1 Short Answer

a) What is a foreign key and what constraints does this key enforce (in SQL)?

A foreign key is an attribute in a table that indicates a record is related to another record in another table.

The foreign key constraint requires that this attribute, for a particular record, have a matching record attribute in the other table or be null.

b) In SQL++, list two commands that are part of the Data Definition Language

Create Type

Create Dataset

c) Give two transactions and a schedule that illustrates the phantom problem.

T_1

Select * from T

Select * from T

T_2

Insert into T

Schedule

Select * from T

Insert into T

Select * from T

d) Suppose there is a table R(a,b) where B(R) = 1000, T(R) = 50000, V(R,a) = 300 and V(R,b) = 200. Values of R.b range from -20 to 80. What ist the minumum number of disc accesses it would take to retrieve all records in R where 30 ≤ R.b ≤ 60. Why? Assume R.b has an unclustered index.

1000

Since $\frac{30}{100} \cdot 50000 = 15000 > 1000$

It is faster to perform the sequential read than to use the index

e) Does the block nested loop join approach require indexes? Why or why not?

No. All of the blocks of both tables must be read

f) For which of the following does a semi-structured approach have the largest benefit over RDBMS? Many-to-one, many-to-many, one-to-many or one-to-one? Why?

One-to-many because semi-structured data allows nesting collections while RDBMS tables are NF

g) Suppose we are designing two databases to handle transactions under analytical and transactional data usage. Under which do we expect deadlock to be more difficult to prevent? Why?

Transactional. If data is purely analytical (read-only) there would be no need to lock tables. Multiple readers can use the source at once

h) Is it possible for a schedule to be serial, but not conflict-serializable? Explain.

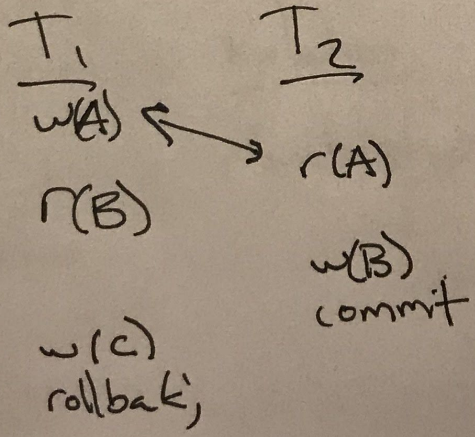
No. If a schedule is serial, then no swaps are necessary to produce a serial schedule

i) Will a Map/Reduce job take more time if a straggler is a Map task or a Reduce task? Explain.

Because reduce tasks must wait for all map tasks to complete, we expect a straggler map to have a larger impact on runtime

j) What does it mean if a schedule is unrecoverable? Give an example.

A schedule is unrecoverable if a write based on a ~~rolled back~~ read which was rolled back is then committed



k) Suppose we have the following schema:

```
Person(pid,name,phonenum,city)
Organization(oid,name,address)
Membership(pid,oid,donationamt,startyear,birthday)
```

Which attributes in which tables would we expect to be functionally dependent? List all that apply.

Membership. pid and birthday.

Presumably this "birthday attribute is relevant to the Person.

l) From the schema above, which of these tables are entities and which are relationships? If a table is a relationship, identify if it is many-to-many, one-to-one or many-to-one.

Entity

Person

Organization

Relation

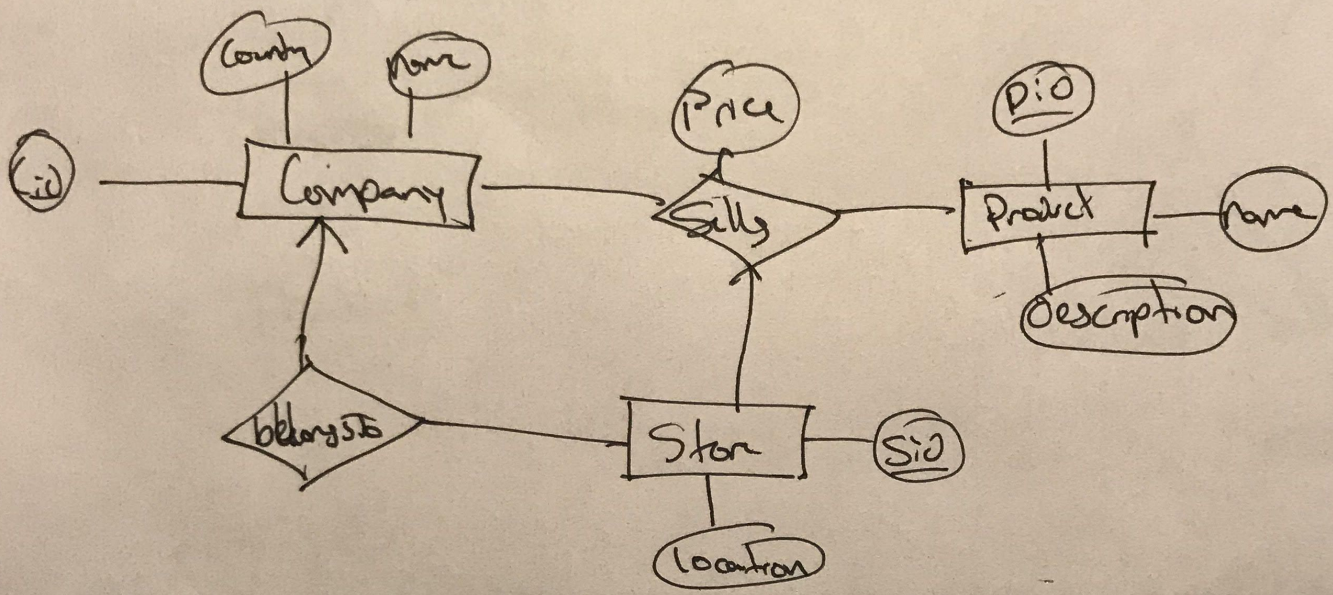
Memberships (many-to many)

2 SQL

For this question, use the following schema:

Company(cid,country,name)
Product(pid,name,description)
Store(sid,cid,location)
Sells(sid,pid,price)

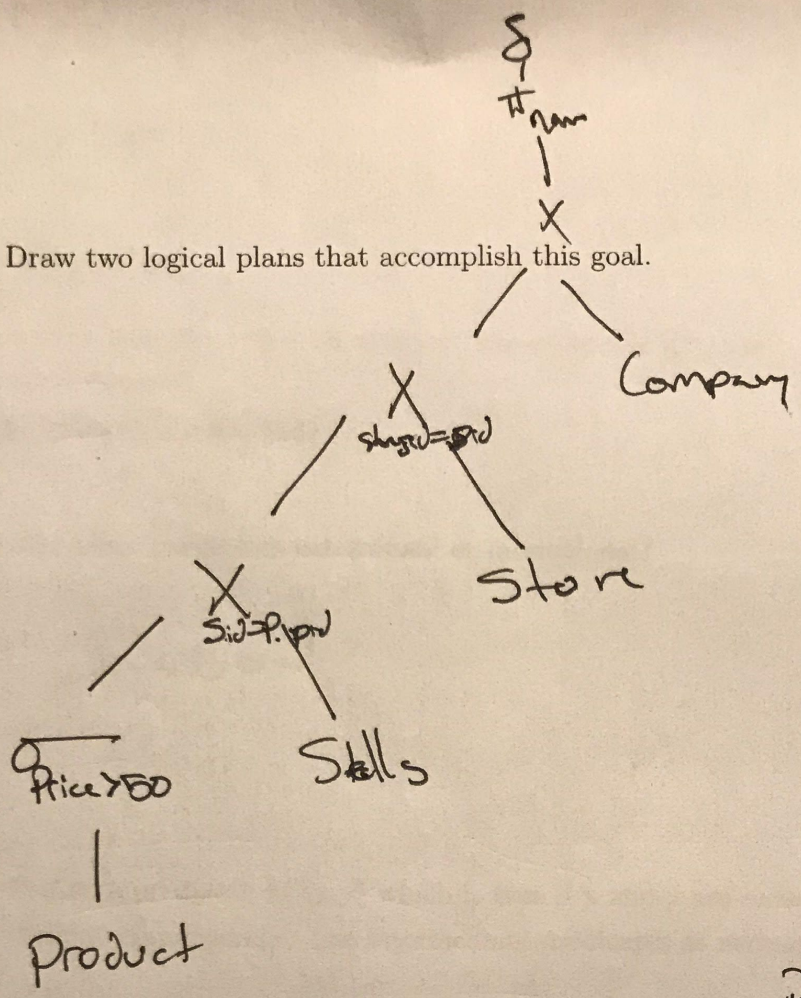
a) Provide the E/R diagram for this table. Assume ~~all relations~~ Sells are many-to-many.

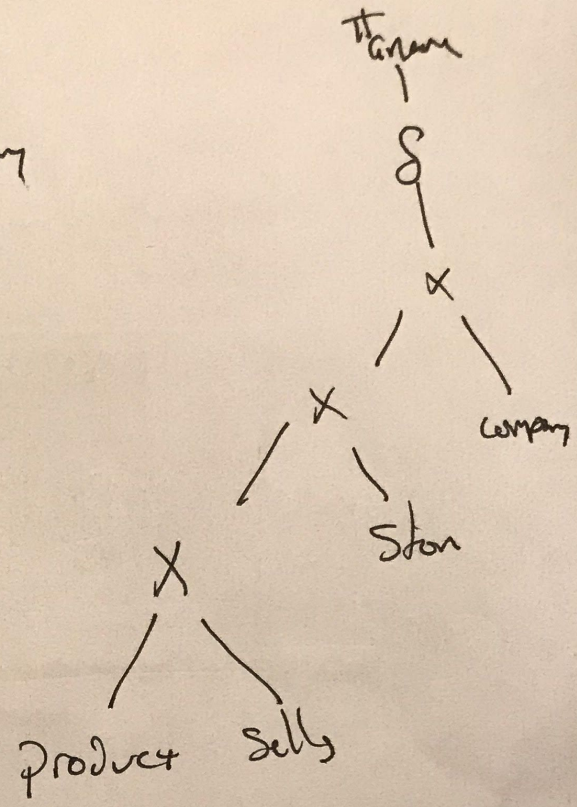


b) Write a SQL query to find the names of all companies who sell a product that costs more than $50. You may assume price is an INT.

```
Select distinct C.name
From Product as P
  Join Sells as Se On
    Se.pid = P.pid
  Join Store as St on
    St.sid = Se.sid
  Join Company as C on
    C.cid = St.cid
where Se.price > 50
```

c) Draw two logical plans that accomplish this goal.





d) Name four pieces of information you would need to find the cheapest physical plan, in terms of disc accesses.

# of Blocks in Product, Sells Store, Company

Range of price

# of Tuples

3 Datalog

Suppose we have the following predicates in a datalog database:
`Person(pid,name)`
`Parent(parentid,childid)`

a) Are these predicates extensional or intenstional?

Intensional

b) Define a predicate SC(x,y) which is true if x and y are second-cousins, i.e. they share a great-grand parent. Use intermediate predicates as necessary.

SC(x,y) :- Parent(z,x), Parent(a,z), Parent(b,a),
Parent(c,y), Parent(d,c), parent(e,d),
Person(x,_), Person(y,_) [, Possibly x!=y
if you assume
- x,y cannot be the same person]

c) Can this be solved using SQL? Why or why not?

Yes, since there are only a finite # of steps that need to be iterated through to find Second-cousins

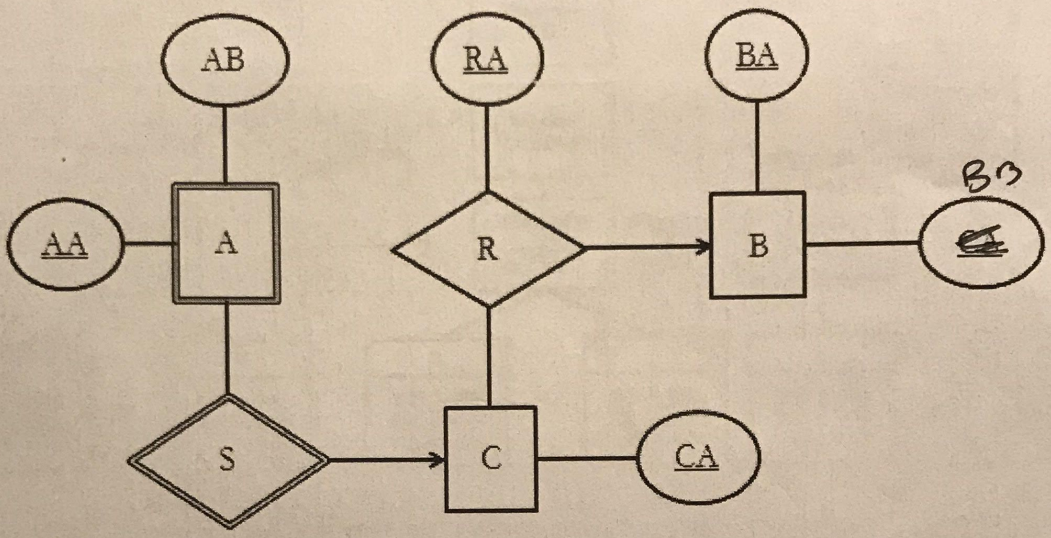
d) Define a predicate cousins(x,y,n) where x and y are people and n is the level of their relationships. Siblings should have n = 0, cousins have n = 1, second cousins n=2 and so forth. For any pair of people (x,y) only one tuple cousins(x,y,n) should exist, even though first cousins must also share great-grandparents.

cousins(X, Y, 0) :- person(X,_), person(Y,_), parent(X, a)
parent(Y, a), X != Y

cousins(X, Y, n+1) :- person(X,_) person(Y,_) parent(X, a)
parent(Y, b), cousins(a, b, n+1),
! cousins(X, Y, Z), Z ≤ n

4 E/R Diagrams

Produce a schema for the following E/R Diagram. Indicate if there are any foreign keys or if anything must be non-null.



double underline indicates fk

A(AA, AB, CA)

B(BA, BB)

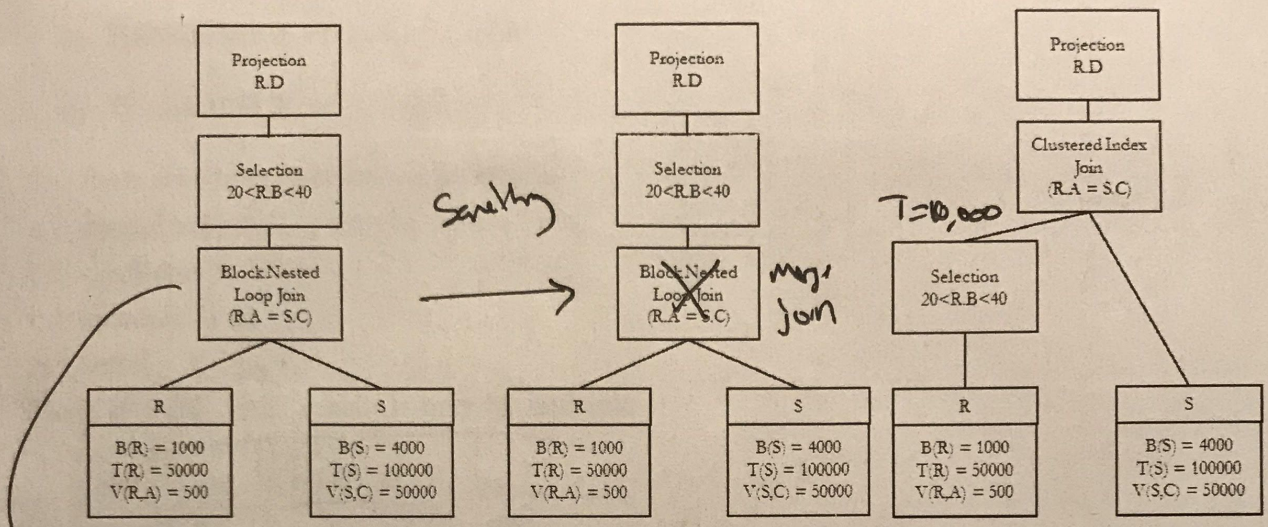
~~C(CA)~~ C(CA, RA, BA)

~~B3~~

5 Cost Estimation

Need N

Give the cost for the following physical plans. Assume the following clustered indexes (R.A and S.C) and no unclustered indexes:



1.) $B(R) + \frac{B(R)\,B(S)}{M-1}$

$P(1) \quad 1000 + \frac{4000000}{M-1}$

2.) Since merge join assume it fits in memory (unrealistic)

$B(R) + B(S) = 5000$

3.) need min/max for R.B (assume 0–100)

B(R) to read into table + #tuples after selection

$1{,}000 + 10{,}000 = 11k$

6 Transactions

Suppose a scheduler is trying to schedule the following transactions.

a) Transaction 1: $r(A), w(B), r(A)$

b) Transaction 2: $r(B), w(B), w(A)$

c) Transaction 3: $w(A), w(B)$

For each schedule, indicate whether it is:
a.) a valid schedule given the transactions
b.) conflict-serializable,
c.) serializable or
d.) serial
Keep in mind, each schedule may be multiple.

a) $r_1(A), w_3(A), r_2(B), w_2(B), w_1(B), w_2(A), r_1(A), w_3(B)$

a.

2 → 1 → 3

b) $r_2(B), r_1(A), w_1(B), w_2(B), w_2(A), r_1(A), w_3(A), w_3(B)$

a.

2 → 1

c) $r_1(A), r_2(B), w_3(A), w_1(B), w_2(B), w_3(B), r_1(A), w_2(A)$

a.

d) Which of the above schedules results in a dirty or inconsistent read? Which transactions are affected?

a.) T_1 ∈ inconsistent read

b.) T_1 ∈ inconsistent read

c.) T_1 ∈ inconsistent read

3 has no reads
2 has one read which never comes after a write