COMP3278 Introduction to Database Management Systems

Assignment 3

Due Date: Nov 30, 2016 5:00pm

Question 1:

Suppose that each B+ tree node can hold up to 4 pointers and 3 keys. Assume that in any operation, if merging or redistribution of a node is necessary, its previous sibling, if exists, is used first.

a) Insert 2, 3, 5, 7, 9, 8, 17, 13, 11, 10 (in the same order as listed) into an empty B+ tree. Show the tree after each insertion operation.
b) Remove 9, 10, 8 (in the same order as listed) from the tree in part a. Show the tree after each removal operation.

Question 2:

Suppose that we are using extendable hashing on a file that contains records with the following search-key values: 0x80, 0x92, 0x13, 0x02, 0x81, 0x78, 0x44, 0xFF. Show the extendable hash structure after each insertion for this file. The hash function is $h(x) = x \ mod \ 4$ and each bucket can hold two records. Assume that a maximum of 2 bits of a hash value are used.

Question 3:

Suppose that a toy DBMS has a single hard disk and a four-slot buffer in the memory space, and the Least-Recently-Used (LRU) strategy is used. The following read/write requests are executed:

1. Read data block 1
2. Read data block 2
3. Read data block 3
4. Write data block 1
5. Read data block 4
6. Read data block 5
7. Write data block 4
8. Read data block 6
9. Read data block 7

Suppose the buffer is initially empty before the above requests are executed. For each of the requests above, write down the sequence of data transfer actions between the buffer space and the disk, and the resulting data blocks in the buffer. For example, for the first request, you may write:

1. Read data block 1
   Data transfer: Read block 1 from disk to buffer.
   Buffer: | 1 | | | |

Question 4:

Consider the following set *F* of functional dependencies on the relation *feedback(customerid, customername, feedbacknumber, comments)*:

customerid → *customername*

customerid, feedbacknumber → *comments*

a) Suggest a superkey for the relation *feedback* and prove (using Armstrong's axioms) it.
b) Show all the functional dependencies (with two attributes on the left hand side) including trivial functional dependencies of the above set *F*.
c) Find a canonical cover of the functional dependencies in relation *feedback*.
d) Is relation *feedback* in BCNF? If yes, give your explanation. If no, give a lossless join decomposition of *feedback* into relations in BCNF.

Question 5:

Consider the following database schema of a movie database.

Students (<u>StudentID</u>, FirstName, LastName, Email)

Courses (<u>CourseID</u>, CourseName, Description, Quota)

Enrollments (<u>EnrollmentID</u>, StudentID, CourseID, EnrollmentDate)

Write a relational algebra expression (or a set of relational algebra expressions) to answer each of the following queries.

a) Find the CourseID(s) that contains "Programming" in the description (i.e. Course.Description = "Programming").
b) Report the number of students enrolled in Each Course.
c) Find the Email(s) of students that contains "David" in his FirstName (i.e. Students.FirstName = "David") but not contains "Cheung" in his LastName (i.e. Students.LastName="Cheung").

Hand in:

Please hand in your written or printed assignment into assignment box A2 (Located at CYC Building 3rd floor). Electronic copies are not accepted. Please feel free to use the Moodle forum if you have any questions.