# CS 61A: Solutions for Quiz 3

*Due by 11:59pm on Thursday, 12/4*

**Solutions:** You can find the Python file with solutions for all questions here.

Quiz submissions were graded automatically for correctness. Implementations **did not** need to be efficient, as long as they were correct.

In addition to the doctests provided to students, we also used extra doctests to check for corner cases. These extra test cases are highlighted below.

**Readings:** You might find the following references useful:

- Section 4.3

## Table of Contents



To complete this quiz, you will need to use Sqlite version 3.8.3 or greater. You can

- Download Sqlite3 and install it on your own computer
- Run `sqlite3` from an instructional machine using your course account
- Use this online Sqlite interpreter

After installing Sqlite, you can run the starter file using the command:

```
sqlite3 -init quiz3.sql
```

### Data

In each question below, you will define a select statement that processes the following tables, which **include an additional dog named Jackson**!

```
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"        union
  select "delano"           , "jackson";

create table dogs as
  select "abraham" as name, "long" as fur, 26 as height union
  select "barack"         , "short"      , 52           union
  select "clinton"        , "long"       , 47           union
  select "delano"         , "long"       , 46           union
  select "eisenhower"     , "short"      , 35           union
  select "fillmore"       , "curly"      , 32           union
  select "grover"         , "short"      , 28           union
  select "herbert"        , "curly"      , 31           union
  select "jackson"        , "long"       , 43;
```

Your select statement should still perform correctly even if the values in these tables are changed. For example, if you are asked to list all dogs with a name that starts with h, you should write:

```
select name from dogs where "h" <= name and name < "i";
```

Instead of assuming that the `dogs` table has only the data above and writing

```
select "herbert";
```

The former query would still be correct if the name `grover` were changed to `hoover` or a row was added with the name `harry`.

## Question 1

(*1 point*) Write a SQL query that selects all possible combinations of three dogs with the same fur and lists them in order of increasing height.

```
-- All triples of dogs with the same fur that have increasing he

select "=== Question 1 ===";
select a.name, b.name, c.name from dogs as a, dogs as b, dogs as
  where a.height < b.height and
        b.height < c.height and
        a.fur = b.fur and b.fur = c.fur;

-- Expected output:
--   abraham|delano|clinton
--   abraham|jackson|clinton
--   abraham|jackson|delano
--   grover|eisenhower|barack
--   jackson|delano|clinton
```

Your output table should have three columns. You should assume that all dogs have different heights and different names. The rows of the output can appear in any order.

## Question 2

(*1 point*) Write a SQL query that selects the sum of the heights of at least 3 dogs with the same fur, ordered by the total sum. Each dog should be used at most once in a sum.

```
-- The sum of the heights of at least 3 dogs with the same fur,

select "=== Question 2 ===";
with
  sums(furs, sum, max, n) as (
    select fur, height, height, 1 from dogs union
    select fur, sum+height, height, n+1 from dogs, sums
      where fur = furs and max < height
  )
select furs, sum from sums where n >= 3 order by sum;

-- Expected output:
```

```
--   long|115
--   short|115
--   long|116
--   long|119
--   long|136
--   long|162
```

You should assume that all dogs have different heights and different names. Rows with the same total sum may appear in any order.

*Hint*: This question is similar to Homework 10 Question 4.

*Hint*: A sum of 162 can be reached by adding the heights of abraham, clinton, delano, and jackson.

## Question 3

(*1 point*) Recall the sequence from Homework 3:

```
g(n) = n,                                          if n <= 3
g(n) = g(n - 1) + 2 * g(n - 2) + 3 * g(n - 3), if n > 3
```

Write a query that lists out the first `20` terms of g in order. Use a recursive table; queries that explicitly list out terms of g other than 1, 2, and 3 will be marked as incorrect.

```
-- The terms of g(n) where g(n) = g(n-1) + 2*g(n-2) + 3*g(n-3) a

select "=== Question 3 ===";
with
  g(prev1, prev2, curr, n) as (
    select 1, 2, 3, 1 union
    select prev2, curr, 3*prev1+2*prev2+curr, n+1 from g where n
  )
select prev1 as n from g;

-- Expected output:
--   1
--   2
--   3
```

```
--   10
--   22
--   51
--   125
--   293
--   696
--   1657
--   ...
--   9426875
```

*Hint*: This question is similar to the Fibonacci example from the textbook.

**Final Note**: Please **do not** add any statements that create output in addition to the expected output.