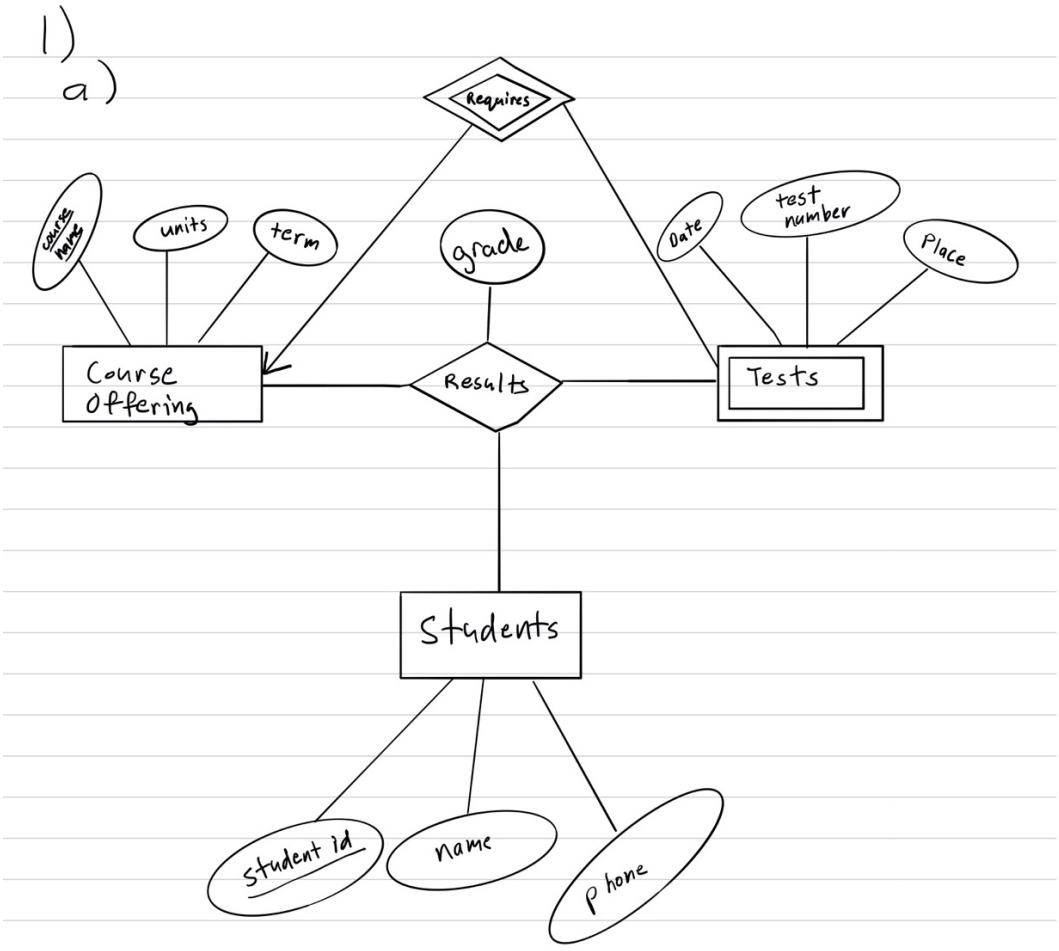
Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 1 (10 pts)

A **CourseOffering** (*coursename*, *units, term*) requires many **tests** (*test number*, *date, place*). A test number can be 1, 2, 3, etc. A test is associated with exactly one CourseOffering.

A **Student** (*studentId*, *name, phone*) takes a **test** and is given a *grade* for the test.

a) Show the ER diagram for the above scenario (**use text book notation**). If you do not use a drawing tool, hand-draw the diagram very neatly. If it is not clear, it will not be graded.



Mandeep Singh
COEN 178
Midterm 2
06/03/2020

b) Show the schema of resulting tables with primary and foreign keys.

b)

CourseOffering
Course Name
Units
Term

Primary Key

Tests
Test Number
Course Name
date
place

Primary Key (Course Name, Test Number)
Course Name: Foreign Key Reference Course Offerings

Students
Student ID
name
phone

Primary Key

Results
Course Name
Student ID
Test Number
Grade

Primary Key (Course Name, Student ID, Test Number)
Course Name: Foreign Key Reference Course Offering
Student ID: Foreign Key Reference Students

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 2 (10 pts)

An *International Science Consortium* is set up to hold details of scientists with expertise in various fields. Each **scientist** has an *id, name, affiliated university* and *email address*. A **field** (of expertise) has the *name* of the field. A scientist can have expertise in more than one field and each field has at least one scientist. Each field also has exactly one **lead scientist**.

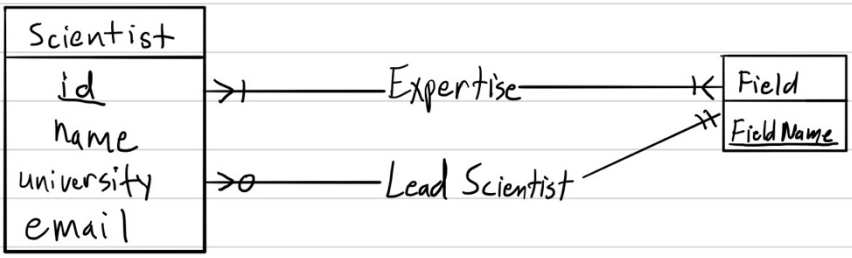
a) Show the ER diagram with Crows Foot notation.

b) Translate your ER diagram into a relational model and show the resulting tables. Identify the primary keys and foreign keys.

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

2)

a)



b) Scientist(id, name, university, email)
Primary Key (id)

Field (FieldName)
Primary Key (FieldName)

Expertise (id, Fieldname)
Foreign Keys (id, FieldName)

LeadScientist (id, FieldName)
Foreign Keys (id, FieldName)
Primary Key (FieldName)

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 3 (10 pts)

In *BookMarkers publishing company,* **a reviewer reviews several books and a book is reviewed by several reviewers.**

The following is a schema for table **Review** where each row contains information about a reviewer (**reviewerId, reviewerName and reviewerPhone**), information of the Book (**ISBN, title and author**) reviewed, the **date reviewed** and **reviewer Comments**.

Review (reviewerId, reviewerName, reviewerPhone, ISBN,title, author, dateReviewed, reviewerComments)

What are the problems that result from the above schema? Clearly specify the problems on **inserting**, **updating** and **deleting** records from this table. Your answer should refer to the table given and not a general discussion.

3) One of the problems with inserting is since there are no primary keys, the table is prone for repetitive entries. We don't want repetitive entries as that makes the database messy and more confusing. One of the problems with updating is since there are so many attributes, one has to be very specific to update an exact entry or else risk having unintended consequences of other entries being changed as well. The cost for updating and inserting are also more expensive and to write the code for them becomes much more complex. Also, the data may be inconsistent and there could be multiple rows for the same entry. If partial data is entered multiple times, the data could be in multiple different rows all over the place. The data redundancy also necessitates more storage. Also, data integrity is not maintained. The problem with deletion in this table is how specific you have to be when doing so. Since multiple reviewers can review multiple different books, many attributes have to be known in order to delete a specific entry. Otherwise, for example, if only the ISBN or reviewerid is provided, you risk having all data for the respective field pertaining to that value being deleted. With such a large relation, it is difficult to manage it and you would end up writing complex queries which ends up with slow performance. It is better to normalize the relation by having the minimum number of key attributes needed to identify a row and splitting information up into their respective tables. By adding more unnecessary attributes, it just makes the relation larger and more complex, reducing efficiency. All we really need is reviewerID, ISBN, date reviewed, and reviewer comments and have the reviewer information in a separate table and the book information in another. By creating a well maintained, structured database, we will improve performance across it.

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 4 (10 pts)

a) Show the schema of the tables that result from the diagram below. Show schema in the form, *table name*(*attribute1, attribute 2* …). Make sure you clearly identify the primary keys and foreign keys for the tables.

4)

a) Appointment(Day, Time, SSN, DoctorId)
Primary Key(SSN, DoctorId, Day, Time)
Foreign Keys(SSN, DoctorId)

Patient(SSN, Phone)
Primary Key(SSN)

Doctor(DoctorId, Location)
Primary Key(DoctorId)

Lab Tests(TestName)
Primary Key (TestName)

TestOrder(SSN, DoctorId, Day, Time, TestName)
Foreign Keys(SSN, DoctorId, Day, Time, TestName)

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

b) Now change the relationship, **Pat_App** as a non supporting (a single diamond) relationship. Show the schema of the tables that result after the change. Show all the tables with primary and foreign keys.

b) Appointment(Day, Time , DoctorId)
Primary Key (Day, Time, DoctorId)
Foreign Keys(DoctorId)

Patient(SSN, Phone)
Primary Key (SSN)

Doctor(DoctorId, Location)
Primary Key(DoctorId)

Lab Tests(Test Name)
Primary Key (Test Name)

Test Order (DoctorId, Day, Time , Test Name)
Foreign Keys (DoctorId, Day , Time, Test Name)

Patient App (SSN , DoctorId, Day , Time)
Foreign keys(SSN, DoctorId, Day, Time)

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 5 (5 pts)

Given the relation R (ABCDE) and

FDs:

B --→ DE
A -→ B
C --→ E,

Mark the statement below as True or False.

Applying BCNF decomposition algorithm to relation R, one possible collection of decomposed relations is **AB**,

CE and **ACD** (**True** or **False**)

5) TRUE

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 6 (5 pts)

Given the relation, R (ABCDE)

FDs:

AB --→ C

DE ----> C

B ---→ D

And decompositions, **R1 (DEC), R2 (ABDE),** show if the decomposition is loss-less or not, using the **Chase Test** with **tableau**.

6)

FDs: AB → C

DE → C

B → D

Initial

A	B	C	D	E
A_1	B_1	C	D	E
A	B	C_2	D	E

Chase 1
AB → C

A	B	C	D	E
A_1	B_1	C	D	E
A	B	C_2	D	E

Chase 2
DE → C

A	B	C	D	E
A_1	B_1	C	D	E
A	B	C	D	E

Chase 3
B → D

A	B	C	D	E
A_1	B_1	C	D	E
A	B	C	D	E

Since there is an unsubscripted row, the decomposition for R is lossless for this set of FDs.

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

Question 7 (10 pts)

Consider the schedule below and answer questions **a** and **b**.

T1	T2
Read (X)	
X = X - 10	
Write (X)	
	Read (X)
	X = X * 1.01
	Write (X)
Read (Y)	
Y = Y + 10	
Write (Y)	
	Read (Y)
	Y = Y * 1.01
	Write (Y)

a) Is the schedule above conflict-serializable? Give your reasons.

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

7) Let S be the original schedule

a) Yes, we can convert this schedule S into a schedule S' by a series of swaps of non-conflicting instructions S.t. S' is:

T1	T2
Read (X)	
X = X - 10	
Write (X)	
Read (Y)	
Y = Y + 10	
Write Y	
	Read (X)
	X = X * 1.01
	Write (X)
	Read (Y)
	Y = Y * 1.01
	Write (Y)

Therefore, the schedule S is conflict serializable

Mandeep Singh
COEN 178
Midterm 2
06/03/2020

b) Does 2-phase locking allow this schedule? Give reasons.

b) Yes, this schedule allows 2PL. It could look like this:

Time	T1	T2
t_1	Lock(X)	
t_2	Read(X)	
t_3	X=X-10	Lock(X)
t_4	Write(X)	Wait
t_5	Unlock(X)	Wait
t_6	Lock(Y)	Lock(X)
t_7	Read(Y)	Read(X)
t_8	Y=Y+10	X=X*1.01
t_9	Write Y	Write(X)
t_{10}	Unlock(Y)	Unlock(X)
t_{11}		Lock(Y)
t_{12}		Read(Y)
t_{13}		Y=Y*1.01
t_{14}		Write(Y)
t_{15}		Unlock(Y)

Although, theoretically, there could be a deadlock, 2PL works and allows this schedule.