Brac University
Department of Computer Science and Engineering
Solution and Marking Rubrics

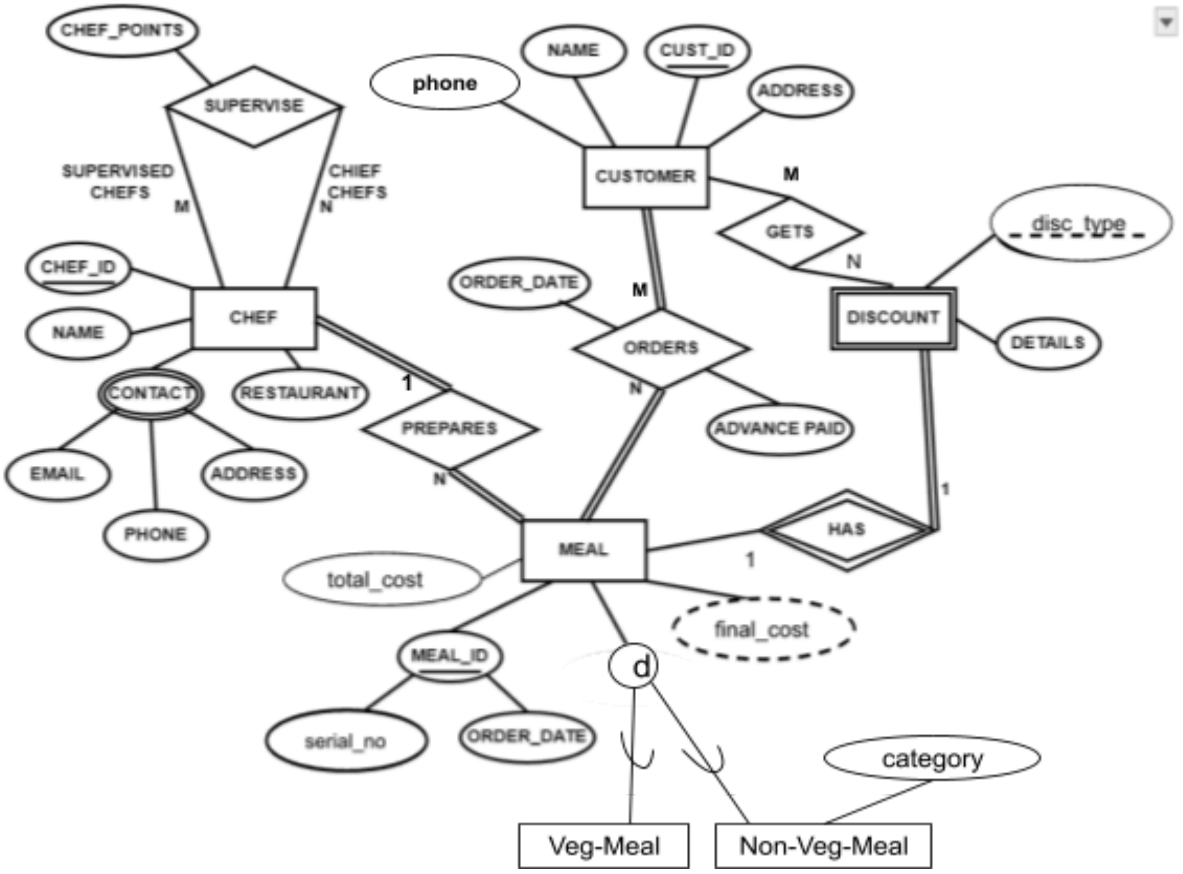
Examination: Final | Semester: Fall 2023
Duration: 1 Hour 40 Minutes | Full Marks: 45

CSE 370: Database Systems

Answer ALL of the following questions. Understanding the question is part of the exam.

Please Note, the question and rubrics are only an example. Marking may vary slightly from question to question and also every scenario for mark deduction may not be possible to mention in the rubrics. However, similar questions will be graded in a similar fashion. Also Note that this sample does not contain question from one chapter (Transaction) which has been added to syllabus after this particular semester.

1. [CO3] Construct a relational Schema by mapping the following EER diagram for an online Meal Delivery System. For the specialization/generalization portion, choose any applicable option except 8A: separate tables for subclasses and superclasses.	**13**

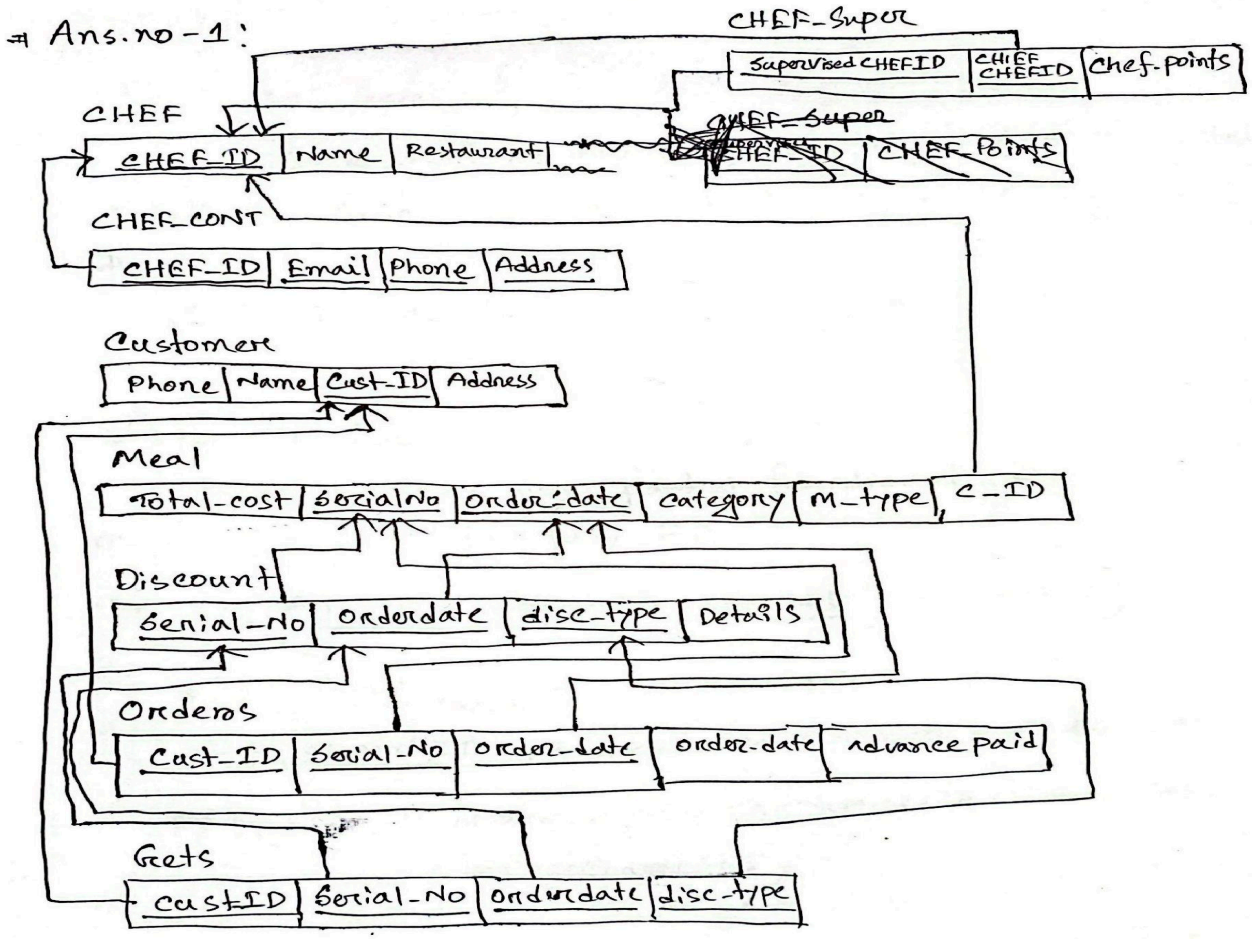


Marking Rubric
Total Marks : 13
Correct mapping of the the following:

- Mapping 3 regular entity, with correct attributes and primary keys -> 1 mark each = 3 marks (if pk incorrect or attributes missing deduct 0.5 from that entity's table)
- Composite key meal ID correctly shown and whole of it used as foreign key -> 1 mark
- Multivalued composite -> 1 mark
- 3 binary relationships and 1 recursive relationship with correct fk, pk and relationship attributes -> 1 mark each = 4 marks
- Subclasses correctly mapped using applicable option -> 1 mark (if 8A is used then 0 marks)
- Weak entity and correct identifying relationship -> 2 marks

- Derived attribute should not be shown in schema-> 1 marks

Solution



2. [CO4] Consider the following relation:
RentalService (clientID, propertyID, ownerID, cName, cContactNo, pAddress, size, rentStart, rentFinish, rentAmount, oName, oContactNo, oNomineeID, oNomineeName, oNomineeContactNo, contractID)

10

The primary key of the relation is underlined. The relation has the following additional functional dependencies (FDs):

FD1: clientID → cName, cContactNo
FD2: propertyID → pAddress, size, rentStart, rentFinish, rentAmount
FD3: ownerID → oName, oContactNo, oNomineeID, oNomineeName, oNomineeContactNo
FD4: size, rentStart, rentFinish → rentAmount
FD5: oNomineeID → oNomineeName, oNomineeContactNo

a. **Explain** if the above relation is in the first normal form (1NF) or not? If not, **apply** 1NF normalization. **2**
b. **Explain** if the relation(s) of no (a) is/are in the second normal form (2NF) or not? If not, **apply** 2NF normalization. **4**
c. **Explain** if the relation(s) of no (b) is/are in the third normal form (3NF) or not? If not, **apply** 3NF normalization. **4**

Marking Rubric
Total Marks : 10
a. 1 mark for stating it is in 1 NF and 1 mark for correct explanation (should mention all criteria for being

in 1 NF, if 1/2 is mentioned then deduct 0.5)= 2 marks

b. 1 mark for stating it is not in 2NF, 1 Mark for complete explanation(explanation should include which FDs are responsible for causing partial dependency), 2 Mark for correct Normalization and showing complete schema = 4 marks

c. 1 mark for stating it is not in 3NF, 1 Mark for complete explanation(explanation should include which FDs are responsible for causing transitive dependency and in which table), 2 Mark for correct Normalization and showing complete schema = 4 marks

Solution

a. In 1NF. Because there are no composite or multivalued attributes or nested relations.
b. Not in 2nf. Partial dependency on pk exists due to FD1, FD2, FD3.

Client(cliendID, cName, cContactNo)
Property(propertyID, pAddress, size, rentStart, rentFinish, rentAmount)
Owner(ownerID, oName, oContactNo, oNomineeID, oNomineeName, oNomineeContactNo)
RentalService (clientID, propertyID, ownerID, contractID)

c. Not in 3nf. Transitive dependency exists due to FD4, FD5 in tables property and owner respectively.

Client(cliendID, cName, cContactNo)
Property(propertyID, pAddress, size, rentStart, rentFinish)
RentInfo(size, rentStart, rentFinish, rentAmount)
Owner(ownerID, oName, oContactNo, oNomineeID)
Nominee(oNomineeID, oNomineeName, oNomineeContactNo)
RentalService (clientID, propertyID, ownerID, contractID)

3. [CO5] Consider the following relational database schema for a library management system.

Book

copyNo	ISBN	title	edition	year	price	available

Borrower

borrowerNo	borrowerName	borrowerAddress	country

BookLoan

copyNo	dateOut	dateDue	borrowerNo

[**Foreign Key**: copyNo references Book (copyNo) and borrowerNo references Borrower (borrowerNo)]

The primary keys are underlined and foreign keys are mentioned in bold under each table that has any foreign keys.

Write appropriate SQL statements for the following questions (for each question write a **single query**):

a. Retrieve all the Book title, ISBN, edition and year sorted by descending order of year. If the book is published in the same year, then sort based on the title alphabetically. **2**
b. Retrieve the number of books that have been published each year if more than 1 book was published in that year. Print the year and the number of books. **2**
c. List all the book titles starting with letter 'T', whose price is more than all the books published in 2014. **3**
d. Calculate the total price of the books borrowed by borrower 'John'. **3**

Note: There may be multiple possible queries for each question. This rubric is provided for one of the solution (given below).

Marking Rubric
Total Marks : 10

a. 1 marks for correct columns after select and 1 marks for order by correctly used = 2 marks

b. 1 mark for correctly using group by and 1 mark for correct usage of having and aggregate function count(*) = 2marks
c. 1 mark for correct usage of like and % and 2 marks for correctly using subquery and max function = 3 marks
d. 1 mark for using sum function and where clause correctly and 2 marks for joining the 3 tables correctly = 3 marks

Solution

Ans. no – 3 :

a) Select title, ISBN, edition, year
→ From Book
→ Order BY year DESC, title ASC; [Descending order]

b) Select year, COUNT(*) AS number-of books
FROM Book
Group BY Year
Having Count(*) > 1;

c) Select title
FROM Book
WHERE title LIKE "T%"
AND
price > (SELECT MAX(price) FROM Book
WHERE
year = 2014); [nested query]

d) SELECT SUM(B.price) FROM Book B
Join Booklean L on B.copyNo = L.copyNo
Join Borrower R on L.borrowerNo = R.borrowerNo
WHERE
R. borrowerName = "John";

4. [CO6]

a. **Construct** a B+ tree of order n = 4 for the following search key values inserted in the given order: 20, 15, 25, 10, 5, 12, 22, 30, 27, 17, 19, 28 **.** Each time there is a split, **a new B+ tree must** be drawn. **7**

b. **Construct** a hash index on attribute "Product_Code" of the "Product" table. The hash index has 5 buckets, each capable of holding a maximum of 2 index entries. Bucket overflow is resolved using forward chaining. **5**

The "Product" table is provided below:

Product_Code	Name	Price
EV5	Laptop	999.99

BU6	Smartphone	499.99
SP5	Smart TV	799.99
TR5	Wireless Headphones	129.99
AF8	Digital Camera	349.99
FK0	Coffee Maker	79.99
MP5	Bluetooth Speaker	59.99
UJ9	Fitness Tracker	149.99

ASCII Value Chart							
Character	**Value**	**Character**	**Value**	**Character**	**Value**	**Character**	**Value**
0	48	9	57	I	73	R	82
1	49	A	65	J	74	S	83
2	50	B	66	K	75	T	84
3	51	C	67	L	76	U	85
4	52	D	68	M	77	V	86
5	53	E	69	N	78	W	87
6	54	F	70	O	79	X	88
7	55	G	71	P	80	Y	89
8	56	H	72	Q	81	Z	90

Steps involving hash function is summarized below:

- Find the sum of the ASCII values for each character in the given Product Code string.
- Square the sum. Extract the two middle digits from the squared result. If the squared result has n digits then you take the (n // 2)-th and ((n // 2) +1)-th digits.
- Then, calculate the sum of the two extracted digits and take the remainder after dividing the sum with the number of buckets in the hash index.

Consider the example below:

- For Product_Code = **'GX2'**, the ASCII values of the corresponding characters, 'G' = 71, 'X' = 88 and '2' = 50. The sum of the individual ASCII values = 71 + 88 + 50 = 209.
- Square of the sum 209 = (209 * 209) = 43681. This result contains n = 5 digits. So the (5// 2)-th and ((5 // 2) +1)-th digits of the squared sum are 3 and 6 respectively. Sum of the middle digits = 3+ 6 = 9. Remainder = 9 % 5 = 4. So the index entry of **'GX2'** will be stored in bucket 4.

Marking Rubric

Total Marks : 12

a. There are 12 numbers to insert, correct insertion of each number is 0.5 marks -> so 6 marks and 1 mark for identifying maximum number of values in each node given n = **7 marks**

b. 1 mark for showing the right number of buckets and 0.5 marks for correct calculation and insertion of each number, there are 8 numbers -> so 4 marks. **Total 5 marks**

Solution

Ans.no-4:

(4)

a) order, n = 4

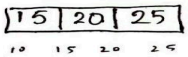
value = 1 to (n-1) = 1 to (4-1)
= 3

20, 15, 25, 10, 5, 12, 22, 30
27, 17, 19, 28

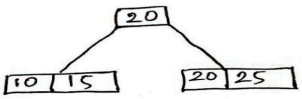
(i)

(ii)

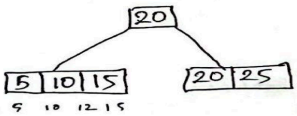
(iii)



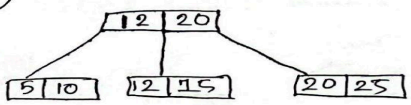
(iv)



(v)

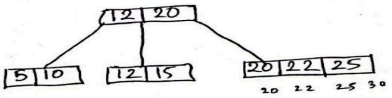


(vi)

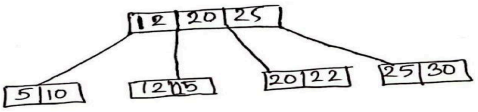


P.T.O.

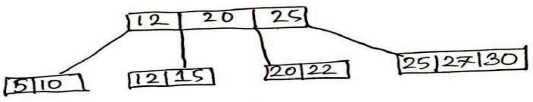
(vii)



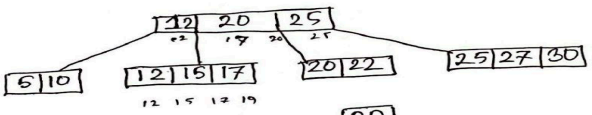
(viii)



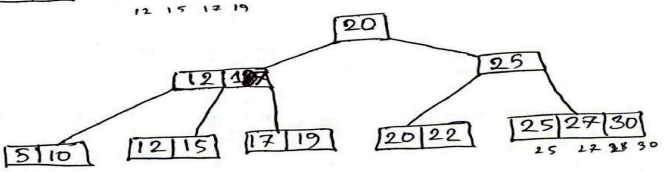
(ix)



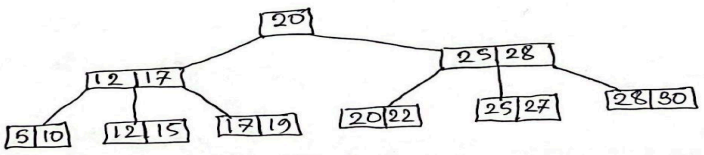
(x)



(xi)



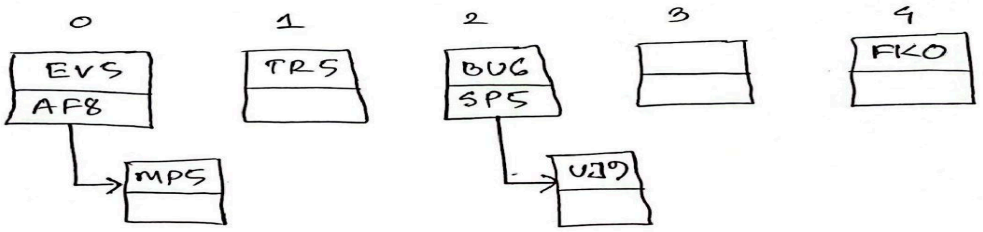
(xii)



b)

Buckets = 5

Entries = 2



h(EVS) = 69+86+53 = 208
= 208 × 208 = 43264
= (5//2) = 2
= (5//2+1) = 3
= 3+2 = 5 %.5 = 0

h(BUG) = ~~1+6 = 7 %.5~~ 2 +0
~~= 2~~ = 2 %.5 = 2

h(SPS) = 6+6 = 12 %.5
= 2

h(TRS) = 7+9 = 16 %.5
= 1

h(AF8) = 6+9 = 10 %.5 = 0

h(FKO) = 7+2 = 9 %.5 = 4

h(MPS) = 4+1 = 5 %.5 = 0

h(U39) = 6+6 = 12 %.5 = 2