1 Short Answer

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

If an attribute is a foreign key, that means it references an attribute in another table.

This attribute must be either NULL or a value in that other table's column.

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

Create Dataverse

Create Type

Create Dataset

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

No. Each tuple in the primary table is compared against each tuple in the second. Because all cross-wise comparisons are made, no indexes are needed

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~~ estimated minimum 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.

Selectivity factor $\frac{60-30}{80--20} = \frac{30}{100}$

Estimated Tuples $= T(R) \cdot \frac{30}{100} = 50{,}000 \frac{30}{100}$

$= 15{,}000$

Because the table is unclustered, each would require its own I/O.

Since there are only 1000 blocks, it would be faster to read all blocks w/pipelined selection

1000

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

Since all reduce tasks must wait for all Map tasks, a straggler in map can more seriously impact total runtime

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.

Semi-structured data allows joins on nested collections, whereas this is not allowed in INF

g) In our semi-structured data, what is the difference between a dataverse and a dataset?

A dataset is a single relation containing objects of a type.

A dataverse contains all datasets in the reference data.

h) If a database has queries that are analytical, rather than transactional, should we prefer duplication or partitioning?

Duplication

→ harder to maintain throughput if there are many writes, analytical is read-heavy

→ distributing data means the DB can distribute query requests, increasing throughput

i) Suppose we have the schema `R(a,b) S(b,c)`. When are R.a and S.c `NULL` in a `FULL OUTER JOIN` of R and S?

1.) R.b has no match S.b and R.a is NULL

2.) S.b has no match R.b and S.C is NULL

3.) There is a match R.b = S.b but R.a and S.c are null.

j) In NoSQL (semi-structured) data bases, what is heterogeneity and why does the problem arise?

Heterogeneity is when an attribute's type varies from object to object.

This is possible because semi-structured data does not enforce types as strongly as SQL.

k) Why would a DBMS prefer uncorrelated subqueries?

If a subquery is uncorrelated, it only needs to be run once.

Correlated subqueries must be re-run for each matching value.
This results in longer runtime.

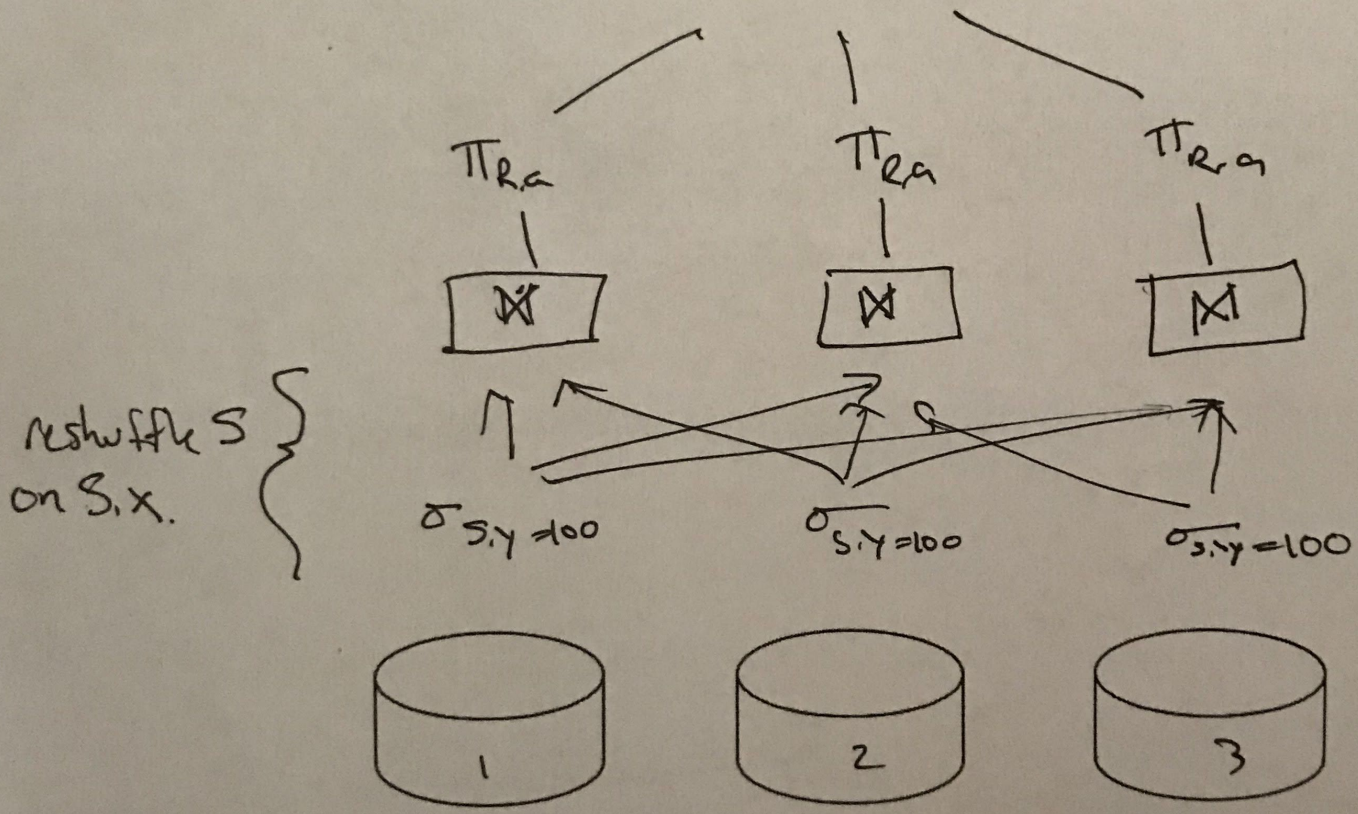
l) Give an example of a safe, non-linear recursive datalog rule.

Path(x,y) :- Edge(x,y)
Path(x,y) :- Path(x,z), Path(z,y)

m) We have the schema R(a, b), S(x, y). Given the following query, draw a parallel query plan that reduces the amount of **data** sent over the interconnection network. Assume all attributes are INTs and B(R) = B(S).

```
SELECT R.a
FROM R, S
WHERE R.b = S.x AND
      S.y = 100
```

R has been hash partitioned on R.b and S is block partitioned



Since R is hash partitioned on R.b, the data in R does not need to be reshuffled.

2 SQL

For this question, use the following schema:

`Company(cid,country,name)`

`Product(pid,name,description)`

`Store(sid,cid,location)`

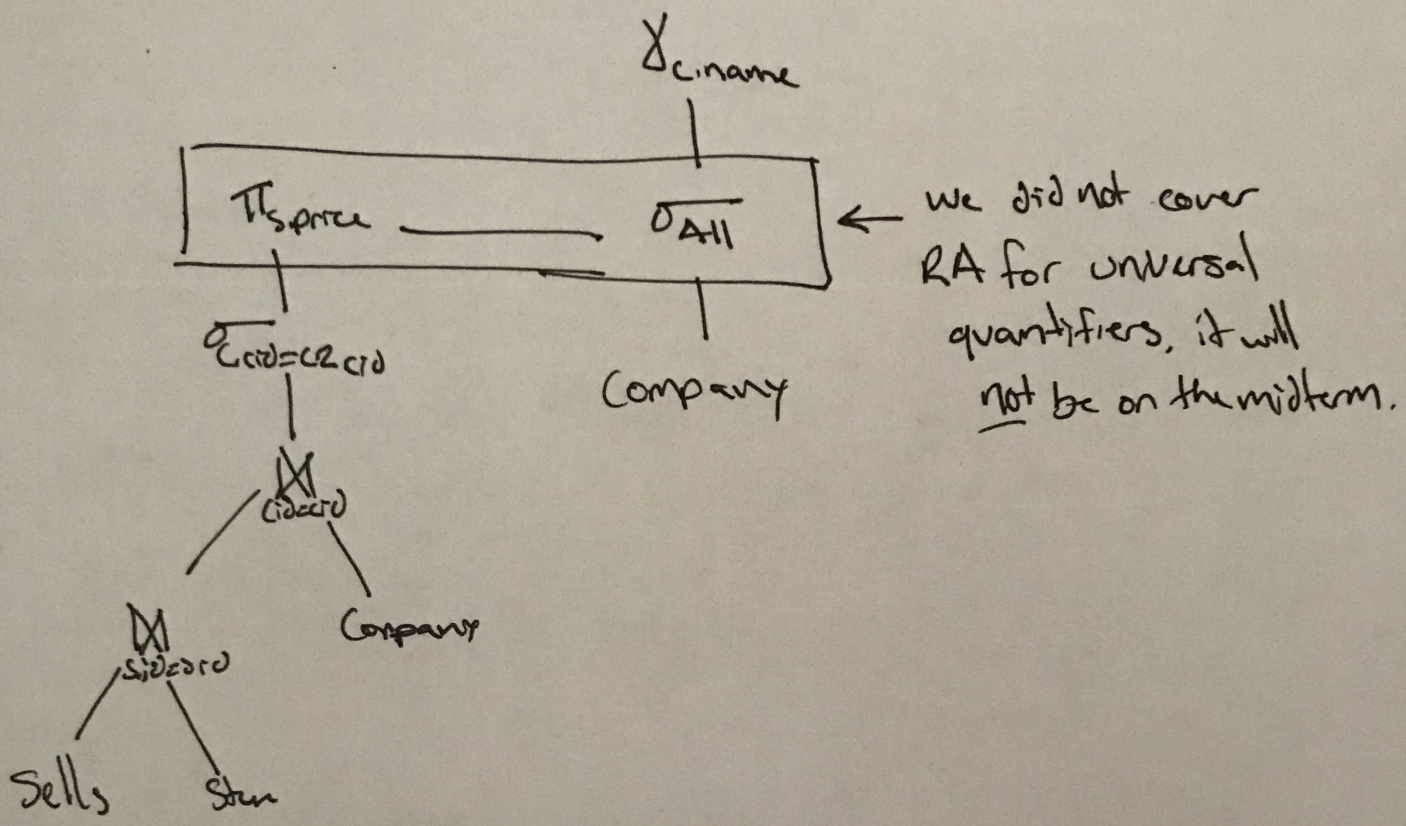
`Sells(sid,pid,price)`

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

```
Select Distinct C.name
From Company as C, ~~Store as Sto~~

where 50 > All (Select S.price
                From Sells as S, Store as St, Company as C2
                Where S.sid = St.sid)
                      St.cid = C2.cid
                      C2.cid = C.cid
```

b) Draw two logical plans that accomplish this goal.



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

~~III~~ B(Company) B(Store) B(Sells)

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?

Extensional

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. ← assume x can be y and that x any are ids

not necessary if all parent relations are people

SC(x,y) :- person(x,a), person(y,b), parent(i,x), parent(j,y)

parent(k,j), parent(l,y), parent(m,l), parent(k,l).

common great grandparent

Alternate, non stratified

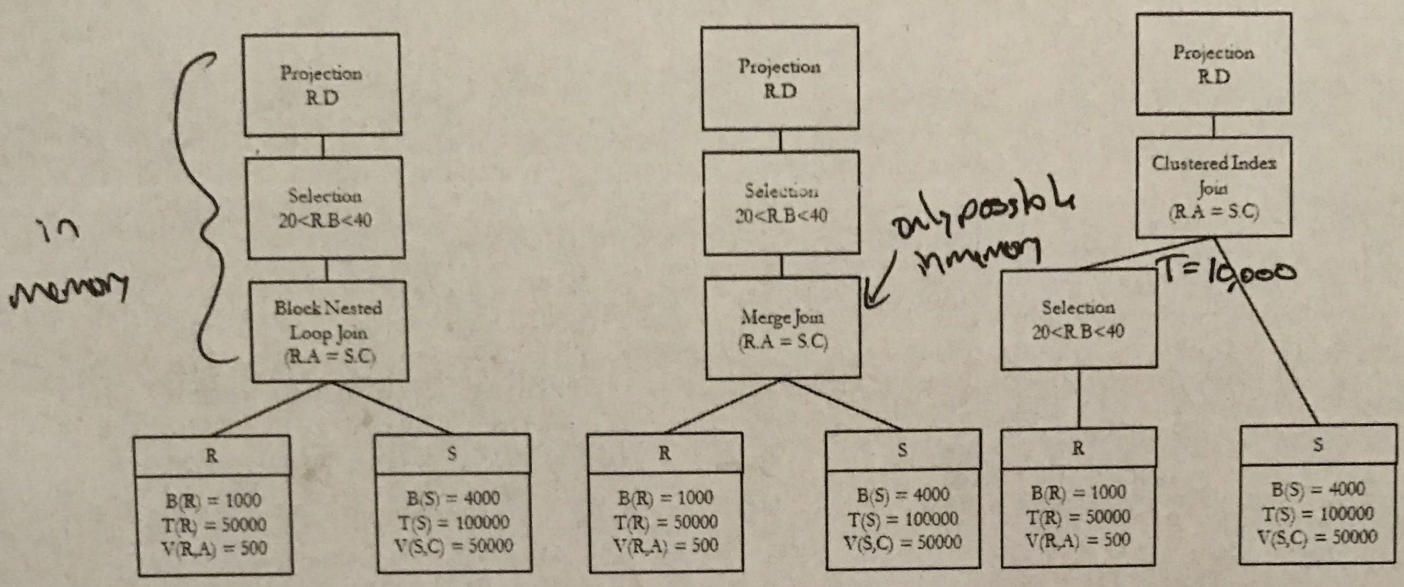
Siblings(x,y) :- parent(a,x), parent(a,y).

Cousins(x,y) :- parent(a,x), parent(b,y), siblings(a,b).

SC(x,y) :- parent(a,x), parent(b,y), cousins(a,b).

4 Cost Estimation

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



in memory

only possible in memory

T = 10,000

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

$1000 + 1000 \cdot 4000 =$

4001,000

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

$1000 + 4000 =$

5000

Unlikely to fit in memory

Clustered selection

Selectivity $\frac{40-20}{100-0} = \frac{20}{100}$

$I/Os = \frac{20}{100} \cdot B(R) = 200$

Index Join

10,000 Tuples from left, each joins w $\frac{T(S)}{V(S,C)}$ tuples from right $= \frac{100,000}{50,000} = 2$ tuples

$10,000 \cdot 2 = 20,000$

total $= 20,200$