CS582 – Test 1

Solution

February 22, 2005

Note:

- There are 3 questions (100 points). This account for 25% of the total grade.
- Be precise in your answers. Provide justifications for your answers.

1. (Total: 30 points) Consider the database schema:

- STUDENT(SSN, Name, DOB, Status, Address, Major)
- TRANSCRIPT(SSN, CrsCode, Semester, Grade)
- COURSE(CrsCode, Title, DeptCode)
- DEPARTMENT(DeptCode, Name)

Note: The underlined attributes are the keys of the relations.

Consider the following query:

```
SELECT    S.SSN, S.Name, T.CrsCode, T. Semester, T.Grade, C.Title
FROM      Student S, Transcript T, Course C
WHERE     S.SSN = T.SSN AND T.CrsCode = C.CrsCode
          AND S.Status = ''Senior'' AND Major = ''CS''
```

(1.a) (5 points) Give an English description of the query.

(1.b) (5 points) We all know that that the above query is equivalent to a naive relational expression of the form:

$$\pi_L(\sigma_C(\text{STUDENT} \times \text{TRANSCRIPT} \times \text{COURSE}))$$

What are L (a list of attributes) and C (a condition) in the above expression?

(1.c) (10 points) What will be a fully pushed relational expression obtained from the expression in (1.b)?

(1.d) (5 points) A student suggests that the expression in (1.b) is equivalent to an expression of the form:

$$\pi_{L_1}(\sigma_{C_1}((\text{STUDENT} \rhd\lhd \text{TRANSCRIPT}) \rhd\lhd \text{COURSE}))$$

What are L_1 and C_1?

(1.e) (5 points) Suppose that you have to select either the expression in (1.c) or (1.d) to evaluate the query. Which one will you select? Explain why.

2. (Total: 40 points) Consider the database schema:

- STUDENT(SSN, Name, DOB, Status, Address, Major)

- TRANSCRIPT(SSN, CrsCode, Semester, Grade)
- GPA(SSN, GPA)

with the following properties:

- The STUDENT relation has 20,000 tuples with 10 tuples/page.
- The TRANSCRIPT relation has 100,000 tuples with 10 tuples/page.
- The GPA relation has 20,000 tuples with 40 tuples/page.
- There are 100 majors.
- On average, each senior completed around 40 courses.
- 10% of the student population are senior which are distributed evenly across all departments.
- There are 10 different values in the GPA column of the GPA; they are also distribute evenly across the student population and the department.
- For the relations STUDENT and GPA, there is a clustered, 2 level B^+ tree index of SSN.
- There is an unclustered hash index of GPA in the GPA's relation.
- There is also an unclustered hash index on SSN for TRANSCRIPT.
- The main memory has a 51 page buffer.

Consider the following SQL-query:

```
SELECT    DISTINCT S.SSN, S.Name
FROM      Student S, Transcript T, GPA G
WHERE     S.SSN = T.SSN AND S.SSN = G.SSN AND
          S.Status = ''Senior'' AND S.Major = ''CS'' AND G.GPA = 4.0
```

(2.a) (10 points) Estimate the size of the output. Provide the argument for your answer.

(2.b) (5 points) Draw a fully pushed query tree that can be used to evaluate the query.

(2.c) (5 points) In the query, you will have different ways to order the join operations between the relations STUDENT, TRANSCRIPT, GPA. Draw a fully pushed query tree in which the join between GPA and STUDENT is done before the join with TRANSCRIPT.

(2.d) (10 points) Analyze the trees in (2.b) and (2.c) to find out which one will yield better performance.

(2.e) (5 points) Based on your analysis (from 2.a to 2.d), suggest a better SQL query that yields the same result as the above SQL query.

3. (Total: 30 points)

(3.a) (10 points) Use ODMG ODL to define the classes STUDENT (with the attributes: id, name, phones, major, courses), COURSE (with the attributes: id, title), and DEPARTMENT(with the attributes: id, dname) and the following characteristics:

- *id* in STUDENT is a number in the range 100 to 999;
- *id* in COURSE is a number in the range 1 to 99;
- *id* in DEPARTMENT is a number in the range 0 to 9;

- *name*, *title* or *dname* is a string;
- *phones* is a collection of phone numbers each is a string;
- *courses* is a collection of courses that have been taken by the student;
- Each student majors in one department;
- In each class, *id* is a key (e.g., the id in the student class is a key of this class);
- Each student might have a close friend who is also a student;

Identify other (possible) integrity constraints and make sure that your definitions satisfy them.

(3.b) (10 points) Give some examples of objects belonging to the classes STUDENT, COURSE and DEPARTMENT (at least one for each class).

(3.c) (5 points) Write an OQL query that returns the courses of the student whose id is 111.

(3.e) (5 points) Write an OQL query that returns the name and id of all students who have taken the course 'CS582'.