Answers:

1997 Database Comprehensive Exam
30 minutes, January 19 1998

1. Design (7 minutes)
List the normalized (3NF) relations needed to represent the data corresponding to this E-R model.

Underline the key-fields.

Patients (idno, name, address)

Personnel (pidno, pname, paddress, type, works-at[Clinic])

/ works-at: Clinic has multiple personnel */*

Clinic (title, location)

/ has: Patient can have multiple diagnoses */*

/ is-of: Disease-class can have multiple Diagnoses */*

Diagnosis (idno[Patient], ICD, ~~idno~~, date, severity, is-of[Diseaseclass])

Diseaseclass (code, system)l

sees (idno[Patient], pidno[Personnel]) / ER m-n relationship set requires distinct relation */*

certified (pidno[Personnel], code[Diseaseclass]) / ER m-n relationship set */*

serves (title[Clinic], code[Diseaseclass]) / ER m-n relationship set */*

Information in [ ] is redundant, since all attributes have unique names.

2. Design revision (5 minutes)
The committee decides on a refinement: for three values of type: { M.D., nurse, clerk },

distinct data are needed:

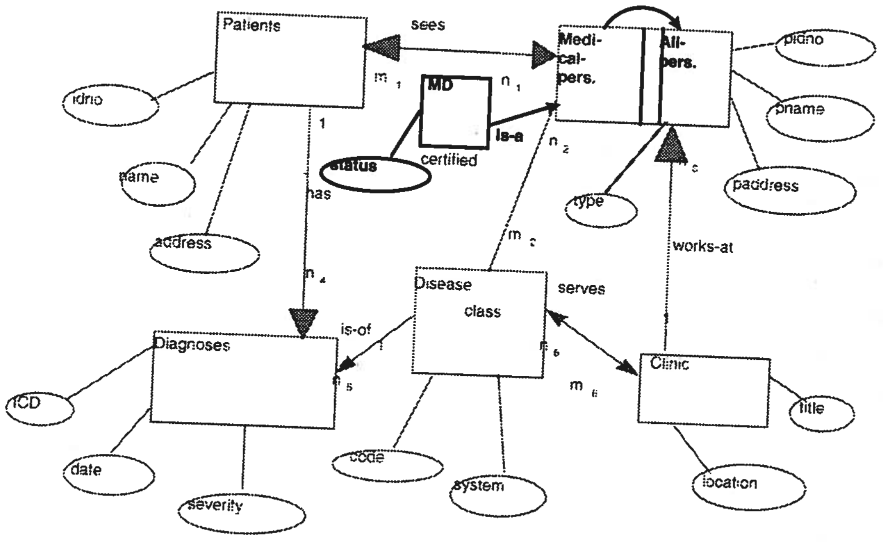
a. for the M.D.: Status {intern, resident, staff, community, consultant}
add MD ((pidno[(Medical-)Personnel], status)

b. for the clerk, no certification or patient seen data is needed.
split Personnel into two:
Medical-personnel(pidno[All-Personnel])
All-personnel(pidno, pname, paddress, type, works-at[Clinic]))
and change the association relations to
sees (idno[Patient], pidno[Medical-personnel])
certified (pidno[Medical-personnel], code[Diseaseclass])

An alternative is to create 3 subrelations of Personnel, one each for MD, nurse, and clerk, but then there will also be two `sees' and two `certified' relations, which have to managed distinctly.

Letting the certified and sees field for a clerk be NULL is a lazy (and common) solution, but will give problems when joining along these links, as in question 4.

Sketch the diagram revision (only the changes) and list the new relations needed.



3. SQL (5 minutes)
In the **paddress** text field appears a town name that can match the **location** of a **Clinic.**

Write the SQL query that lists **Personnel pnames** living in the same town as a **Clinic title.**

SELECT pname FROM Clinic, Personnel WHERE paddress CONTAINS location.
If the format of paddress is structured with town name in front then
SELECT pname FROM Clinic, Personnel WHERE location LIKE paddress
should work,

Note: the query was for all personnel, not just for WHERE works-at= title

4. Relational Algebra (8 minutes)
Write relational algebra statement that create a relation listing

Patients name, Diagnoses, and **Personnel pname ,** where the personnel was NOT certified for the **Disease-class** for the **Diagnoses.**

Create two relations with matching attributes, one with all Personnel seen and the Disease class for the Diagnoses, where the Disease code = is-of, and one with the Personnel seen that is certified for the Disease code, then take the difference, and project the code out of the final result. The Disease-class.code must be included in both intermediate relations to avoid subtracting out an uncertified personnel member for some code, who also seen the patient for a good certified code. (That would in fact the most common problem in practice)

Π name, ICD, pname (

Π name, ICD, pname, code (* ⋈

Personnel {pidno |X| pidno}sees{idno |X| pidno}Patients{has |X| ICD}Diagnoses) -

Π name, ICD, pname, code ←is-of (*

Patient {idno |X| idno} sees {pidno |X| pidno}Personnel {pidno |X| pidno} certified))

*The indicated {join attributes} are optional. The * projections can also be omitted at a cost.*

5. Performance (5 minutes)
A database optimization heuristic is to perform selections and projections prior to join operations.

What is the reason for this heuristic?
The heuristic is to reduce the sizes of the inputs to a join, since a join is O(n x m). Selection reduces the number of rows and projection the number of columns.

When is it invalid?

Some joins, as joins among sorted relations O(m+n) are linear, and even other joins, as hash-joins with small sizes are quite fast O(n(1+d)), while projection is typical O(n log n) and selection without an index is O(n).
Don't project out attributes used in subsequent join or selects, consider if doing two projections is worth it.
<< Don't project out potentially useful indexes. >>

6. Culture (1 minute)
Who defined the relational model and when?
E.F. *(Ted) Codd, of IBM San Jose Research, the prime paper appeared in Comm.ACM, 1970. There were some earlier IBM reports and later elaborations in ACM SIGFDT (predecessor of SIGMOD) Proceedings by Codd.*

What were the prior alternatives in designing databases?
Hierarchical, following systems specifications as IBM's IMS or others.
Network, following the CODASYL model.
The E-R model came later (Chen in ACM TODS 1975).