National Exams May 2015
98-Comp-B3, Data Bases & File Systems

3 hours duration

Notes:

1. If doubt exists as to the interpretation of any question, the candidate is urged to submit with the answer paper a clear statement of any assumption made.

2. This is a Closed Book exam. Candidates may use calculators.

3. Answer **five** questions as follow:
   a. **One** question from **questions 1 and 2** (only one question will be marked)
   b. **One** question from **questions 3 and 4** (only one question will be marked)
   c. **Three** questions from **questions 5, 6, 7, and 8** (only three questions will be marked)

4. All questions are of equal value. The marking scheme is as follows:

   Question 1: (a) 8 marks; (b) 4+4+4 = 12 marks

   Question 2: (a) 4 marks; (b) 4 marks; (c) 6+6 = 12 marks

   Question 3: 20 marks

   Question 4: (a) 10 marks; (b) 10 marks

   Question 5: (a) 4+6 = 10 marks; (c) 4+6 = 10 marks

   Question 6: (a) 4 marks; (b) 6 marks; (c) 7 marks; (d) 3 marks

   Question 7: (a) 3 marks; (b) 3 marks; (c) 10+4 = 14 marks

   Question 8: (a) 6 marks; (b) 6 marks; (c) 8 marks

5. All answers should be clear, legible and brief.

Question 1

Let a file have the following set of key values: (2, 5, 8, 10, 15, 18, 23, 25, 28, 31, and 33)

a) Construct a B+ tree index. Assume that the tree is initially empty, the values are added in ascending order, and the number of pointers that will fit in one node is 4.

b) Show what changes occur to the index when each of the following operations is done in turn: (i) insert 24; (ii) delete 2; (iii) insert 35.

Note that question 1(b) demands that you show a B+ tree (or partial B+ tree) for each insertion and deletion.

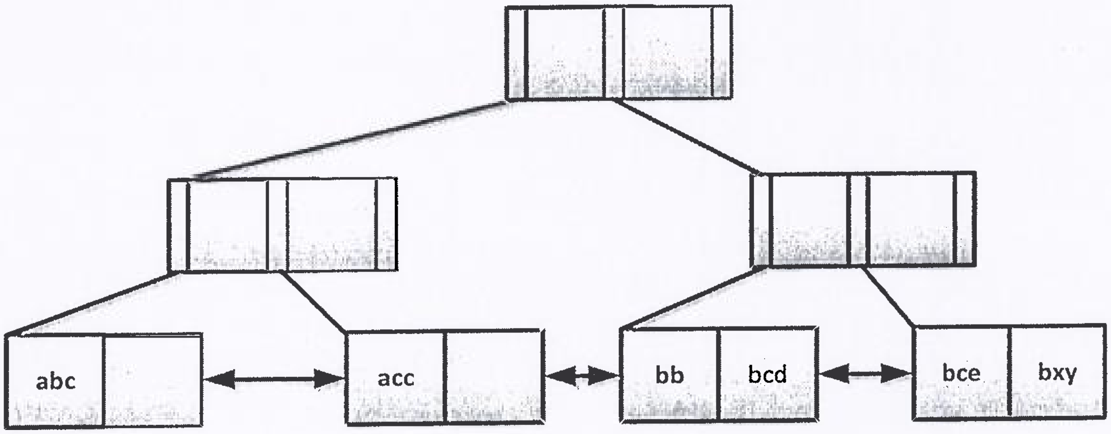
Question 2

a. Explain the difference between an equality search and a range search.

b. Does the final structure of a B+ tree depend on the order in which the items are added to it? Explain your answer.

c. Consider the partially specified B+ below:

i. Fill in the internal (empty) nodes without adding new keys.



ii. Add the key **bbb**. Show how the tree changes by drawing the new B+ tree.

Question 3

Downtown Records has decided to store information about musicians who perform on its albums (as well as other company data) in a database. The company has wisely chosen to hire you as a database designer (at your usual consulting fee of $1500/day). The following information describes the situation that the Downtown database must model.

- Each musician that records at Downtown has an SIN, a name, an address, and a phone number. Poorly paid musicians often share the same address, and no address has more than one phone.
- Each instrument used in songs recorded at Downtown has a unique identification number, a name (e.g., guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, E-flat).
- Each album recorded on the Downtown label has a unique identification number, a title, a copyright date, a format (e.g., CD or MC), and an album identifier.
- Each song recorded at Downtown has a title and an author.
- Each musician may play several instruments, and a given instrument may be played by several musicians.
- Each album has a number of songs on it, but no song may appear on more than one album.
- Each song is performed by one or more musicians, and a musician may perform a number of songs.
- Each album has exactly one musician who acts as its producer. A musician may produce several albums, of course.

Design a conceptual model for Downtown using an ER diagram. Be sure to indicate all key and cardinality constraints and any assumptions you make. Identify any constraints you are unable to capture in the ER diagram and briefly explain why you could not express them.

Question 4

Create an Entity-Relationship (ER) model for the the system below. Identify the entities and their attributes, as well as the relationships between entities. For every relationship, identify its cardinality, possible participation constraints (or structural constraints) and, if appropriate, role names, weak entity types and identifying relationships.

- Reservation system – A trip reservation consists of a sequence of flight reservations, where each flight reservation refers to a specific flight.
- Sometimes another flight is substituted for a booked flight because of equipment problems, weather delays, or customer preference.
- The passenger may reserve a seat for each flight.
- A trip reservation is made on some date; the passenger must purchase a ticket within a certain number of days or the reservation becomes void.
- The airlines use record locators to find a particular trip reservation quickly and unambiguously.
- A trip is reserved by an agent, who either works for an airline or a travel agency.

- The frequent flyer account may be noted for a passenger. The owner of the frequent flyer account must be the passenger.
- Multiple payments may be made for a trip, such as two credit-card charges.
- Payment may also be made by cash or check.

Question 5

Consider the following schema:

Employee (ID,Name,Address)
Supplier (ID,Name)
PurchaseOrder (OrderID,EmpIssuerID,SupplierID,Date)
PurchaseItem (ItemID,OrderID,ItemName,ItemCost)

The 1st two schemas are self-explanatory. Each tuple in PurchaseOrder describes a purchase order issued by a particular employee to a particular supplier. The last relation, PurchaseItem, describes each ordered item and its relationship to the corresponding order.

a. Write the following queries using the **SQL language**:

i. For each supplier, list the name and the total cost of all items ever ordered from this supplier.
ii. Names of employees who have issued a purchase order to *every* supplier.

b. Now, write the same queries (i.e. i and ii above) in **Relational Algebra**

Question 6

Consider the following relations containing student and course information:
Student (*Id, Name, Country*)
Course (*CrsCode, CrsName, Type, Instructor*)
Results(*Id, CrsCode, Grade)*

All the key fields are underlined. The *Type* field specifies the course type, e.g. MATH, STAT, SYSC, TTMG, ELEC, etc. The Results relation lists the grade that students (in Student relation) obtain for courses (in Course relation). Write the following queries a, b, and c in **SQL**.

a. Find the *Id* of students who take TTMG or SYSC course.
b. Find the *Id* of students who take every course.
c. Find the *Id* of students who take every SYSC course or take every TTMG course.
d. Consider the schema (i.e. student and course relations), state in English what the following relational algebra query computes.

$$\pi_{\text{Name}}(\pi_{\text{CrsCode}}(\sigma_{\text{Type = `SYSC'}}\textbf{Course}) \rhd\!\lhd (\sigma_{\text{Grade='D'}}\textbf{Result}) \rhd\!\lhd \textbf{Student})$$

Question 7

a. Define the term functional dependency.

b. Why is it that some functional dependencies are called trivial?

c. Consider a schema with the attribute set ABCDFG and the following FDs: AB → CD, BC → FG, A → G, G → B, C → G.

   i. Find a minimal cover of this set of FDs.

   ii. Is the decomposition of the previous schema into ABCD and CFG lossless?

Question 8

a. Given below is a schedule that is produced by a non-strict two-phase locking concurrency control: $\mathbf{w_1(x)\ r_2(x)\ w_2(x)\ commit_2\ commit_1}$

   (i) Is the schedule serializable and why?

   (ii) Is the schedule in commit order and why?

b. Give an example of a schedule at the READ COMMITTED isolation level in which a lost update occurs. Explain your example.

c. What happens to the schedule below at a REPEATABLE READ isolation level? Explain your answer.

   $\mathbf{r_1(x)\ r_1(y)\ w_1(x)\ r_2(y)\ r_2(x)\ w_1(y)\ commit_2\ commit_1}$