中山大学软件学院软件工程专业 **2008** 级 (2010 秋季学期)

《SE-304 数据库系统原理》 期末试题参考答案 (A 卷)

(考试形式：开卷　　考试时间：2 小时)

Question 1. Database Design (18 points)

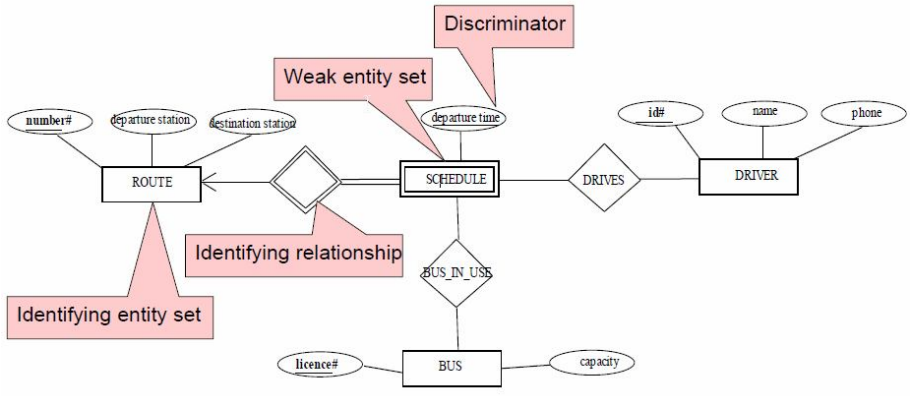
A bus company wants to keep track of its bus routes and schedules. Answer the questions according to the following description:

- Each bus route has a route number, a departure station and a destination station.
- For each bus route, there is a schedule, which records the departure times of buses.
- For each departure time of each route, a driver and a bus can be assigned (however this is not necessary - information about the driver or the bus may sometimes be missing)
- A driver has an employee Id, a name and a phone number.
- A bus is identified by its license number. The database also records the seating capacity of each bus.

(a) (10 points) Draw an ER diagram for the database, identify all constraints and keys, and **write down any assumptions** you have made.

(b) (8 points) Construct appropriate relation schemas for the ER diagram in (a), **underline** the primary keys.

Solution: (a) 10 points



(b) 8 points

Route(Number#, departure station, destination station)

Schedule(Route_Number#, departure time)

Driver(id#, name, phone)

Bus(licence#, capacity)

Drives(Number#, departure time, Driver_id#)

Bus_In_Use(Number#, departure time, licence#)

Question 2. RA and SQL (16 points)

Suppose a bookstore has the following five relational tables:

- BOOK (**BID**, TITLE, ***AID***, SUBJECT, QUANTITY-IN-STOCK)
- AUTHOR (**AID**, FIRST-NAME, LAST-NAME)
- CUSTOMER (**CID**, FIRST-NAME, LAST-NAME)
- ORDER-DETAILS (***OID, BID***, QUANTITY)
- ORDER (**OID**, ***CID***, ORDER-YEAR)

ASSUMPTIONS:

Keys are **underlined** and foreign keys are in ***italics***. Each author has authored at least one book in the store. Each book has exactly one author. Each order is made by exactly one customer and has one or more associated record in ORDER-DETAILS (e.g., an order may contain different books).

Write the following queries in **RA** (relational algebra) and **SQL**:

(a) Find all distinct book titles of the author whose last name is "Lee".

(b) Find the last name and first name of all authors who wrote books in at least two subjects.

Solution: ρ

(a) algebra:

$\pi_{\text{TITLE}}(\sigma_{\text{LAST-NAME = "Lee"}}(\text{BOOK} \rhd\lhd_{\text{AUTHOR.AID = BOOK.AID}} \text{AUTHOR}))$

SQL:

```
SELECT DISTINCT B.TITLE
FROM BOOK B, AUTHOR A
WHERE A.LAST-NAME = "Lee" AND A.AID = B.AID
```

(b)

algebra:

$\pi_{\text{LAST-NAME,FIRST-NAME}}(\sigma_{\text{B1.SUBJECT} \neq \text{B2.SUBJECT}}(\rho(\text{B1,BOOK}) \text{ JOIN}_{\text{AID}} \text{ AUTHOR JOIN}_{\text{AID}} \rho(\text{B2,BOOK})))$

SQL:

```
SELECT A.LASTNAME, A.FIRSTNAME
FROM AUTHOR A, BOOK B1, BOOK B2
WHERE B1.AID=A.AID AND B2.AID=A.AID AND
B1.SUBJECT ≠ B2.SUBJECT
```

Question 3. Constraints and Normalization (16 points)

In reality, FDs are given implicitly in the form of constraints when designing a database. Let a relation R=(Title, Length, Theater, City) where *title* is the name of a movie, *length* is the movie length, *theater* is the name of a theater playing the movie and *city* is the city where the theater is located.

Given the following constraints:

- The same name movies have the same length.
- Two different cities cannot have theaters with the same name.
- Two different theaters in the same city cannot play the same movie.
- A theater can play many movies.

(a) (6 points) Write the set of functional dependencies implied by the above assumptions.

Title→Length

Theater→City

{City, Title}→{Theater}

(b) (4 points) Identify the candidate key(s), and is R a 3NF or BCNF relation?

City, Title and Theater, Title. 3NF, not BCNF

(c) (6 points) If R is not BCNF, decompose it to BCNF.

R1{Title, Length}, R2{Theater, City}, R3{Theater, Title}

Question 4. Query Optimization (18 points)

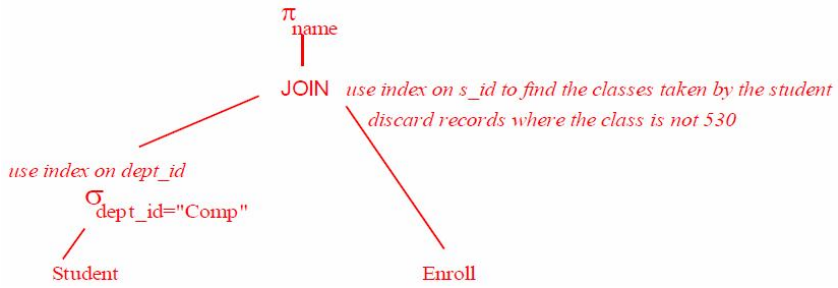
Consider two relational files **Student**(s_id, name, dept_id, address), **Enroll**(class_id, student_id, semester, grade). The **Student** file contains 10,000 records in 1,000 pages and the **Enroll** file contains 50,000 records in 5,000 pages. There are 10 different departments and 25 different classes. All attributes have the same length. Each index, wherever available, is a tree with 3 levels. For non-clustering indexes, each pointer is assumed to lead to a different page. Our goal is to process the query:

SELECT S.name
FROM Student S, Enroll E
WHERE S.dept_id="COMP" AND E.class_id="530" and S.S_id=E.Student_id

Some useful statistics:

A student enrolls on the average in 5 classes
A department contains on the average 1,000 students
Each class contains an average of 2000 enrolment records

(a) (5 points) Consider that the Student file contains a clustered index on dept_id, and the Enroll file contains a non-clustered index on student_id. Describe a fully pipelined plan (i.e., do not materialize anything) for processing the query by using Students as the outer relation and taking advantage of both indexes.



(b) (4 points) Estimate the approximate cost of your plan.

Selection needs to read 3+100 pages of Students (clustered index on dept_id) and will return 1,000 student ids and names.

For each (of the 1,000) student we use the index on s_id (for Enroll) to find the corresponding record: cost=1,000 (3+5)=8,000 (for each s_id we retrieve 5 records in classes).

Total cost= 8103 pages. (this cost does not include an extra level of indirection for the non-clustered s_id index - if we also include this the cost becomes 9103)

(c) (9 points) For the following questions, assume that there are no indexes. Using the above file sizes, estimate the cost of **block nested loops** with a main memory buffer of 102 pages (for each case explain briefly):

i) Student as the outer relation:

1,000+5,000*10 = 51,000

ii) Enroll as the outer relation:

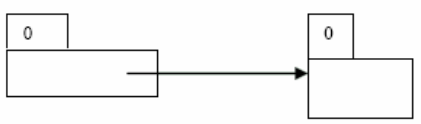
5,000+1,000*50 = 55,000

iii) What is the minimal cost that you can achieve and how many memory pages you need to achieve it?

The minimal cost is that of reading both files (6,000 pages). It can be achieved using a buffer of 1,002 pages (assuming Students as the outer relation).

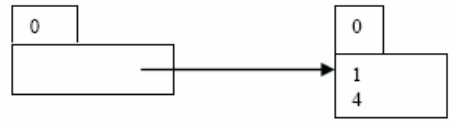
Question 5. Indexing and Storage structure (18 points)

Suppose that we are using extendable hashing index that contains the following search-key values K: 1, 4, 5, 7, 8, 2, 20. Assuming the search-key values arrive in the given order (i.e. 1 being the first coming key and 20 being the last one). Show the extendable hash structure for each insertion of the above key values file if the hash function is $h(x) = x \bmod 8$ and buckets can hold two keys. The initial configuration of the structure and the binary form (assuming the choice of bits starting from Least Significant Bit (LSB)) of all keys are given in the diagram below.

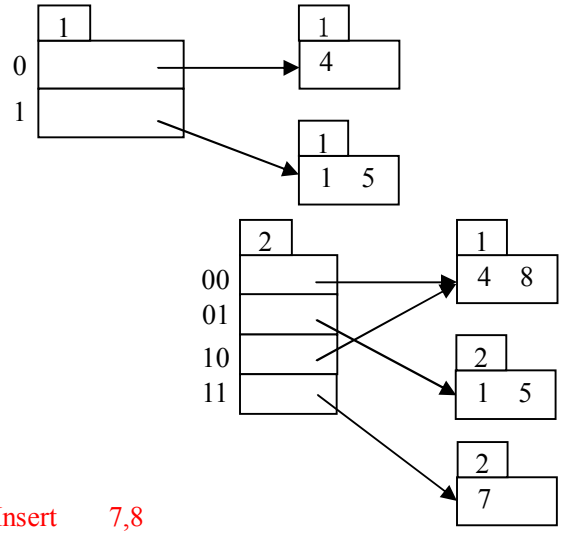


1	4	5	7	8	2	20
001	100	101	111	000	010	100

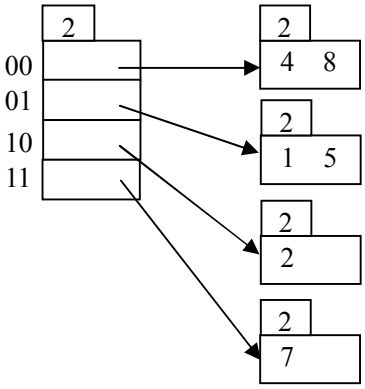
Initial hash table Empty bucket Binary values of all keys K

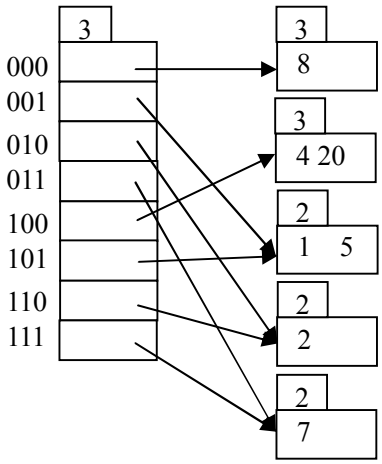


Insert 1, 4 Insert 5



Insert 7,8 Insert 2


2
00
01
10
11
2
4 8
2
1 5
2
2
2
7


3
000
001
010
011
100
101
110
111
3
8
3
4 20
2
1 5
2
2
2
7

Insert 20

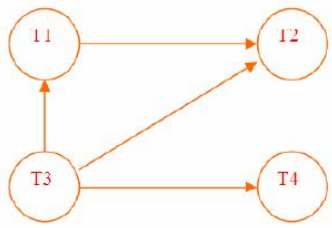
Question 6. Transaction Management (14 points)

Consider the schedule S that consists of four transactions as follows: S = < T1_R(X), T1_W(X), T2_R(X), T3_R(Y), T3_W(Y), T2_W(X), T3_R(Z), T4_R(Z), T4_W(Z), T1_W(Y), T2_R(Y)>. The notation is self-explanatory. For example, T2_R(X) means that transaction T2 reads item X.

T1	T2	T3	T4
R(X)			
W(X)			
	R(X)		
		R(Y)	
		W(Y)	
	W(X)		
		R(Z)	
			R(Z)
			W(Z)
W(Y)			
	R(Y)		

(a) (8 points) Construct the precedence graph of S. Explain why or why not the schedule is conflict-serializable.

Precedence Graph of S:



No cycle.

(b) (6 points) If you find that S is **serializable** in (a), write down all equivalent serial schedules of S.

T3, T4, T1, T2;

T3, T1, T4, T2;

T3, T1, T2, T4;