Technion – Israel Institute of Technology

Faculty of Data and Decision Sciences

0960224 – Distributed Database Management

Spring 2025

Homework Assignment 1: MapReduce, Spark, and Relational Query Optimization

General Guidelines:

1. **Due Date:** 21/05/2025 at 23:55 - with a two day automatic extension.
2. Submission **in pairs** only (unless permitted otherwise) and to be submitted in the designated submission box in moodle **by only one of the members**.
3. Your code MUST be documented, and each code cell must be briefly explained afterwards in a markdown text cell.
4. The solution to Part C must be **typed** using a word processor (Word, LaTeX, etc.)
5. **DO NOT submit a zip folder.** You are required to submit **6 files in total**. Wherever id1,id2 is mentioned, it refers to the ID numbers of the submitting students.

Detailed submission information – and a submission checklist for you when you're done:

- **Parts A and B**: MapReduce and Spark

    1. 'hw1_code_id1_id2.ipynb' – a single Jupyter notebook file with parts A and B clearly separated using markdown headings.
    2. 'hw1_mrA1_id1_id2.py' – the python file generated by the first MRJob.
    3. 'hw1_mrA2_id1_id2.py'– the python file generated by the second MRJob.
    4. 'hw1_code_id1_id2.html' – the HTML version of your notebook (including results and outputs).
    5. 'hw1_code_id1_id2.pdf – the PDF version of your notebook (including results and outputs).

- **Part C:** Relational Query Optimization

    1. 'hw1_dry_id1_id2.pdf' – a PDF file containing your typed solution.

Tip: We advise you to read **the entire** assignment before you start attempting to solve it.

Introduction for Parts A and B

You two are part of the final three contestants on the hit reality TV show *Big Data Brother*. To secure your spot in the grand finale and avoid eviction from the *Big Data Brother* House, you must team up and complete two final "Head of Household" competitions.

The *Big Data Brother* is always watching and knows that you have been taking the Distributed Database Management class, so he decided to give you the following dataset for some MapReduce and Spark tasks – the '440k_data.csv' dataset, which has the following schema:

- **'title'** – Program title of a movie, show, episode, or sports event.
- **'prog_code'** – Unique program identifier. (episode number for example)
- **'genre'** - Comma separated words or groups of words that classifies a show, episode, movie, or sports event.
- **'air_date'**- Air date of the program (YYYYMMDD).
- **'air_time'**- Air time of the program (HHMMSS).
- **'Duration'**- Duration of the program in minutes.

Important Notes:

1. A **program** refers to a TV show, movie, sports event, etc.
2. The following rows can (and do) exist in the data:

title	prog_code	genre	air_date	air_time	Duration
Blaze and the Monster Machines	EP020169630012	Children,Fantasy,Adventure,Educational,Animated	20151227	183000	30.0
Blaze and the Monster Machines	EP020169630024	Children,Fantasy,Adventure,Educational,Animated	20151101	3000	30.0
Blaze and the Monster Machines	EP020169630017	Children,Fantasy,Adventure,Educational,Animated	20151130	3000	30.0
Blaze and the Monster Machines	EP020169630017	Children,Fantasy,Adventure,Educational,Animated	20151217	80000	24.0
Blaze and the Monster Machines	EP020169630011	Children,Fantasy,Adventure,Educational,Animated	20151113	32400	24.0

Notice how the same title can have different 'prog_code' values because they are viewings of different episodes of this program.

Part A: MapReduce (30 points)

For your first challenge the *Big Data Brother* gave you the following task:

A "*Big Data Brother* APPROVED program" is a program that satisfies the following criteria:

mappe

- **Criteria 1**: Has (there exists) showings (airings) that start between 13:30 and 16:30 (including 13:30 and excluding 16:30). 133000 ≤ air_time < 163000
- **Criteria 2**: Has at least one of the following genres: if
  `['Reality', 'Community', 'Adventure', 'Animated']`

  נבדוק את הרשימה
  אם קיים עדכון
  לא כלום אם לא

mapper

- **Criteria 3**: Its title has at least two of the following letters (case insensitive):
  `['p', 'w', 'm']`

  הגיעה מהמו מגיע
- Criteria 4: Its title does not have any of the following letters (case insensitive):
  `['a','b']`

  if

The high-level of the challenge is as follows:

For every "*Big Data Brother* APPROVED program" you will return:

1. Its 'title'.
2. A list of its genres that satisfy criteria 2.

reducer

3. The number of (unique) dates of viewings of that program that satisfy criteria 1 (meaning you do this by title). *
4. The number of different genres it has (ALL of them including those not in criteria 2).**

Question 1 (20 points)

You are tasked with executing this challenge using **only one** class, the MRJob library, and Python 3.6+.

You need to return all the programs that are a "*Big Data Brother* APPROVED program" in the following format:

```
[title, approved genres], [sum_unique_dates, amount_genres]
```

=== A made up example (not from the data) ===

```
["The Perfume", "'Adventure', 'Animated'"], [116, 5]
```

Explanation: "The Perfume" happened to match all criteria of a "*Big Data Brother* APPROVED program" – it has **5** genres **but only 2 of them satisfied criteria 2**: 'Adventure' and 'Animated'.

Guidelines:

- Make sure you download the '.py' file this MRJob produces and name it according to the guidelines.
- Make sure the following command runs and provides the correct output:
  ! python hw1_mrA1_id1_id2.py 440k_data.csv -q

Question 2 (10 points) – COMPETITVE QUESTION

The "BEST *Big Data Brother* APPROVED program" is the program with the highest *Big Data Brother* score, which is the **sum** of **the number of dates** it was aired in and **the number of different genres** it has ('3'* + '4'** from what you had to return in the previous part).

You need to write a new MRJob by **only adding** code the MRJob you wrote in Question 1 which will return the "BEST *Big Data Brother* APPROVED program".

Note: The code to this question should include **the entire** class you wrote in question 1 with some extra code!

The output should be in the following format:

```
[title, total_score]
```

=== Example ===

```
["The Perfume", 121]
```

You calculate the total score in the following way:

```
total_score = sum_unique_dates + amount_genres
```

Competitive Part:

Returning the right answer gives you 5 points, the rest of the points will be awarded to those with minimal code addition – that is you should aspire to make as little additions to your code as possible.

Guidelines:

- Notice that you are to return only one program title.
- Maybe think how to answer Question 1 to minimize code additions in Question 2.
- Make sure you download the '.py' file this MRJob produces and name it according to the guidelines.
- Make sure the following command runs and provides the correct output: ! python hw1_mrA2_id1_id2.py 440k_data.csv -q
- The output to this question is a sneak peak into your first project assignment ;)

Part B: PySpark (35 points)

Congratulations! You're just one challenge away from the *Big Data Brother* finale — but don't celebrate yet. *Big Data Brother*, all-knowing as always, spotted your Spark skills and has one final task for you in the **same dataset** as Part A.

The *Big Data Brother* has a very specific taste in what he likes to watch, so he developed a scoring mechanism to rate the programs in the dataset we have given you, the scoring mechanism (which is per viewing) is as follows:

1. The *Big Data Brother* hates uncertainty, so any program that has a genres list of only ONE genre gets +10 points (per viewing).
2. The *Big Data Brother* likes programs that has at least one of the genres 'Adventure' or 'Animated', so programs that satisfy this condition get +90 points (per viewing).
3. The *Big Data Brother* like longer programs, so each viewing gets points that amount to +its duration in minutes divided by 5.
4. The *Big Data Brother* automatically awards +100 points to each viewing of a program that has the word 'girls' (case insensitive) in its title.
5. The *Big Data Brother* would NEVER watch a program that **has** an airing during the Distributed Database Management lecture, that is on Thursdays 13:30-15:30, as he prefers to attend all Distributed Database Management Lectures!

   Note: a program can have an air_time that is not in that interval, but due to its duration it does end up landing in that interval, so do take that into account. E.g. An episode of "Euphoria" can air on Thursday at 13:00, but since the episode is 1 hour long, part of the episode lands between 13:30 and 14:00… so the *Big Data Brother* would NEVER watch "Euphoria" – and neither would whoever wrote this question…

6. The *Big Data Brother* would NEVER watch a program that has one of the following words in its title (case insensitive):

```
['friends', 'bang', 'breaking', 'montana', 'doctor', 'fox', 'news']
```

An example for calculating a score of a viewing event

title	prog_code	genre	air_date	air_time	duration
Good Girls Go To Paris	MV000173880000	Comedy	20150211	141000	110

This viewing gets the following score

```
Score = 10 (only one genre) + 100 (contains 'girls' in its title) +
(110/5) (duration divided by 5) = 132
```

To get the total score of a program that the *Big Data Brother* would watch, we **sum up all the scores of its viewings**. We are interested in showing the title and its genres in the output.

For every (title, genres) pair – sum up all the scores over all the viewing. Return the TOP 20 pairs with the highest score. Notice that in the final output each pair should appear only once.

Question 3 (25 points)

You are to print the TOP 20 pairs with the highest scores in a **descending** order in the following format (a table with three columns):

```
[title | genres | total score]
```

Make sure the entirety of the output's content is visible, for that use .show(Truncate=False).

Guidelines:

- You are only permitted to use PySpark, RDDs, and DataFrames (Python 3.6+).
- You are NOT ALLOWED to use Pandas or any other library that isn't PySpark for data processing.
- You may use Pandas only for Data Visualization and Exploration on a small subset of the dataset.
- You are strictly NOT allowed to use User Defined Functions (UDFs), even if suggested by ChatGPT or any other LLM.

Part C: Relational Query Optimization (35 points)

רשויות התחבורה הימית באגן הים התיכון יצרו מערכת המתעדכנת בזמן אמת ומכילה מידע (והיסטוריה) על מיקום כלי שיט בים התיכון ובנמליו. מבנה המידע הוא המבנה הבא:

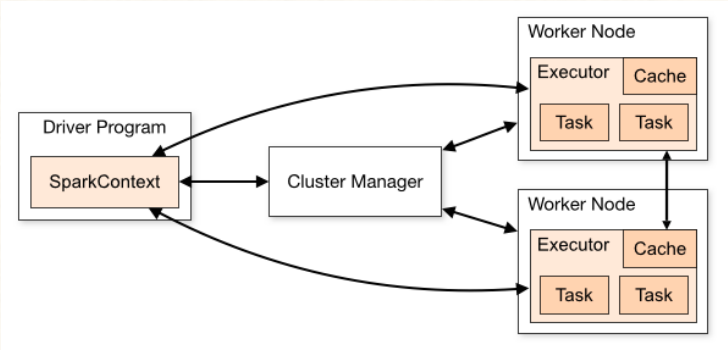
Ship (SHIP_ID, TIMESTAMP, SHIPTYPE, SPEED, LONG, LAT, COURSE, HEADING, DEPARTURE_PORT_NAME, REPORTED_DRAUGHT)

1. SHIP_ID is the anonymized id of the ship (מספר זיהוי ייחודי לכל ספינה)
2. TIMESTAMP is the time at which the message was sent (UTC)
3. SHIPTYPE is defined to be one of: (סוג הספינה)

   1 = Reserved
   2 = Wing In Ground
   3 = Special Category
   4 = High-Speed Craft
   5 = Special Category
   6 = Passenger
   7 = Cargo
   8 = Tanker
   9 = Other

4. SPEED - measured in knots (מהירות הספינה ביחידות קשר)
5. LONG - the longitude of the current ship position (קואורדינטת קו האורך של המיקום הנוכחי)
6. LAT – the latitude of the current ship position (קואורדינטת קו הרוחב של המיקום הנוכחי)
7. COURSE is the direction in which ship moves (כיוון מצפן)
8. HEADING (שונה מכיוון מצפן בגלל זרמים בים) לאן כלי השיט פונה
9. DEPARTURE_PORT_NAME is the name of the last port visited by the vessel (שם הנמל האחרון שהספינה ביקרה בו)
10. REPORTED_DRAUGHT (עומק הספינה מתחת לקו המים)

להלן תיאור סכמטי של ניהול תהליכי עבודה בספארק.[1]



איך זה בא לידי ביטוי?

טְרִיפָה (shuffling) בספארק הוא תהליך המבוצע על ידי מנהל האֶשְׁכּוֹלוֹת (cluster manager) שמטרתו ביזור נתונים על פני אֶשְׁכּוֹלוֹת (clusters), כאשר כל אֶשְׁכּוֹל מנוהל על ידי עובד (Worker Node) ולצורך ביצוע עיבוד מקבילי. הסיבות לטְרִיפָה מגוונות: כאשר הנתונים לא מבוזרים בצורה אחידה, כאשר יש לארגן את הנתונים בצורה מסוימת לעיבוד, או כאשר אין מספיק זיכרון באשכול יחיד לאחסן את כל הנתונים הנדרשים לעיבוד. בזמן פעולת הטְרִיפָה לא ניתן לבצע פעולות נוספות באֶשְׁכּוֹלוֹת המושפעים.

נניח שהרלציה Ship לא עברה התרוססות. נתונה השאילתה הבאה:

לא הפתיעה
לתרוסיס.

Select SHIPTYPE, max(SPEED)

From Ship

Where DEPARTURE_PORT_NAME='Haifa'

Group By SHIPTYPE

ידועים הפרטים הבאים:

1) לצורך עיבוד השאילתה ניתן להקצות חמישה אשכולות, ממספרים 1-5.
2) כל נתוני הרלציה נמצאים באשכול מספר 1.
3) את תוצאות השאילתה יש לשמר באשכול מספר 2.
4) מספר הרשומות בכל סוג כלי שיט הוא לפי הטבלה הבאה, כאשר a הוא ערך כלשהוא.

SHIPTYPE	מספר רשומות
1	a
2	2a
3	3a
4	4a
5	5a
6	4a
7	3a
8	2a
9	a
TOTAL	25a

5) עלות עיבוד נתונים על ידי עובד בכל אחד מהאשכולות תלויה בכמות הרשומות המעובדות. עבור b רשומות העלות היא αb.

6) **ביכולתכם לשלוט מאיפה ולאן תתבצע טריפה!** עלות טריפה תלויה אף היא בכמות הרשומות המעובדות. עבור b רשומות שהועברו מאשכול אחד לאשכול אחר כחלק מהטריפה, העלות היא $\gamma + \beta b$.

7) $\alpha = 1, \beta = 0.9, \gamma = 9$. כמו כן, a>10.

ענו על השאלות הבאות תוך שימוש בנתונים לעיל.

1. (15 נקודות) תוך שימוש בכלים שנלמדו בכיתה (עץ אופרטורים, תכנית שאילתה) הציגו תכנית לשאילתה אשר מבצעת את השאילתה כולה באשכול 1 ומשתמשת בטריפה להעברת תוצאות השאילתה לאשכול 2. חשבו את עלות התכנית.

2. (15 נקודות) תוך שימוש בכלים שנלמדו בכיתה (עץ אופרטורים, תכנית שאילתה) הציגו תכנית לשאילתה אשר ראשית מעבירה בטריפה את כל הנתונים לאשכול 2 ושם מבצעת את השאילתה כולה. חשבו את עלות התכנית.

3. (20 נקודות) נניח כעת שכאשר פעולות מתבצעות במקביל בכמה אשכולות, העלות היא עלות העיבוד המקסימלית באשכול כלשהוא. למשל אם באשכול 3 מעבדים a רשומות ובאשכול 4 מעבדים במקביל 2a רשומות, העלות הכוללת תהיה $\max(\alpha a, \alpha 2a) = \alpha 2a$.

הציעו תכנית לביצוע השאילתה אשר עלותה מינימלית. גבו את הצעתכם בהסבר ובחישוב העלות.

1. (15 נקודות) תוך שימוש בכלים שנלמדו בכיתה (עץ אופרטורים, תכנית שאילתה) הציגו תכנית לשאילתה אשר מבצעת את השאילתה כולה באשכול 1 ומשתמשת בטריפה להעברת תוצאות השאילתה לאשכול 2. חשבו את עלות התכנית.

Select SHIPTYPE, max(SPEED)

From Ship

Where DEPARTURE_PORT_NAME='Haifa'

Group By SHIPTYPE

כמות ל SHIPTYPE =9

SHIPTYPE	מספר רשומות
1	a
2	2a
3	3a
4	4a
5	5a
6	4a
7	3a
8	2a
9	a
TOTAL	25a

on-the-fly -

אפשרי לא נעשים ב on the fly

מבר נבנה לפחות שיש

לתוך המבנה (לזכרון מאגר

מעבר בתוך שאילתא).

טְרִיפָה (shuffling) בספארק הוא תהליך המבוצע על ידי מנהל האֶשְׁכּוֹלוֹת (cluster manager) שמטרתו ביזור נתונים על פני אֶשְׁכּוֹלוֹת (clusters), כאשר כל אֶשְׁכּוֹל מנוהל על ידי עובד (Worker Node) ולצורך ביצוע עיבוד מקבילי.

Shuffling

↑

Group by shipType

↑

DEPARTURE_PORT_NAME='Haifa', max(speed)

↑

SHIP_TYPE, speed

25a

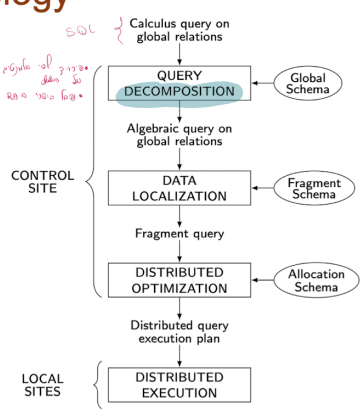
SHIPTYPE	speed

5) עלות עיבוד נתונים על ידי עובד בכל אחד מהאשכולות תלויה בכמות הרשומות המעובדות. עבור b רשומות העלות היא $\rightarrow ab$.

6) **ביכולתכם לשלוט מאיפה ולאן תתבצע טריפה!** עלות טריפה תלויה אף היא בכמות הרשומות המעובדות. עבור b רשומות שהועברו מאשכול אחד לאשכול אחר כחלק מהטריפה, העלות היא $\gamma + \beta b$.

7) $\alpha = 1,\ \beta = 0.9,\ \gamma = 9$. כמו כן, a>10.

Distributed Query Processing Methodology



Shuffle

$25a + \overbrace{9 + 0.9 \cdot 9} = 25a + 17.1$

שאלה:

1. אם נעשה בשאילתא projection

אז כמות הרשומות לא ירדה.

2.

9

SHIPTYPE	מספר רשומות
1	a
2	2a
3	3a
4	4a
5	5a
6	4a
7	3a
8	2a
9	a
TOTAL	25a

5) עלות עיבוד נתונים על ידי שובר בכל אחד מהאשכולות תלויה בכמות הרשומות המעובדות. עבור b רשומות העלות היא ab.

6) **ביכולתכם לשלוט מאיפה ולאן התבצע טריפה!** עלות טריפה תלויה אף היא בכמות הרשומות המעובדות. עבור b רשומות שהועברו מאשכול אחד לאשכול אחר כחלק מהטריפה, העלות היא $\gamma + \beta b$.

7) $\alpha = 1, \beta = 0.9, \gamma = 9$, כמו כן, a>10.

2. (15 נקודות) תוך שימוש בכלים שנלמדו בכיתה (עץ אופרטורים, תכנית שאילתה) הציגו תכנית לשאילתה אשר ראשית מעבירה בטריפה את כל הנתונים לאשכול 2 ושם מבצעת את השאילתה כולה. חשבו את עלות התכנית.

$\pi_{shipType, speed}$

↑

Group by

↑

$\sigma_{port = "Haifa"}$

↑

Shuffling

shuffling

$9 + 0.9 \cdot 25a + 25a = 9 + 47.5a$

3. (20 נקודות) נניח כעת שכאשר פעולות מתבצעות במקביל בכמה אשכולות, העלות היא עלות העיבוד המקסימלית באשכול כלשהוא. למשל אם באשכול 3 מעבדים a רשומות ובאשכול 4 מעבדים במקביל 2a רשומות, העלות הכוללת תהיה $\max(\alpha a, \alpha 2a) = \alpha 2a$.

הציעו תכנית לביצוע השאילתה אשר עלותה מינימלית. גבו את הצעתכם בהסבר ובחישוב העלות.

1	2	3	4	5
סוג ספינה 1 סוג ספינה 4	סוג ספינה 2 סוג ספינה 3	סוג ספינה 5	סוג ספינה 6 סוג ספינה 9	סוג ספינה 7 סוג ספינה 8

מכיוון שנשתמש ב $\max(\alpha a, \alpha 2a) = 5a$ ומכיוון שבמהלך כל אשכול בזמן שווה לזמן 5a

סכום נכסל: 5a

1. נעשה shuffle של 25a מאשכול 1

$9 + 0.9 \cdot 20a = 9 + 18a$

2. מבצעים את הקיבוץ (selection מיון בכל אשכול)

5a

3. מעבר של כל הנתונים לאשכול 2.

$9 + 0.9 \cdot 7 = 15.3$

יש לנו 9 נתונים
לעומת 1-2 אבל כבר נמצאים
באשכול 2.

לכן סכום נכסל: $23a + 24.3$, כאילו כל הבעיות של שאילתא מס. היו הכי גרוע

1 ו-2 ועוד בנוסף כי מס. 5a (והמספר הגדל של הנתונים לפעמים עדיף 5a, לכן החלוקה הזו. הא אידיאלי)

Tips and more:

- We advise you to work with **Google Colab** as it provides you with an easy interface, shared workspace with your teammates, and your own local machine.

- **Study your data** so you can write your code accordingly, save time, and avoid unnecessary bugs along the way.

- **Be organized!** An organized, well-structured, and clear solution that shows your thought process will win our hearts (be graded more favorably). A messy one… not so much.

- You may use ChatGPT or other LLMs, but you are strongly encouraged to understand the logic yourself, as future assignments may not be solvable using LLMs alone.

- We gave you two csv files: a small one (50k rows) and a large one (440k rows). To make it easier for you to develop the code, we prepared the '50k_data.csv' file for you to run your code on for faster processing time. However, you need to submit the output based on the '440k_data.csv' file!

- We also provided a file called '50k_outputs.txt containing the correct outputs on the '50k_data.csv' file for you to check yourself.

GOOD LUCK

COURSE STAFF AND THE BIG DATA BROTHER