

Department of Computer Science,
New Mexico State University
Qualifying Exam for Databases
Spring, 2013

Open book.
Good luck!

2hrs
(100 points)

Answer Questions 1–4 by using the following relational database schema, where the underlined attribute(s) form(s) the primary key of the corresponding schema.

- Applicant(Loginid: varchar(16), Passwd: varchar(8), Name: varchar(64), Email: varchar(32), PhoneNo: char(10), SSN: char(9), Gender: char(1), Nationality: varchar(32))
- Education(Loginid: varchar(16), Degree: varchar(8), University: varchar(64), StartDate: date, EndDate: date)
Foreign key: *Loginid* references *Applicant(Loginid)*
- Deficiency(ApplicantLoginid: varchar(16), cid: char(5))
Foreign key: *ApplicantLoginid* references *Applicant(Loginid)*
Foreign key: *cid* references *Course(cid)*
- Course(cid: char(5), title: varchar(128))
cid is the id of a course, whose first two characters represent the department (e.g., 'CS', 'EE')
- Prereq (course_id: char(5), prereq_id: char(5))
Foreign key *course_id* references *Course (cid)*
Foreign key *prereq_id* references *Course (cid)*

1. (20%) Write a relational algebra expression (or a set of relational algebra expressions) using ONLY the five basic operators to answer each of the following queries.
 - A. (10%) For all the courses in 'CS' department, find their prerequisites. Note that the prerequisite courses may not come from 'CS' department. Show the course id, course title, prerequisite course id, and prerequisite course title.
 - B. (10%) Find the applicants who have the largest number of deficiency courses. Show their login ids and names.
2. (20%) Write an SQL statement to answer each of the following queries.
 - A. (10%) Find the applicants who have all the EE300 level courses as deficiencies. Show the ids and names of these applicants. (All EE300 level courses start with letters 'EE3')
 - B. (10%) Find all the applicants who already obtained a master's degree (i.e., in one education background item, *Degree='MSC'*). Show these applicants' login ids, names, and the universities that they graduated from. (Note: the universities should include all the universities from which the applicants have obtained some degree.)
3. (10%) Suppose that the following function dependencies hold on the *Education* relation.
 - *Loginid, Degree, University, StartDate* \rightarrow *EndDate*
 - *Loginid, Degree* \rightarrow *University*
 - A. (5%) What is/are the candidate key(s) for this relation? Please justify your answer.
 - B. (5%) Is the relation *Education* in BCNF? Please justify your answer.

4. (35%) Besides the schema information, you are also given the following information about the buffer, disk, and the relations:

- 4 buffer pages are available where each buffer page is 1K (i.e., 1024) bytes.
- Each disk page size is 1K (i.e., 1024) bytes.
- No record spans multiple disk pages.
- *Applicant* relation contains 10,000 tuples.
- *Education* relation contains 20,000 tuples.
- Each relation has primary B^+ index on primary key.
- There is a hash index on the *StartDate* attribute of the *Education* relation.

Note: If the provided information is not sufficient to answer the question and you need to make assumptions, please write down these assumptions clearly.

- A. (15%) Can the following query be efficiently answered by utilizing the hash index on *Education*? Please justify your answer.
 Can this query be effectively answered using the primary index on *Education*? Please justify your answer.
 $\sigma_{StartDate < '2009-01-01'}(Education)$
- B. (20%) Given a relational algebra expression “*Applicant* \bowtie *Education*”, please estimate (1) the number of results generated, and (2) the I/O cost if you apply block nested loop join algorithm.

5. (15%) Given the log file as shown in Figure 1,

LSN	LOG
00	begin_checkpoint
05	end_checkpoint
10	update: T1 writes P1
20	update: T2 writes P2
30	T2 abort
40	update: T3 writes P1
45	CLR: Undo T2 LSN 20
50	T1 commit
60	T1 End
70	* CRASH, RESTART

Figure 1: Log records

Answer the following questions to perform recovery when DBMS restarts:

- A. (7%) After the analysis phase, what are the content of dirty page table and transaction table?
- B. (8%) After finishing the UNDO phase, show the newly added log records and denote clearly the *undoneNextLSN* and *ToUndo*.

— End —