DAT470 Assignment 3

Olof Lindberg
Rikard Roos

March 2025

Problem 1

a)

1.MAP – Input: lines from planets.csv

```
For each line in lines:
    emit (constellation + star, RU)
```

2. REDUCE – Input: (constellation + star, List(RU))

```
For each constellation + star:
    emit (constellation + star, RU_sum)
```

b)

The data flow of problem 1 can be seen in figure 1.

c)

Star	Ru
Capella	1248
Aplha Cancri	2082
Gamma Sagittae	2899
Beta Lyrae	2431
Alpha Geminorum	4245

Problem 2

a)

The data flow of problem 2 can be seen in figure 2.

b)

Star	RU
Beta Scorpii	12680
Delta Tauri	12080
Alpha Ceti	11739
Alpha Centauri	11476
Delta Aurigae	11005
Beta Cephei	10447
Zeeman	10421
Zeta Vulpeculae	9404
Beta Normae	9153
Delta Brahe	9100

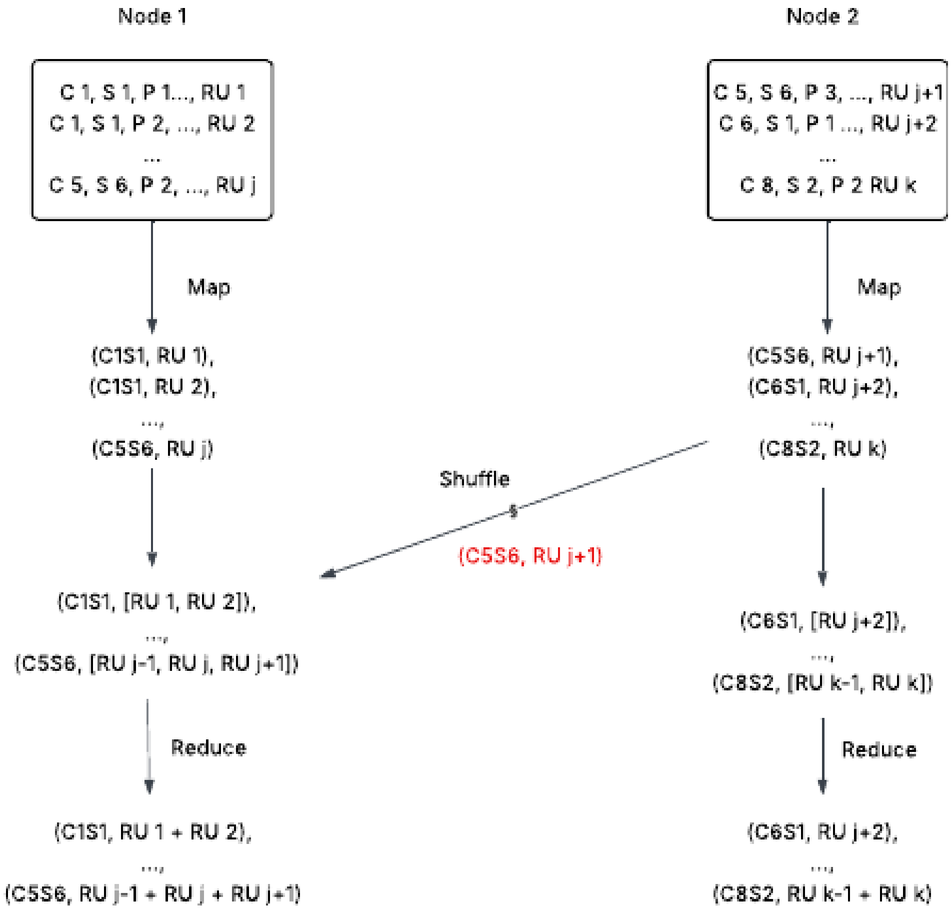


Figure 1: A diagram of the data flow for problem 1.

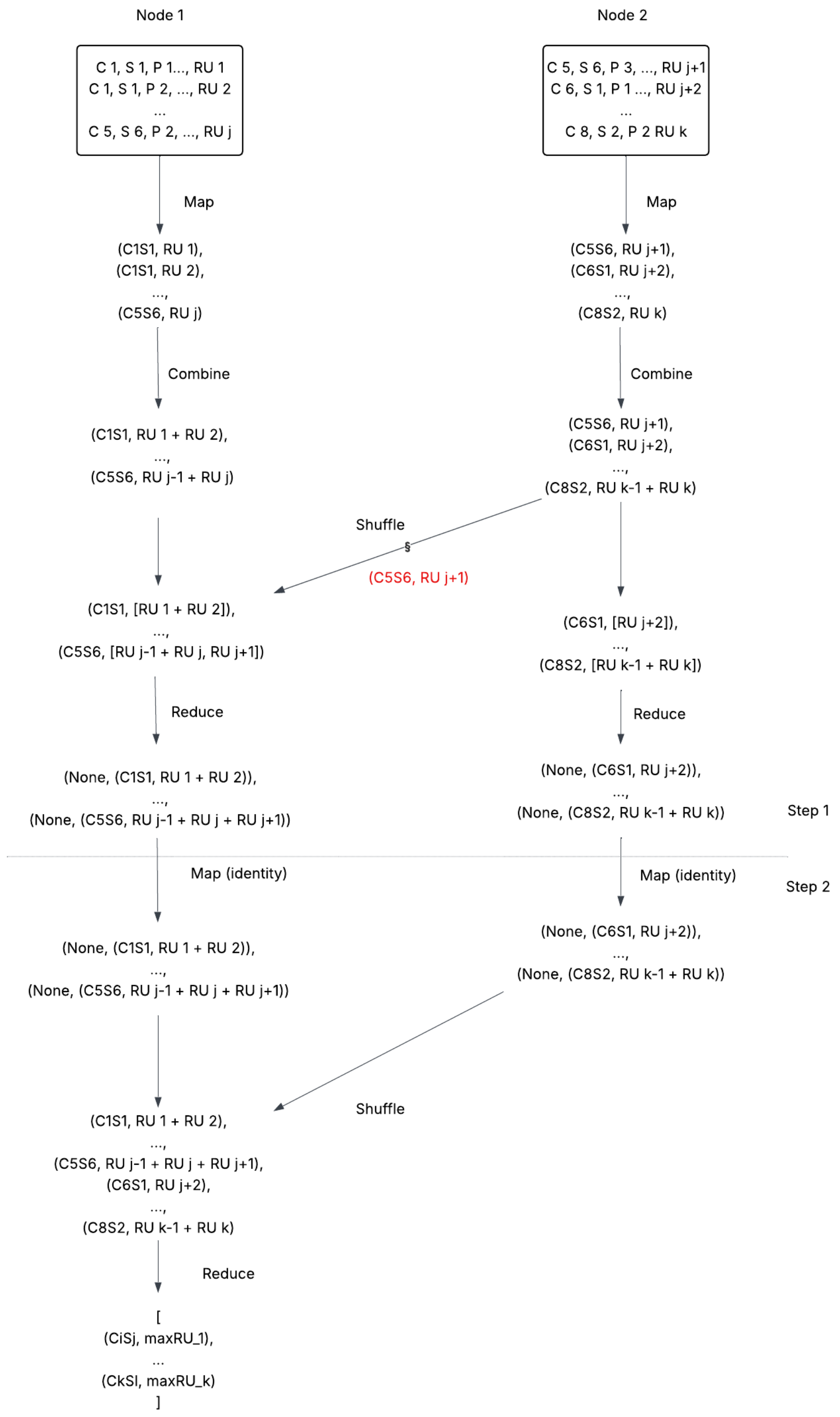


Figure 2: A diagram of the data flow for problem 2.

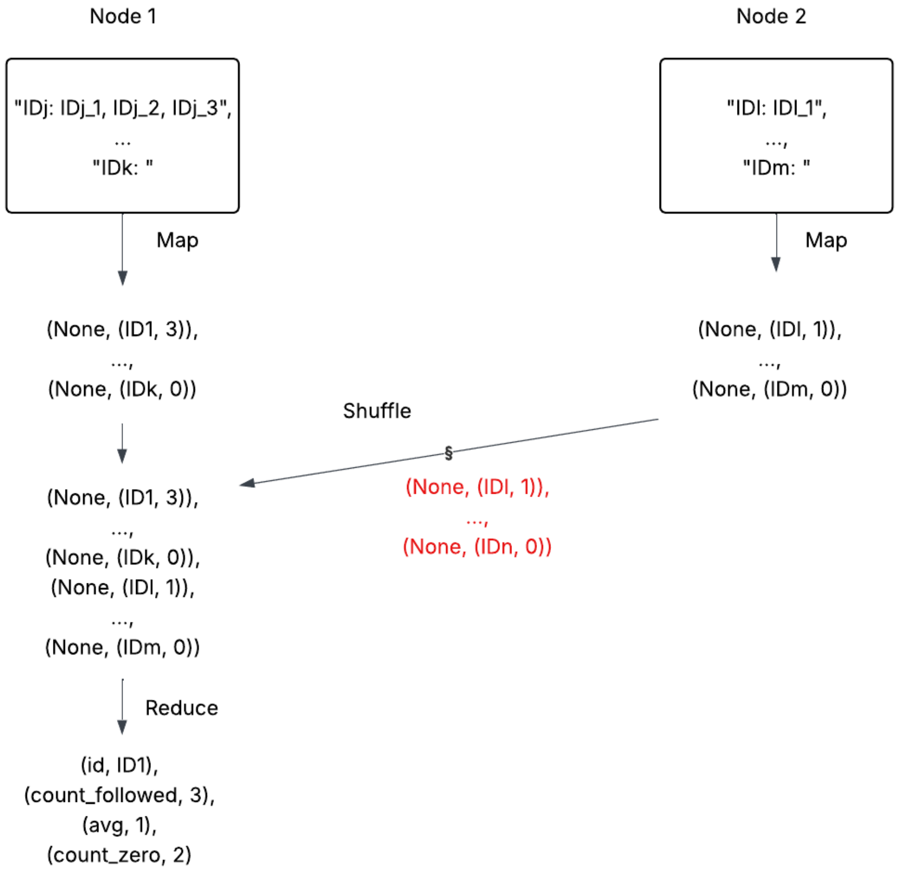


Figure 3: A diagram of the data flow for problem 3.

Problem 3

a)

1.MAP – Input: lines from twitter-2010_full.txt

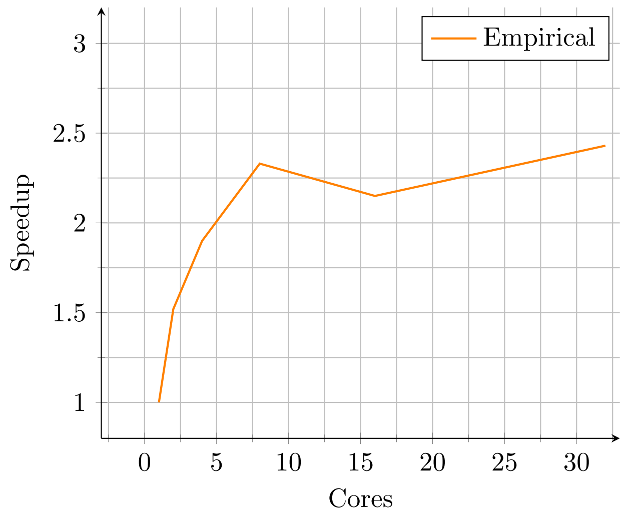
```
For each line in lines:
    emit (dummy_key, (id, nr_accounts_followed))
    // All data will be sent to one node
```

2. REDUCE – Input: (dummy_key,(id, nr_accounts_followed))

```
sort the data pairs in descending order after nr_accounts_followed.
save the first element to most_followed.
count average followed accounts per user.
count users with 0 accounts_followed.
emit the user id of most_followed.
emit the followed_count of most followed.
emit the average followed account per user.
emit the count of users with 0 accounts_followed.
```

The data flow of problem 3 can be seen in figure 3.

c)



runtime with one core: 67.31 seconds

d)

```
most followed id 813286
most followed 770155
average followers 35.25297882010159
count follows no-one 5963082
```

Problem 4

a)

1.MAP – Input: lines from twitter-2010 full.txt

```
For each line in lines:
    For each follower in accounts_followed:
        emit(follower, 1)
    emit (id, 0)
```

2. COMBINE – Input: (id, ones)

```
emit (id, sum(ones))
```

3. REDUCE – Input: (id, sums)

```
emit (dummy_key, (id, sum(sums)))
```

4. REDUCE2 – Input: ((dummy_key, (id, followers_counts))

```
sort the data pairs in descending order after followers_count.
save the first element to most_followers.
count average followers per user.
count users with 0 followers.
emit the user id of most_followers.
emit the followers_count of most_followers.
emit the average followers per user.
emit the count of users with 0 followers.
```

The data flow of problem 4 can be seen in figure 4.

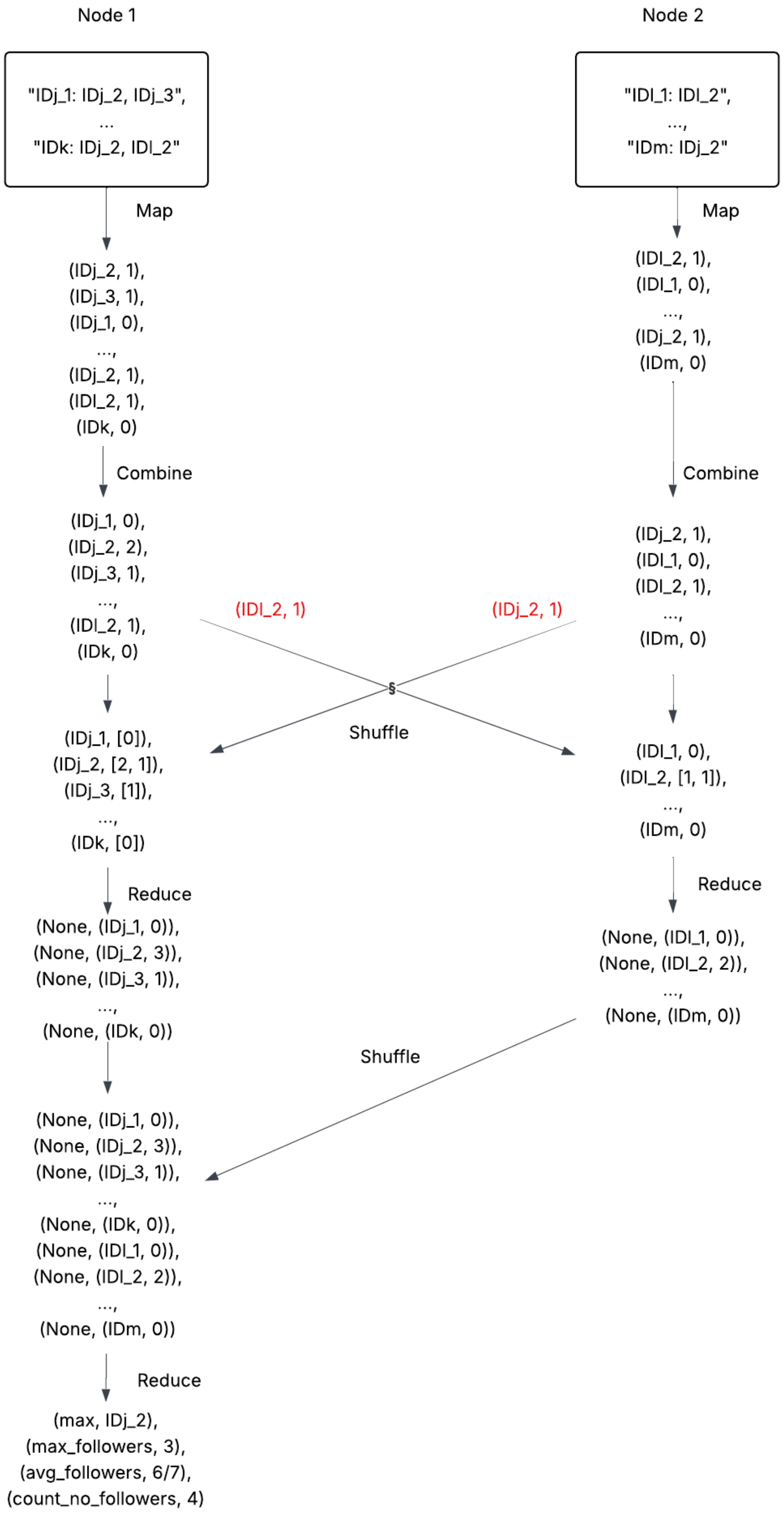
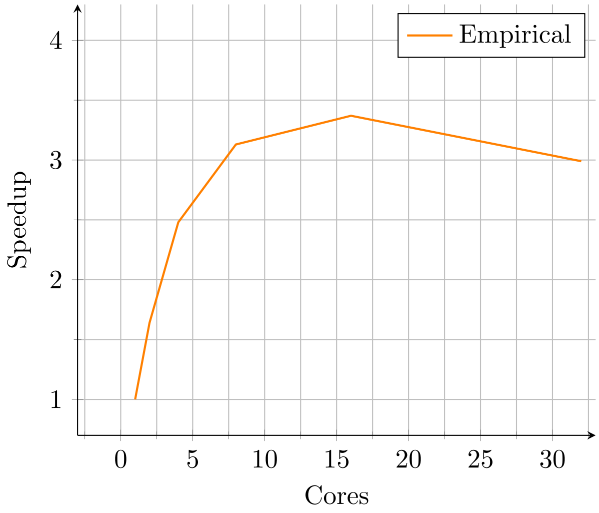


Figure 4: A diagram of the data flow for problem 4.

c)



Number of workers: 1

Time elapsed: 575.6433525085449 s

d)

```
most followers id 19058681
most followers 2997469
average followers 35.25297882010159
count no followers 1548949
```