Fachbereich für Computerwissenschaften

Prof. Dr. Nikolaus Augsten
Jakob-Haringer-Str. 2
5020 Salzburg, Austria
Telefon: +43 662 8044 6347
E-Mail: nikolaus.augsten@sbg.ac.at

UNIVERSITÄT SALZBURG

Datenbanken II
Wintersemester 2018/2019

Prüfung
07.06.2019

Name: _____ **Matrikelnummer:** _____

Hinweise

- Bitte überprüfen Sie die Vollständigkeit des Prüfungsbogens (11 nummerierte Seiten).
- Schreiben Sie Ihren Namen und Ihre Matrikelnummer auf jedes Blatt des Prüfungsbogens und geben Sie alle Blätter ab.
- Grundsätzlich sollten Sie alle Antworten auf den Prüfungsbogen schreiben.
- Sollten Sie mehr Platz für eine Antwort benötigen, bitte einen klaren Verweis neben die Frage auf die Seitennummer des zusätzlichen Blattes setzen.
- Keinen Bleistift verwenden. Keinen roten Stift verwenden.
- Verwenden Sie die Notation und die Lösungsansätze, die während der Vorlesung besprochen wurden.
- Aufgaben mit mehr als einer Lösung werden nicht bewertet.
- Als Unterlage ist ein beliebig (auch beidseitig) beschriftetes A4-Blatt erlaubt.
- Zeit für die Prüfung: 90 Minuten

Unterschrift _____

Korrekturabschnitt — Bitte frei lassen

Aufgabe	1	2	3	4	5	6	7	8	9	10	Summe
Maximale Punkte	1	1	1	1	1	1	1	1	1	1	10
Erreichte Punkte											

Aufgabe 1 - *Indexstrukturen*. 1 Punkt

Zeichnen Sie für die folgende Tabelle einen **3-stufigen Sekundärindex** auf dem Attribut `Player`. Die **erste Indexstufe** soll **dense** und die **zweite und dritte Indexstufe** sollen **sparse** sein. Ein **Datenblock** kann **2 Tupel** speichern. In einen **Indexblock** können **3 Einträge** gespeichert werden.

`Player`	`Team`
Bell	Steelers
Brees	Saints
Brown	Steelers
Donald	Rams
Edelman	Patriots
Fournette	Jaguars
Gurley	Rams
Houston	Chiefs
Jones	Falcons
Kuechly	Panthers
Lattimore	Saints
Leonard	Colts
Mack	Bears
Michel	Colts
Sanu	Falcons
Thomas	Saints

Aufgabe 2 - *Slotted Page*. 1 Punkt

Gegeben sei eine Slotted Page mit folgenden Eigenschaften:

- Größe: $2^{10} = 1024$ Bytes,
- Adressierungstyp: **Word-Adressierung** (d.h. es kann nur jedes 2. Byte adressiert werden)

In dieser Slotted Page werden **3 Tupel** R, S, T gespeichert:

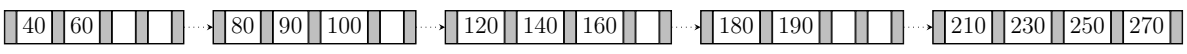
- d_1: $|R| = 111$ Bytes
- d_2: $|S| = 222$ Bytes
- d_3: $|T| = 333$ Bytes

Ergänzen Sie die Slotted Page um die **fehlenden Werte/Adressen**, wobei p_i und g_i sich auf den jeweiligen Datensatz d_i beziehen. **(0.125 Punkte pro Wert/Adresse)**

a	f	g_3	p_3	g_2	p_2	g_1	p_1	$\ldots$	d_1	d_2	d_3
									R	S	T

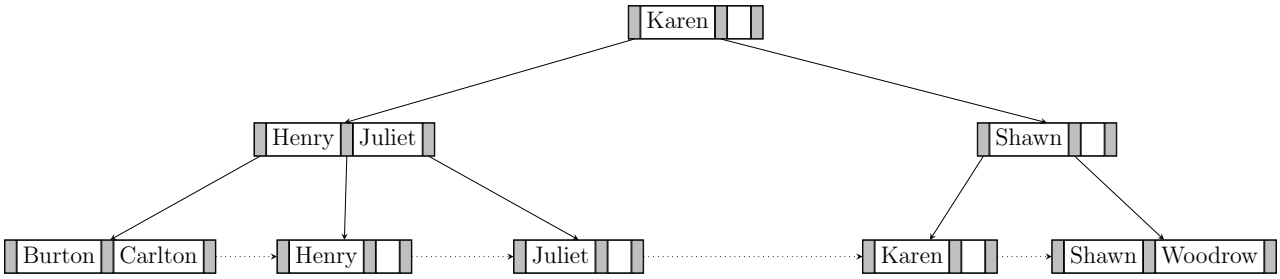
Aufgabe 3 - *B^+-Baum-Konstruktion.* 1 Punkt

Gegeben sind die Blätter eines B^+-Baumes ($m = 5$). Konstruieren Sie die **darüberliegenden Ebenen** (d.h. die inneren Knoten) so, dass ein **gültiger B^+-Baum mit minimaler Höhe** entsteht.



Aufgabe 4 - *B^+-Baum-Einfügen.* 1 Punkt

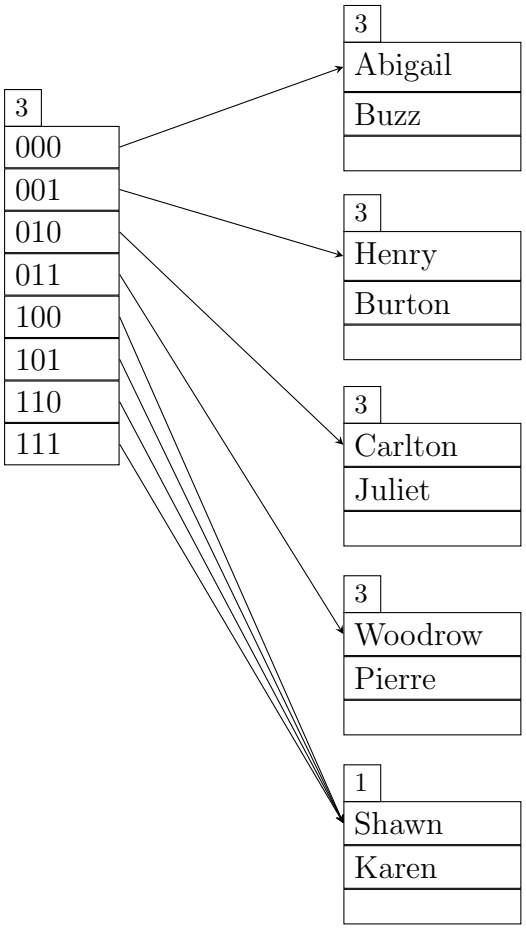
Gegeben ist ein B^+-Baum mit $m = 3$. Zeichnen Sie den B^+-Baum der nach dem Einfügen von **Dexter** entsteht.



Aufgabe 5 - *Erweiterbares Hashing*. 1 Punkt

Die Hashfunktion $h(x)$ liefert die in der Tabelle angegebenen Binärwerte. Es soll das Tupel *Buzz* aus dem gegebenen Hashcontainer gelöscht werden. Ein Bucket im Hashcontainer kann bis zu 3 Tupel speichern. Das Verzeichnis soll so klein wie möglich gehalten werden. **Illustrieren Sie den resultierenden Hashcontainer**.

x	$h(x)$
Abigail	0000
Burton	0010
Buzz	0001
Carlton	0100
Henry	0011
Juliet	0101
Karen	1110
Pierre	0111
Shawn	1100
Woodrow	0110



Aufgabe 6 - *Externes Merge-Sort*. 1 Punkt

Führen Sie **externes Merge-Sort** auf der folgenden Relation $R[A]$ aus.
Jeder **Block fasst 2 Tupel**. Die Größe des **Puffer** beträgt 6 **Blöcke**.

99
35
10
45
95
20
25
5
30
90
40
0
15
50
80
60
55
75
65
70
85

Aufgabe 7 - *Effiziente Anfragebearbeitung.* 1 Punkt

Gegeben sei eine Relation $R[A, B, C]$ mit folgenden Eigenschaften:

- $|R| = 8 \cdot 10^9$ Tupel,
- Pro Datenblock werden $16 \cdot 10^3$ Tupel gespeichert,
- Attribut A hat ganzzahlige Werte gleichverteilt im Bereich $[1; 8 \cdot 10^6]$,
- Attribut B hat ganzzahlige Werte gleichverteilt im Bereich $[1; 8 \cdot 10^9]$,
- Attribut C hat ganzzahlige Werte gleichverteilt im Bereich $[1; 8 \cdot 10^3]$,
- Duplikate werden mittels Tuple Identifier (TID) aufgelöst,
- Folgende Indizes existieren:
  - flacher sparse Index auf Attribut A mit 12.500 Einträgen pro Indexblock,
  - dense B^+-Baum-Index auf Attribut B, $m = 4.096$, minimale Höhe,

Es soll folgende Anfrage beantwortet werden:

$$\sigma_{B<7\cdot10^3 \wedge B>7\cdot10^3}(R)$$

Geben Sie die **Strategie (0.5 Punkte)** an und berechnen Sie die **Anzahl der Blockzugriffe (0.5 Punkte)** um die Anfrage **möglichst effizient** zu beantworten (1 Knotenzugriff im B^+-Baum entspricht 1 Blockzugriff).

Aufgabe 8 - *Join-Algorithmen.* 1 Punkt

Gegeben sind **zwei Relationen** mit folgenden Eigenschaften:

$R[A, B, C, D]$	$S[D, E, F, G]$
$\|R\| = 10.000$ Tupel, $b_R = 500$ Blöcke	$\|S\| = 30.000$ Tupel, $b_S = 1.000$ Blöcke
flacher dense Index auf A dense B^+ Baum Index auf D $(m = 20)$	sparse B^+ Baum Index auf D $(m = 20)$

Die Werte in R und S sind gleichverteilt und unabhängig.

Berechnen Sie die **Kosten (Blockzugriffe)** eines **Natural Joins** zwischen R und S mit Hilfe des **Index-Nested-Loop-Join** Algorithmus und geben Sie die **effizientere Join-Reihenfolge** ($R \bowtie S$ oder $S \bowtie R$) an.

Aufgabe 9 - *Bitmap Index Scan.* 1 Punkt

Gegeben sei eine Relation $R[A, B, C]$ mit folgenden Eigenschaften:

- $|R| = 10^9$ Tupel gespeichert auf $b_R = 10^6$ Blöcken,
- Clustered B^+-Baum-Index auf Attribut B, $m = 2.048$, minimale Höhe,
- Duplikate werden mittels Tuple Identifier (TID) aufgelöst,
- Attribut B hat insgesamt 10^4 mal den Wert 5.678, verteilt auf 3.000 Datenblöcke.

Es soll folgende Anfrage beantwortet werden:

$$\sigma_{B=5.678}(R)$$

Berechnen Sie die **Anzahl der Blockzugriffe**, um diese Anfrage **möglichst effizient**

a. **ohne Anwendung** eines Bitmap Index Scans **(0.5 Punkte)**,
b. **unter Anwendung** eines Bitmap Index Scans **(0.5 Punkte)**

zu beantworten.

Aufgabe 10 - *Anfrageoptimierung, Join-Reihenfolge.* 1 Punkt

Gegeben sind die folgende Relationen:

- $|R[A, B, C]| = 100000$, $V(R, A) = 150$, $V(R, B) = 50$, $V(R, C) = 50$
- $|S[C, D, E]| = 5000$, $V(S, C) = 20$, $V(S, D) = 50$, $V(S, E) = 100$

Die Werten in den Tupeln sind gleichverteilt und unabhängig.

1. Formulieren Sie die folgende Anfrage so um, dass Selektionen und Projektionen so weit wie möglich nach unten geschoben werden (sodass der kleinstmögliche Join durchgeführt wird). Geben Sie entweder den umformulierten relationalen Ausdruck oder den entsprechenden Operatorbaum an.

$$\pi_{B,C,D}(\sigma_{A=a \wedge D=d}(R \bowtie S))$$

2. Schätzen Sie die Kardinalität des Ergebnisses des Joins $R \bowtie S$ ab.