

Course name:

Student Number:

Student Name:

CS3462/CS3402 Midterm examination

Question 1

- (a) Describe the data semantics of Generalization and Categorization. (5 marks)
- (b) What are the major differences between Generalization and Categorization in terms of data volume in their related superclass and subclass entities? (5 marks)
- (c) Is there any special case such that categorization and generalization can be overlapped in terms of data volume? (5 marks)
- (d) What are the major differences between 2NF and 4NF with respect to n-ary relationships? (5 marks)
- (e) What is the ranking in sequence of efficiency of Join, Semi Join, Natural Join, Natural Semi Join and outer join operations and why? (5 marks)
- (f) Can associativity rule be applied in query optimization with or without conditions? Justify your answer with an example. (5 marks)
- (g) What are the major difference between qualified relation and idempotence of select / project operations in query optimization? (5 marks)
- (h) If there is a user requirement of multiple indexing, what kind of file is best appropriate for the application and why? (5 marks)

Question 2

Consider the universal relation $R = \{A, B, C, D, E, F, G, H, I, J\}$ and the set of functional dependencies $F = \{ \{A, B\} \rightarrow \{C\}, \{A\} \rightarrow \{D, E\}, \{B\} \rightarrow \{F\}, \{F\} \rightarrow \{G, H\}, \{D\} \rightarrow \{I, J\} \}$.

- (a) What is the 1NF for relation R? (6 marks)
- (b) Decompose R into 2NF relations R'. (7 marks)
- (c) Decompose R' into 3NF relations R". (7 marks)

Question 3

- (a) Given number of search-key values that fit in one index node is 3 and in one leaf node is 3. Construct a B+-tree for the following set of key values (31, 29, 23, 17, 11, 7, 5, 3, 2, 19). (10 marks)
- (b) what is its performance given both the index block size and the leaf block size are 3. (10 marks)

Question 4

Given a relational schema for retail store inventory system as follows:

Relation Stored_items

<u>*Store_id</u>	<u>*Item_no</u>	Quantity_held
005	004	200

Relation Store

<u>Store_id</u>	<u>*City_id</u>	Phone
005	001	27888888

Relation Ordered_item

<u>*Order_no</u>	<u>*Item_no</u>	Quantity_ordered	Ordered_price
003	004	50	80

Relation Items

<u>Item_no</u>	Description	Size	Weight	Unit_price
004	Lamp	Small	1 pound	100

Relation Orders

<u>Order_no</u>	Order_date	<u>*Customer_id</u>
003	13-11-2008	002

Relation Customer

<u>Customer_id</u>	Customer_name	<u>*City_id</u>	First_order_date
002	Mary Kim	001	13-11-2008

Relation Mail_order_customer

<u>*Customer_id</u>	Return_address
002	135 Market St, SF

Relation Walkin_customer

<u>*Customer_id</u>	Touring_guide
002	John Doe

Relation City

<u>City_id</u>	City_name	Headquarter_address	State
001	SF	123 Main St	CA

Notice that the bold and underlined are primary keys, and prefixed with * are foreign keys.

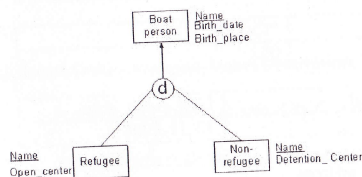
Using SQL,

- (a) show the correct sequence of using insert statements to load data into each one of the nine tables. (10 marks)
- (b) for each customer order, show the items ordered along with description, store id and City name and the stores that can fulfill the ordered items (5 marks)
- (c) find a list of Customer_id, Return Address and Touring_guide of dual Mail_order and Walk-in Customers. (5 marks)

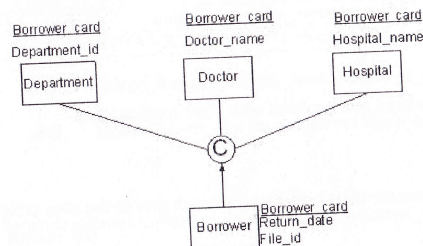
Answer:

Question 1

- (a) Generalization is a data semantic of multiple subclasses relates to one superclass in isa relationship such that they have the same key with the two subclasses' keys referring to superclasses' key as foreign key. Categorization is a data semantic of one subclass relates to multiple superclasses in isa relationship such that they have the same key with the subclass key referring to one of the superclass key.
- (b) Multiple subclass entities can be generalized into a superclass entity provided that the data occurrences of the subclass entities reside in its superclass entity. For example, a refugee and a non-refugee can both be a boat person, but a refugee cannot be a non-refugee, and vice versa as shown in the following EER model.



A subclass entity is a subset of a union (categorized) of its superclass entities such that the data occurrence of a subclass table must appear in one and only one superclass entity. For example, a patient record borrower can be a department, a doctor or a hospital in the following EER model:



- (c) In case of equal number of subclass entities and superclass entities occurrences, there is an overlap between Generalization and Categorization such that either case can be true.
- (d) 2NF is to normalize a relation into two binary relationship relations such that there will not be any partial functional dependency.
- 4NF is to normalize a ternary relationship into two binary relationships such that there will not be any multivalued dependency.

- (e) The major difference between Join and Natural Join operations are their join predicates in which natural join is an equijoin as a special case of Join.

The major differences between Join and Semi Join is that Semi Join adds an additional Projection operation on all attributes of the first operand relation after Join operation.

Similarly, the major difference between Natural Join and Natural Semi Join is that Natural Semi Join adds an additional projection operation on all attributes of the first operand relation after natural join operation.

In general, the resultant attributes of a Join operation is larger than a Semi Join operation because there is an additional projection operation after Join operation in Semi Join operation. As a result, the size of the resultant relation of Semi Join operation is reduced. Once the size of operand relation is reduced, the performance of Semi Join operation is increased. Similarly, the resultant attributes of a Natural Join operation is larger than a Natural Semi Join operation because there is an additional projection operation after Natural Join operation in Natural Semi Join operation. The size of the resultant relation of Natural Semi Join operation is thus reduced. Therefore, the performance of Natural Semi Join operation is increased.

On the other hand, an outer join includes non-matched tuples. As a result, it increases operand size and is in poor performance when compared with other join operations.

(f) Associativity rule of idempotence:

$$SL_F R \rightarrow SL_{F1} SL_{F2} R \quad \text{Condition } F=F1 \wedge F2$$

For example:

$$SL_{C-a} \rightarrow SL_{C-a-b} SL_{C-a-d} R$$

- (g) after qualification, a relation is portioned into divided parts, and then the optimizer needs to choose the relevant part for processing the query.

In select and project of a relation, the optimizer just processes the relevant part of a relation repeatedly or sooner in an operator tree.

- (h) When the requirement needs to process multiple indexes, that is, primary indexes and secondary indexes, inverted file is an appropriate choice because each inverted table can contain an indexing table.

Question 2 Answer:

(a) A minimal set of attributes whose closure includes all the attributes in R is a key. Since the closure of $\{A, B\}$, $\{A, B\}^+ = R$, one key of R is $\{A, B\}$. That is, FD: $A, B \rightarrow C, D, E, F, G, H, I$

Therefore 1NF R $\{\underline{A}, \underline{B}, C, D, E, F, G, H, I\}$ with composite key A, B

(b) To normalize into 2NF, we remove the attributes that are functionally dependent on part of the key (A or B) from R and place them in separate relations R1 and R2, along with the part of the key they depend on (A or B), which are copied into each of these relations but also remains in the original relation, which we call R3 below:

$R1 = \{\underline{A}, D, E, I, J\}$, $R2 = \{\underline{B}, F, G, H\}$, $R3 = \{\underline{A}, \underline{B}, C\}$

The set of relations in 2NF are $\{R1, R2, R3\}$

(c) The new keys for R1, R2, R3 are underlined. Next, we look for transitive dependencies in R1, R2, R3. The relation R1 has the transitive dependency $\{A\} \rightarrow \{D\} \rightarrow \{I, J\}$, so we remove the transitively dependent attributes $\{I, J\}$ from R1 into a relation R11 and copy the attribute D they are dependent on into R11. The remaining attributes are kept in a relation R12. Hence, R1 is decomposed into R11 and R12 as follows:

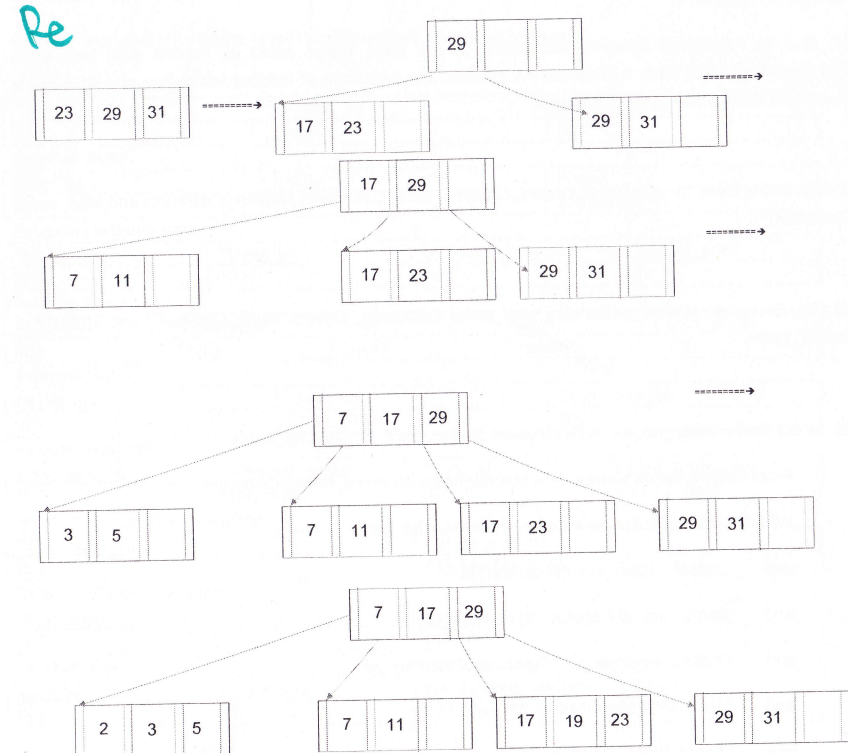
$R11 = \{\underline{D}, I, J\}$, $R12 = \{\underline{A}, D, E\}$

The relation R2 is similarly decomposed into R21 and R22 based on the transitive dependency $\{B\} \rightarrow \{F\} \rightarrow \{G, H\}$:

$R21 = \{\underline{E}, G, H\}$, $R22 = \{\underline{B}, F\}$

The final set of relations in 3NF are $\{R11, R12, R21, R22, R3\}$

Question 3 Answer: (a)



(b) The performance of I where i = number of I/O costs to access leaf block is

$$I \leq 1 + \log_d(n/e)$$

Given, if $n=10$, $e=(3+1)/2 = 2$ and $d=3$, the number read/writes of blocks in an operation is

$$I \leq 1 + \log_3(10/2)$$

$$I \leq 1 + \log_3(5)$$

$$I \leq 1.465$$

$$I = 1$$

As a result, we need $i+1 = 2$ I/O to get the data, one I/O from root to leaf block, and another I/O to data block.

Answer to Question 4:

(a) Due to referential integrity constraint, parent table tuples must be loaded data before its correspondent child table tuple data. As a result, the sequence of loading tables by using SQL Insert statement can be one of the followings:

(i) City, Store, Item, Stored_Items, Orders, Ordered_item, Customers, Walkin_customers and Mail_order Customer;

(ii) City, Customer, Walkin_customers, Mail_order Customers, Orders, Items, Ordered_Item, Stores and Stored_Items.

(b) Select Order.Order_no, Items.Description, Stores.Store_id, City.City_name

From Orders, Items, Stores, City, Stored_Items, Ordered_Items, Customer

Where Oorders.Order_no = Ordered_item.Order_no

and Ordered_item_id = Items.Item_id

and Items.Item_id = Stored_Items.Item_id

and Orders.Customer_id = Customer.Customer_id

and Stores.City_id = City.City_id

and Stored_items. Quantity_held > Ordered_item.Quantity_ordered

and Stores.Stored_id = Stored_Items.Store_id

(d) Select Customer.Customer_id, Customer.Customer_name, Walkin_customer.Touring_guide and

Mail_order.Customer_return_address

From Walkin_customer, Mail_order_customers, Customer

Where Walkin_customer.Customer_id = Customer.Customer_id

and Mail_order_customer.Customer_id = Customer.Customer_id