2019

COMPUTER SCIENCE

Paper : CSM-102

(Advances in Database Management System)

Full Marks : 70

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable.

Answer ***question no.* 1, 2** and ***any four*** from the rest.

1. Answer ***any five*** questions from the following : 2×5

(a) Consider the universal relation R = {A, B, C, D, E, F, G, H, I, J} and the set of functional dependencies F = {A, B} → {C}, {A} → {D, E}, {B} → {F}, {F} → {G, H}, {D} → {I, J}. What is the key for R?

(b) Consider the attribute set R = ABCDEFGH and the FD set F = {AB → C, AC → B, AD → E, B → D, BC → A,E → G}. Is the decomposition of R into (ABC, ACDE, ADG) with the same set of dependencies F, is (a) dependency-preserving? (b) lossless-join?

(c) State the optimizing criteria for Query optimization. Also mention the reason behind such selection.

(d) Comment on the cost of 'Block Nested loop Join algorithm' with justification.

(e) Why the B+ tree is considered as the widely accepted Index structure?

(f) What do you mean by Knowledge Discovery in Database?

(g) State the motivation behind finding out the minimal cover of a given set of FD.

(h) List two major difficulties with log based recovery without checkpoints.

2. Answer ***any five*** questions : 4×5

(a) Describe in brief the left deep plans and explain why the optimizers typically prefer such plans.

(b) Why is the implicit mode locking approach so important? Explain with an example.

(c) Suppose that a B+–tree index on (branch-name, branch-city) is available on relation 'branch'. What would be the best way to handle the following selection?

$\sigma_{(\text{branch-city}=\text{"Brooklyn"})\wedge(\text{assets}<5000)\wedge(\text{branch-name}=\text{"Downtown"})}$**(branch)**

(d) Consider the following schedule. *S* : T1:R(X), T1:R(Y), T1:W(X), T2:R(Y), T3:W(Y), T1:W(X), T2:R(Y) Draw the serializability graph for this schedule. State the importance of it.

(e) State and justify the Thomas Write Rule.

(f) What is meant by the term heuristic optimization? Discuss the main heuristics that are applied during query optimization.

(g) Consider a file of 1,000,000 records and are hashed into a table with 1000 buckets. Assume that 100 records will fit in a block. Also assume that all blocks are kept as full as possible but no two buckets share a block. What are the maximum and minimum number of blocks that needs to store this hash table?

(h) Show that with n relations, there are $(2(n-1)! / (n-1)!$ different join orders.

3. (a) Consider the relation *R*, which has attributes that hold schedules of courses and sections at a university;

R = {Course_no, Sec_no, Offering_dept, Credit_hours, Course_level, Instructor_ssn, Semester, Year, days_hours, Room_no, No_of_students}.

Suppose that the following functional dependencies hold on *R* :

{Course_no} → {Offering_dept, Credit_hours, Course_level}; {Course_no, Sec_no, Semester, Year} → {Days_hours, Room_no, No_of_students, Instructor_ssn}

{Room_no, Days_hours, Semester, Year} → {Instructor_ssn, Course_no, Sec_no}

Try to determine which sets of attributes form keys of *R*. How would you normalize this relation?

(b) How does the recovery manager ensure atomicity and durability of transactions? Describe through an example. (2+4)+4

4. (a) Consider the following schema :

Student (sid, name, age, address)

Book (bid, title, author)

Checkout (sid, bid, date)

Draw a relational algebra tree for the following query, assuming there are no indexes and data is not sorted on any attribute. Also compute the cost of this query and the cardinality of the result.

SELECT S.name **FROM** Student S, Book B, Checkout C **WHERE** S.sid = C.sid **AND** B.bid = C.bid **AND** B.author = 'Olden Fames' **AND** S.age > 12 **AND** S.age < 20.

Assuming :

- There are 10,000 Student records stored on 1,000 pages.
- There are 50,000 Book records stored on 5,000 pages.
- There are 300,000 Checkout records stored on 15,000 pages.
- There are 500 different authors.
- Student ages range from 7 to 24.

(b) Describe, with examples, the types of problem that can occur in a multi-user environment when concurrent access to the database is allowed. 6+4

5. (a) For each of the following locking protocols, assuming that every transaction follows that locking protocol, state and justify whether the 'conflict-serializability' property is ensured or not.

   (i) Always obtain an exclusive lock befor writing; hold exclusive locks until end-of-transaction. No shared locks are ever obtained.

   (ii) In addition to (i), obtain a shared lock before reading; shared locks can be released at any time.

   (iii) As in (ii), and in addition, locking is two-phase.

   (iv) As in (ii), and in addition, all locks held until end-of-transaction.

   (b) Write an algorithm that checks whether the concurrently executing transactions are in deadlock. 6+4

6. (a) Describe Validation based Concurrency control protocol. Is it better than Optimistic concurrency control? Jusify your comment.

   (b) Discuss the concept of Lock upgradation and down-gradation in context of achieving concurrency. (5+1)+4

7. (a) What is the difference between a primary index and a secondary index? Explain through an example.

   (b) Why can we have at most one primary or clustering index on a file, but several secondary indexes?

   (c) Suppose hashed indices is used in your database. Which types of queries you would not prefer?

   (d) "The chosen index strategy affects the cost estimations of a relational algebra operator"— comment with justification. 4+2+2+2

8. (a) What are the different type of measures used in OLAP?

   (b) In a data warehouse 5 dimensions namely A, B, C, D, E are present. Out of these dimension B has 2 abstract forms and D has 3 abstract forms. Remaining demensions are represented in 1 form only. What would be the number of cuboids is represented in the form of lattice?

   (c) Why is Cautious waiting better than no waiting? 4+4+2
