, NUM\_FREELIST\_BLOCKS, CACHE, LAST\_ANALYZED, ...)
- DBA\_SEGMENTS(OWNER, SEGMENT\_NAME, PARTITION\_NAME, SEGMENT\_TYPE, TABLESPACE\_NAME, HEADER\_FILE, HEADER\_BLOCK, BYTES, BLOCKS, EXTENTS, INITIAL\_EXTENT, NEXT\_EXTENT, MIN\_EXTENTS, MAX\_EXTENTS, PCT\_INCREASE, FREELISTS, FREELIST\_GROUPS, RELATIVE\_FNO, BUFFER\_POOL, ...)
- DBA\_DATA\_FILES(FILE\_NAME, FILE\_ID, TABLESPACE\_NAME, BYTES, BLOCKS, STATUS, RELATIVE\_FNO, AUTOEXTENSIBLE, MAXBYTES, MAXBLOCKS, INCREMENT\_BY, USER\_BYTES, USER\_BLOCKS, ONLINE\_STATUS)
- DBA\_OBJECTS(OWNER, OBJECT\_NAME, SUBOBJECT\_NAME, OBJECT\_ID, DATA\_OBJECT\_ID, OBJECT\_TYPE, CREATED, ...)

---

---

---

---

---

---

---

---

**Aufgabe 8 (Backups)****3 Punkte**

Angenommen, Sie sind DBA einer Oracle-Datenbank und u.a. für Backups zuständig. Nennen Sie drei Dinge, auf die Sie achten sollten (diese müssen nicht Oracle-spezifisch sein).

a) \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

b) \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

c) \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_