Database Management Systems

Query Planning

The manner in which tables are filtered, sorted, and joined affects execution time.

Select the parents of curly-furred dogs:

```
select parent from parents, dogs
where child = name and fur = "curly"
```

Join all rows of parents to all rows of dogs, filter by child = name and fur = "curly"

Join only rows of parents and dogs where child = name, filter by fur = "curly"

Filter dogs by fur = "curly", join result with all rows of parents, filter by child = name

Filter dogs by fur = "curly", join only rows of result and parents where child = name

Local Tables

A `create table` statement names a table globally.

```
create table parents
as
select "abraham" as parent, "barack" as child union
select "abraham" , "clinton" union
select "delano" , "herbert" union
select "fillmore" , "abraham" union
select "fillmore" , "delano" union
select "fillmore" , "grover" union
select "eisenhower" , "fillmore";
```

A `with` clause of a `select` statement names a table that is local to the statement.

```
with best(dog) as {
  select "eisenhower" union
  select "barack"
}
select parent from parents, best
where child=dog;
```
Example: Relationships

(A) What are appropriate names for the columns in this result?
(B) How many rows and columns will result?

```sql
WITH
  (first, second) AS {
    SELECT a.child, b.child
    FROM parents AS a, parents AS b
    WHERE a.parent = b.parent AND
      a.child != b.child
  }
SELECT child AS _____, second AS _____
FROM parents, (select a.child, b.child
FROM parents AS a, parents AS b
WHERE a.parent = b.parent)
```

Local Tables can be Declared Recursively

An ancestor is your parent or an ancestor of your parent.

```sql
CREATE TABLE parents AS
SELECT "abraham" AS parent, "barack" AS child
UNION ...
```

Global Names for Recursive Tables

To create a table with a global name, you need to select the contents of the local table.

```sql
CREATE TABLE odds AS
WITH
  odds(n) AS
  (SELECT 1 UNION SELECT n + 2 FROM odds WHERE n < 15)
SELECT n FROM odds;
```

Limits on Recursive Select Statements

Recursive table definitions are only possible within a with clause.

```sql
WITH
  odds(n) AS
  (SELECT 1 UNION SELECT n + 2 FROM odds WHERE n < 15)
SELECT n FROM odds;
```

Language is Recursive

Noun phrases can contain relative pronouns that introduce relative clauses.

```
The dog chased the cat
  that chased the bird
The dog chased the cat
  that the bird chased
The dog the bird that chased chased chased me
Bulldogs bulldogs bulldogs fight fight fight
```
A table containing the inputs to a function can be used to map from output to input.

For example:

```sql
create table pairs as
with
  i(n) as (
    select 1 union
    select n+1 from i where n < 50
  )
select a.n as x, b.n as y
from i as a, i as b
where a.n <= b.n;
```

What integers can I add/multiply together to get 24?

(Demo)

Example: Pythagorean Triples

All triples a, b, c such that $a^2 + b^2 = c^2$

```sql
with
  i(n) as {
    select 1 union
    select n+1 from i where n < 20
  }
select a.n as a, b.n as b, c.n as c
from __________________________________________
    where ______________________ and a.n * a.n + b.n * b.n = c.n * c.n;
```

(Demo)

Example: Fibonacci Sequence

Computing the next Fibonacci number requires both the previous and current numbers.

```sql
create table fibs as
with
  fib(previous, current) as {
    select 0, 1 union
    select current, previous + current from fib
    where current <= __________________________
  }
select _____________________________ as n
from fib;
```

(Demo)

A Very Interesting Number

The mathematician G. H. Hardy once remarked to the mathematician Srinivasa Ramanujan...

(Demo)