Carnegie Mellon University
Department of Computer Science
15-826 Multimedia and Data Mining
C. Faloutsos, Fall 2025

Homework 2 - Solutions

Due: on canvas, at **2:00pm, on Fri 09/19/2025**

VERY IMPORTANT:

- Deposit **pdf copy** of your answers, on canvas.

Reminders:

- *Plagiarism*: Homework is to be completed *individually*.
- *Typeset* your answers. You may use the pdf of the handout, to type/circle your answers. Illegible handwriting may get zero points.
- *Late homeworks*: Please follow the instructions here

For your information:

- Graded out of **100** points; **4** questions total
- Rough time estimate: *2-6 hours*
- Weight: 3% of total course grade.

Revision : 2025/09/23 16:43

Question	Points	Score
Warm-up: duckdb	60	
Z-/Hilbert ordering	15	
R-trees	15	
Graph patterns	10	
Total:	100	

Question 1: Warm-up: duckdb ..... [60 points]

The goal is to become familiar with the properties of duckdb: it is able to answer sql queries on compressed csv files, and even do joins on two of them.

Reminder: The phonecall data etc are all in the box directory We shall refer to it as **DATA_HOME** in this question.

Data As mentioned in the canvas discussion, you are given a script that anonymizes the long sha_hashcodes, into 1-n integers.

- *original file*: For this homework we will use the data of the first two days of November: DATA_HOME/01_data/CMU_2021-11-01_02_cmu.csv.gz We shall refer to it as the *original* file.
- *trimmed file*: For your convenience, we run the anonymizer script and put the result into file DATA_HOME/02_subsets/CMU_2021-11-01_02_trimmed.csv.gz has such short integers for callers and call-examples (and we also omited a few not-so-useful columns).
- *lookup table*: The anonymizer script also produced a look-up table DATA_HOME/02_subsets/CMU_2021-11-01_02_lookup.csv.gz with the obvious two columns: original sha256_code, and our integer id).

Sample queries: Also for your convenience, we provide some queries at DATA_HOME/04_auxiliary/duckdb_for_hw2, with a 'makefile' and self-explanatory names.

Your tasks:

(a) **[10 points]** Run the query *04_outCalls.sql*, and report the integer-ids and number of phonecalls for the top 10 heavy-hitter, that is, the ones with the most out-going phonecalls.

Solution:

source_hash	num_outCalls
172815	2250906
8056	113442
493790	100235
366362	3587
748474	3536
801034	2410
510128	2106
2236151	2101
748285	2099
800816	2082

(b) **[10 points]** Run the query *11_revealHeavyHitters.sql*: That is, as above, but reveal their sha256 codes (instead of our integer-ids). Report the results.

Question 1 continues...

Solution:

sha_hash	num_outCalls
dc1420f83600f7a2ac7cbd9c46fe4e7efb9643e05130535a8cce460851aa9652	2250906
ed3366fda3d8fc9346e6f68d932f334e9425cee892c04b846405e97d1526ea5a	113442
4b340cc617af3db416a232eaee9a08ce6722ee5799e02ec810a82cc6c31b55de	100235
2c945e129f8623dd3b6b28d3cf719a5e98d425e687ecb5b62c970f8349ce3fb8	3587
c2a99544a6fb99a6afb9252bc121c03b15ccd53a2288d457b2c08b372e593e5b	3536
2acf34f3164153a6c81c1303f8afca07dd1210834c62b4de8d26c2c0dffee7ac	2410
d06cf5a2b539c3486bfaf97e03ae190099fbe622b82d293850f347764f945066	2106
114af3b756777ef3bbc9adbae79fbcc510bd0451bf5cb84f9e4f6cd4df923425	2101
7d8ea1a7e394e5b67469fb3131b63e6eeacbbd4d537be4532169d26da369b8c9	2099
ab6c688a87b2fe880db3468681e495f3d9cbf563322d9567c0bcc9f578246aec	2082

(c) **[2 points]** What was the wall-clock time of the above query?

Solution: 23.78 real 46.08 user 4.81 sys (macMini 2018, i3, 16Gb, Sequoia)
Grading info: any answer is fine

(d) **[10 points]** Run the query *12_HeavyHitters_from_original.sql*. Report the results.

Solution:

source_hash	num_outCalls
dc1420f83600f7a2ac7cbd9c46fe4e7efb9643e05130535a8cce460851aa9652	2250906
ed3366fda3d8fc9346e6f68d932f334e9425cee892c04b846405e97d1526ea5a	113442
4b340cc617af3db416a232eaee9a08ce6722ee5799e02ec810a82cc6c31b55de	100235
2c945e129f8623dd3b6b28d3cf719a5e98d425e687ecb5b62c970f8349ce3fb8	3587
c2a99544a6fb99a6afb9252bc121c03b15ccd53a2288d457b2c08b372e593e5b	3536
2acf34f3164153a6c81c1303f8afca07dd1210834c62b4de8d26c2c0dffee7ac	2410
d06cf5a2b539c3486bfaf97e03ae190099fbe622b82d293850f347764f945066	2106
114af3b756777ef3bbc9adbae79fbcc510bd0451bf5cb84f9e4f6cd4df923425	2101
7d8ea1a7e394e5b67469fb3131b63e6eeacbbd4d537be4532169d26da369b8c9	2099
ab6c688a87b2fe880db3468681e495f3d9cbf563322d9567c0bcc9f578246aec	2082

(e) **[3 points]** Report the wall-clock time of the above query. Was it slower than the query on the *trimmed* file?

Solution:
48.01 real 76.06 user 3.39 sys (again, macMini 2018, i3, 16Gb, Sequoia)
Grading info: any answer is fine

(f) **[10 points]** Write a query to find the top 10 most talkative callers - report their sha256 code, and their total duration of their (out-going) phonecalls. Again, sort by duration descending; break ties by sha256 code ascending.

Solution:

sha_hash	out_duration
dc1420f83600f7a2ac7cbd9c46fe4e7efb9643e05130535a8cce460851aa9652	143074976
4b340cc617af3db416a232eaee9a08ce6722ee5799e02ec810a82cc6c31b55de	1684943
ed3366fda3d8fc9346e6f68d932f334e9425cee892c04b846405e97d1526ea5a	1570129
2c945e129f8623dd3b6b28d3cf719a5e98d425e687ecb5b62c970f8349ce3fb8	200821
2acf34f3164153a6c81c1303f8afca07dd1210834c62b4de8d26c2c0dffee7ac	169308
7e2f27be1f95c11249d87f7b379546f2f3860ba2fd82978a35043dd2c6b3e7e6	146862
ab6c688a87b2fe880db3468681e495f3d9cbf563322d9567c0bcc9f578246aec	141882
c0c55c81f88e69a69696fd1983bc9dac1207a1622ae4f892b5c94efa5286a126	126347
010dee70105709c9faa42bb425e16bd8d439a196eff7f97ef92e16ea19c3b68a	108022
87e709813a3aeac3e535009b7a1153401e6fa9fccdbc260ed45e3c38cd0334f4	102878

(g) **[15 points]** Give the SQL (duckdb) code of your 'talkative callers' query.

Solution:

Fast version, using the look-up table: (24.66 real 47.58 user 4.64 sys; macmini 2018)

```
-- read-in the lookup table, and give names to its two columns
create table lookup ( sha_hash varchar, our_id int);
copy lookup from 'lookup.csv.gz';

select lookup.sha_hash, sum(call_dur) as out_duration
from read_csv('phonecalls.csv.gz')  -- we read it on the fly!
    join lookup
    on lookup.our_id = source_hash
group by lookup.sha_hash
order by out_duration desc, lookup.sha_hash
limit 10;
```

Slow version (52.04 real 79.05 user 5.24 sys; macmini 2018):

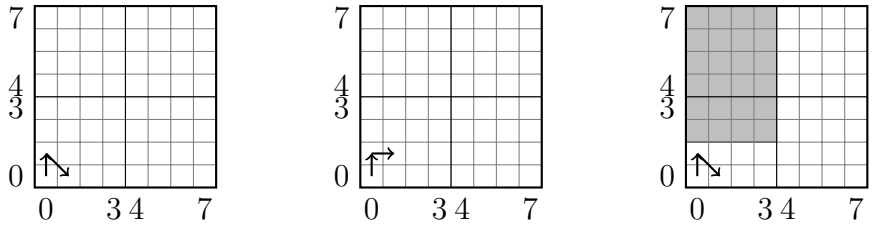
```
select source_hash, sum(call_dur) as out_duration
from read_csv('phonecalls_full.csv.gz') --we read it on the fly!
group by source_hash
order by out_duration desc, source_hash
limit 10;
```

Question 2: Z-/Hilbert ordering . . . . . [15 points]

Consider a $2^n \times 2^n$ grid, and the z-curve on it. As usually, its first step is *vertical*, that is:

- the (0,0) cell has decimal z-value = 0
- the (0,1) cell is next, with decimal z-value = 1

Figure 1(a) shows the first two steps (arrow) of such a z-curve, on an 8×8 grid (which obviously has ranges (0,7) × (0,7)).



(a) z-values (b) hilbert-values (c) z-values for shaded area

Figure 1: Grids, for z- and hilbert-values

(a) **[2 points]** Which is the cell with the highest z-value? (Give (x,y) coordinates, like, e.g., (3,3))

(a) **(7,7)**

(b) **[5 points]** Which is the cell with the highest **hilbert**-value? (Figure 1(b) shows the first two steps of the curve).

(b) **(7,0)**

(c) **[3 points]** How many z-values do we need for the shaded area of Figure 1(c)? (Recall that '*'s can only be at the end - that is, e.g., 01**01 is not valid.)

(c) **3**

Grading info: [-1] if '2'

(d) **[5 points]** Give the z-value(s) for the shaded area of Figure 1(c)

Solution: 01-**-**, 00-01-**, 00-11-**

Grading info: [+1] for first; [+2] for each of the longer ones

*Grading info: [-3] if 01-**-** 00-1*-**

Question 3: R-trees ..... [15 points]

The foils, and Pagel's formula, refers to intersection queries. Here, we focus on *inclusion* queries: A query Q (shaded rectangle in Figure 2) would retrieve all the rectangles that completely contain it, like rectangle A.

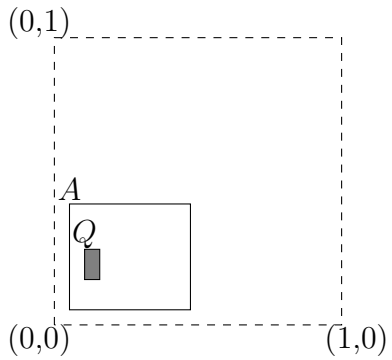


Figure 2: Example of inclusion query, in 2-d: the white rectangle A includes the shaded query rectangle Q, and thus it qualifies under the inclusion query with Q.

Consider a **3-d** setting. Consider a parent node P in an R-tree, whose MBR has sides x_1=0.2, x_2=0.3 and x_3=0.4. Consider also a query Q with sides (q_1, q_2, q_3). Assume the following:

- Everything is in the unit **cube**
- Assume that $x_i > q_i$ $(i = 1, 2, 3)$
- As in the foils, the query center is uniformly distributed in the available space, with wrapping around (as Pagel's formula also assumes).

(a) **[2 points]** What is the probability that parent P will be retrieved, under a point query $(q_1 = q_2 = q_3 = 0)$

(a) **0.024**

Grading info: *0.2*0.3*0.4 - Pagel's formula*

Grading info: *-1pt if not multiplied out, but correct otherwise.*

(b) **[5 points]** What is the formula for the probability that a parent node with dimensions x_1, x_2, x_3, will completely contain the query box Q with dimensions q_1, q_2, q_3.

A. $(x_1 + q_1) * (x_2 + q_2) * (x_3 + q_3)$
B. $(x_1/q_1) * (x_2/q_2) * (x_3/q_3)$
C. $(q_1/x_1) * (q_2/x_2) * (q_3/x_3)$
D. $(x_1 - q_1) * (x_2 - q_2) * (x_3 - q_3)$
E. $(q_1 - x_1) * (q_2 - x_2) * (q_3 - x_3)$
F. None of the above - the correct formula is: .....

(c) **[8 points]** Compute the probability that the query Q with sides $q_1 = q_2 = q_3 = 0.1$, will be completely inside parent node P (x_1=0.2, x_2=0.3 and x_3=0.4)

Give the exact, *numerical* answer.

(c) **0.006**

Grading info: $(x_1 - q_1) * ...$ = *0.1*0.2 *0.3*

Question 4: Graph patterns ..... [10 points]

(a) **[4 points]** Suppose you are monitoring a million-scale un-directed graph like FaceBook (who-isFriendsWith-whom), with new nodes and edges added over time. The diameter $D(t)$ at month t was 3, 5, 15, 9 ($t = 1, \ldots, 4$). According to the book and the lecture notes, what will the value of the diameter $D(5)$ be, on the next month ($t = 5$)?

A. 15 *# it will go back to its highest value*
B. between 9-15 *# it will almost go back, but a bit lower*
C. 9 *# it will stay there*
D. 3 *# it will keep on dropping with the same rate*
E. between 5-9 *# it will keep dropping, approaching ≈6.*
F. 2-3 *# it will densify - graph is clearly beyond 'gelling' point*
G. something else - explain: .....

(b) **[3 points]** Consider a different graph, with N isolated nodes (ie, E=0 edges). What is its diameter? (Reminder: the radius of a node is the distance to the most remote (but reachable) node; the diameter of a graph is the maximum radius over all nodes.)

(b) **0**

(c) **[3 points]** We have yet-another graph with N' nodes. What is the smallest number of edges E that will make the graph to have diameter D=1?

(c) **1 edge**

Grading info: [-1] if $(N')^2$ *or* $\binom{N'}{2}$*, ie, they go for a clique.*