

## Aufgabe 1: I/O-System

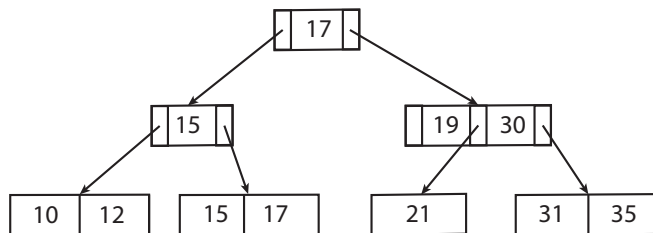
(1 P.)

- a) Gegeben ein Puffer, der maximal drei Seiten umfasst. Geben Sie für die folgende Sequenz bei einer Puffergröße von 3 an, welche Seiten sich nach jedem Zugriff im Puffer befinden und ob es ein Hit oder Miss war, einmal unter Benutzung von FIFO, einmal LRU und einmal LFU:  
**C B D D C E B C E A A E**
- b) Finden Sie ein Beispiel für eine Seitenzugriffssequenz, bei dem die Strategie FIFO echt weniger Cachemisses hat als LFU, und ein Beispiel bei dem FIFO echt weniger Hits hat als LFU. Nehmen Sie eine Puffergröße von 2 an.
- c) Berechnen Sie den Break-Even Point für eine Datei mit Größe 800MB. Die Festplatte habe Positionierungszeit 10ms, Leserate 150MB/s und Seitengröße 4KB. Wir rechnen mit 1KB = 1024 Byte und 1MB = 1024KB.

## Aufgabe 2: B<sup>+</sup>-Bäume

(1 P.)

- a) Betrachten Sie folgenden “B<sup>+</sup>-Baum” vom Typ (1, 2).



Welche Fehler sind in diesem Baum enthalten?

- b) Fügen Sie folgenden Schlüssel in der angegebenen Reihenfolge in einen leeren B<sup>+</sup>-Baum vom Typ (2, 1) ein. Zeichnen Sie die Bäume vor jedem Split.  
**M N Y P U J Z Q R O H**
- c) Wie viele weitere Schlüssel könnten in diesen Baum maximal eingefügt werden, ohne dass sich die Höhe ändert? Gehen Sie davon aus, dass zwischen der Wertebereich zwischen zwei Schlüsseln beliebig groß ist, etwa dass zwischen “A” und “B” noch “AA” ... “AZ” passen.
- d) Löschen Sie aus dem Resultat von Aufgabenteil a) die Schlüssel **Y**, **U** und **Q**. Zeichnen Sie die Bäume nach jedem Löschrisschritt.

## Aufgabe 3: Potpourri

(1 P.)

Um diese Aufgabe als gelöst zu markieren, müssen Sie 5 Teilaufgaben lösen.

- Geben Sie ein Beispiel an, bei dem eine rekursive SQL Anfrage mit `UNION` ein anderes Ergebnis liefert als mit `UNION ALL`.
- Wie muss ein Trigger aufgebaut sein, damit er das Datenbanksystem “lahmlegt”, wenn er ausgelöst wird? Illustrieren Sie an einem Beispiel.
- Gegeben zwei Relationen  $R$  und  $S$  wobei  $S$  zwei verschiedene Fremdschlüssel auf  $R$  hat, nämlich  $a$  und  $b$ , wobei  $b = a$  oder  $b = NULL$ . Welches Verhalten bei Änderung der referenzierten Daten muss für diese Schlüssel gelten, damit es egal ist, in welcher Reihenfolge sie überprüft werden?
- Begründen Sie: Hat eine Relation zwei Kandidatenschlüssel, so ist der eine **voll** funktional abhängig vom anderen.
- Gegeben die MVDs  $X \twoheadrightarrow Y$  und  $X \twoheadrightarrow Z$  in der folgenden Relation:

PersNr	Sprache	ProgSprache
1	griechisch	C
1	lateinisch	Pascal
1	griechisch	Pascal
1	lateinisch	C

Welche Tupel müssen eingefügt werden, wenn der Mitarbeiter 1 die Sprache “englisch” und die Programmiersprache “Rust” lernt?

- Geben Sie ein Beispiel an, was schiefgehen kann, wenn eine Zerlegung verlustlos aber nicht abhängigkeiterhaltend ist.
- Gelten die beiden Regeln sowohl unter Mengen- als auch unter Multimengensemantik?  
 $T \cap (R \cup S) = (T \cap R) \cup (T \cap S)$  und  $R \cup (S \cap T) = (R \cup S) \cap (R \cup T)$
- Begründen oder widerlegen Sie: Immer wenn ein Tupel in die Datenbank eingefügt wurde, enthalten alle definierte Sichten gleich viele oder mehr Einträge wie vor der Einfügung.
- Geben Sie eine äquivalenzerhaltende Transformationsregel für den Gruppierungsoperator  $\gamma$  (analog zu Vorlesung 9) an.
- Gegeben die Relationen  $L(A,B,C)$  und  $R(C,D,E)$ . Formulieren Sie eine Anfrage im Domänenkalkül, deren Antwort dem Semi-Join  $L \bowtie R$  entspricht und eine Anfrage im Tupelkalkül, deren Antwort dem Anti-Join  $L \bowtie R$  entspricht.
- Begründen Sie, wieso die Anfrage  $\{x \mid Rezept(-, x, -, -, -) \vee x = 5\}$  sicher ist, diese aber nicht:  $\{x \mid Rezept(-, x, -, -, -) \vee x \geq 5\}$
- Sind SQL-Anfragen, die nur aus `SELECT DISTINCT`, `FROM` und `NATURAL JOIN` bestehen, und Anfragen im konjunktiven Kalkül gleich mächtig?