Azat Dovgeldiyev

CSC 555

Homework 1

Problem 1.

a.

```
SELECT First, AVG(Grade)
FROM Student
GROUP BY First;
```

The mapper produces key value consisting of First as the key and values are the grades. Reducer will perform an average in all grades of each key.

b.

```
SELECT City, State, COUNT(Name)
FROM Student
GROUP BY City, State;
```

For this block, City, State are the keys and Name is the Value. The reducer will perform a count of all the occurrences of each key. The mapper assigns a value 1 for each key and the reducer sum all of the records for each key.

c.

```
SELECT a.First, a.Last, e.EID, a.AID, e.Phone
FROM Employee as e, Agent as a
WHERE e.Last = a.Last AND e.First = a.First;
```

The mapper will produce keys containing First and Last name for employee block, and agent block. The values are EID, Phone employee, and AID for Agent keys. The reducer will combine values where the keys from Employee and match the keys from Agent.

Problem 2.

a. How would you attempt to speed up the repeated execution of the query? (2-a is intentionally an open-ended question, there are several acceptable answers)

If we make the size of reducer smaller, there will be more reducer work in the given time.

b. If a Mapper task fails while processing a block of data – what is the location (which node) where MapReduce framework will prefer to restart it?

The Master or Name node will schedule a Worker when one becomes available to work if a Mapper task fails while processing a block of data.

c. If the job is executed with 4 Reducers

i. How many files does the output generate?
Each reducer produces 1 output file, so 4 files will be generated.

ii. Suggest one possible hash function that may be used to assign keys to reducers.
H(x) = x MOD 4 is one of the most common and widely used hash function in mapper function to generate keys to send reducers.
Time(minutes/hours) MOD 4

Problem 3.

a) You need to choose your own (small) prime numbers, compute ø(n), choose one key and compute the other key.
Let p=3 and q=11 two
$n = p * q \rightarrow 3 * 11 = 33$ next we compute phi(n),
$phi(n) = (p - 1) * (q - 1) \rightarrow 2 * 10 = 20$, next random e=3
$e * d = 1\ mod\ phi(n)\ and\ 0 \leq d \leq n$, and we can find ‘d’ with extended Euclidean algorithm,
$3 * 7\ mod\ 20 = 1$
d=7
public encryption key:KU={33,3}
private decryption key:KR={33,7}

b) Verify that a numeric message can be send by computing the cyphertext and decrypting it with the other key.
To encrypt message $m = 4, c = m^e\ mod\ n, c = 4^3\ mod\ 33 \rightarrow 64\ mod\ 33 = 31$
To decrypt the ciphertext C, $m = c^d\ mod\ n, 31^7\ mod\ 33 = 27512614111\ mod\ 33 = 4$

Problem 4.

a) One Hadoop per node then 54 * 1=54 minutes to process the file.

b) 20 Hadoop worker nodes?
Without any failure 54/20 = 2.7, therefore, it will take 3 minutes

c) 50 Hadoop worker nodes?
54/50=1.08, it will take 2 minutes

d) 100 Hadoop worker nodes?
1 minutes

e) Now suppose you were told that the replication factor has been changed to 3? That is, each block is stored in triplicate, but file size is still 38 blocks. Which of the answers (if any) in a)-c) above will have to change?
Without any failure in terms of mapping process the working time for every node is the same, we change replication factor to 3 to protect when node fails.

Problem 5.

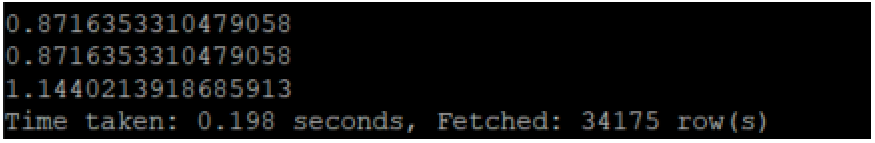
a)
(Copy the query output and report how many rows you got as an answer.)

1 row

```
hive> SELECT COUNT(*) FROM VehicleData;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = ec2-user_20201003014557_122be78f-d900-45c3-8334-8950a3f7ca82
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1601689147366_0001, Tracking URL = http://ip-172-31-48-241.ec2.internal:8088/proxy/application_1601689147366_0001/
Kill Command = /home/ec2-user/hadoop-2.6.4/bin/hadoop job  -kill job_1601689147366_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-10-03 01:46:12,066 Stage-1 map = 0%,  reduce = 0%
2020-10-03 01:46:20,104 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.26 sec
2020-10-03 01:46:28,917 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 2.58 sec
MapReduce Total cumulative CPU time: 2 seconds 580 msec
Ended Job = job_1601689147366_0001
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 2.58 sec   HDFS Read: 11775010 HDFS Write: 6 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 580 msec
OK
34175
Time taken: 32.151 seconds, Fetched: 1 row(s)
```

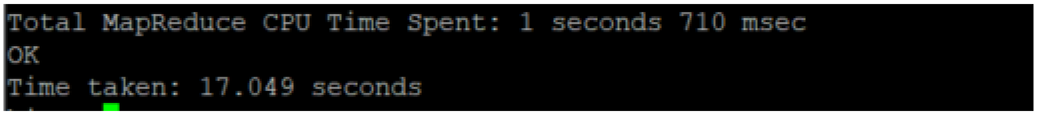
```
hive> SELECT MIN(barrels08), AVG(barrels08), MAX(barrels08) FROM VehicleData;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = ec2-user_20201003014953_1125c4e6-5a7e-4aae-89fc-56fd63ad6a8d
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1601689147366_0002, Tracking URL = http://ip-172-31-48-241.ec2.internal:8088/proxy/application_1601689147366_0002/
Kill Command = /home/ec2-user/hadoop-2.6.4/bin/hadoop job  -kill job_1601689147366_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-10-03 01:50:02,288 Stage-1 map = 0%,  reduce = 0%
2020-10-03 01:50:09,974 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.58 sec
2020-10-03 01:50:17,680 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 3.03 sec
MapReduce Total cumulative CPU time: 3 seconds 30 msec
Ended Job = job_1601689147366_0002
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 3.03 sec   HDFS Read: 11777415 HDFS Write: 37 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 30 msec
OK
0.059892        17.820177449476272      47.06831
Time taken: 25.925 seconds, Fetched: 1 row(s)
```

SELECT (barrels08/city08) FROM VehicleData;



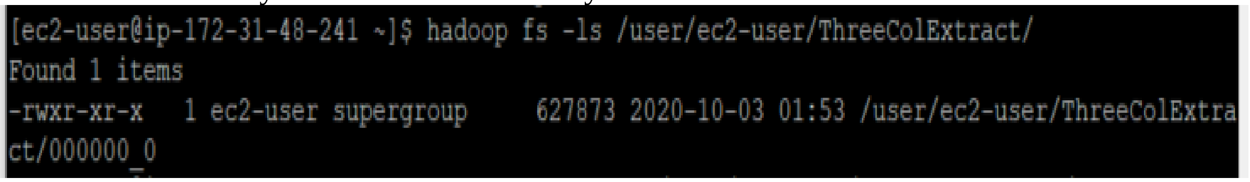
```
0.8716353310479058
0.8716353310479058
1.1440213918685913
Time taken: 0.198 seconds, Fetched: 34175 row(s)
```

INSERT OVERWRITE DIRECTORY 'ThreeColExtract'
SELECT barrels08, city08, charge120
FROM VehicleData;



```
Total MapReduce CPU Time Spent: 1 seconds 710 msec
OK
Time taken: 17.049 seconds
```

The size of the newly created file is 627873 bytes.



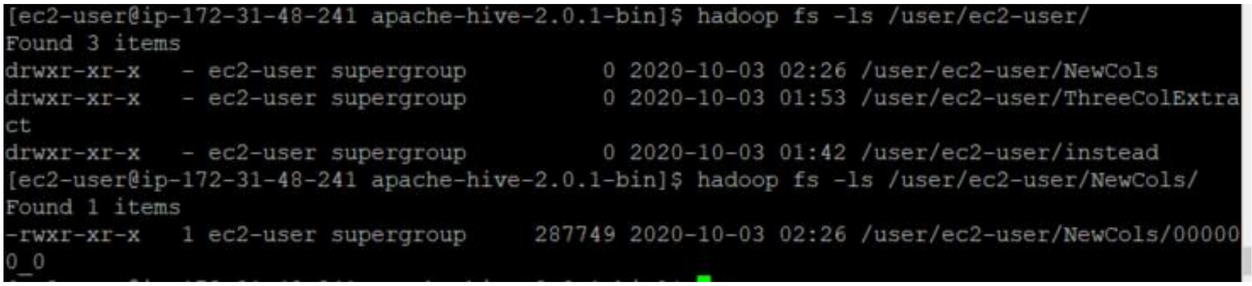
```
[ec2-user@ip-172-31-48-241 ~]$ hadoop fs -ls /user/ec2-user/ThreeColExtract/
Found 1 items
-rwxr-xr-x   1 ec2-user supergroup     627873 2020-10-03 01:53 /user/ec2-user/ThreeColExtra
ct/000000_0
```

```
hive> CREATE TABLE NewVehicleData (
    > barrels08 FLOAT, barrelsA08 FLOAT,
    > charge120 FLOAT, charge240 FLOAT,
    > city08 FLOAT, city08U FLOAT,
    > cityA08 FLOAT, cityA08U FLOAT)
    > ROW FORMAT DELIMITED FIELDS
    > TERMINATED BY ',' STORED AS TEXTFILE;
OK
Time taken: 2.055 seconds
```

```
hive> INSERT OVERWRITE DIRECTORY 'NewCols'
    > SELECT city08U, cityA08U
    > FROM NewVehicleData;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future
versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X
releases.
Query ID = ec2-user_20201003022549_2fdb0d9b-eb35-4193-bf5f-2a8b6d525eca
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1601689147366_0004, Tracking URL = http://ip-172-31-48-
241.ec2.internal:8088/proxy/application_1601689147366_0004/
Kill Command = /home/ec2-user/hadoop-2.6.4/bin/hadoop job  -kill
job_1601689147366_0004
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2020-10-03 02:26:00,693 Stage-1 map = 0%,  reduce = 0%
2020-10-03 02:26:08,797 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.72 sec
MapReduce Total cumulative CPU time: 1 seconds 720 msec
Ended Job = job_1601689147366_0004
Stage-3 is selected by condition resolver.
Stage-2 is filtered out by condition resolver.
Stage-4 is filtered out by condition resolver.
Moving data to: hdfs://localhost/user/ec2-user/NewCols/.hive-staging_hive_2020-10-
03_02-25-49_754_6765530815332055603-1/-ext-10000
Moving data to: NewCols
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 1.72 sec   HDFS Read: 11770728 HDFS Write:
287749 SUCCESS
```

Total MapReduce CPU Time Spent: 1 seconds 720 msec
OK
Time taken: 20.269 seconds

The size of newly created file is 287749 bytes.



```
[ec2-user@ip-172-31-48-241 apache-hive-2.0.1-bin]$ hadoop fs -ls /user/ec2-user/
Found 3 items
drwxr-xr-x   - ec2-user supergroup          0 2020-10-03 02:26 /user/ec2-user/NewCols
drwxr-xr-x   - ec2-user supergroup          0 2020-10-03 01:53 /user/ec2-user/ThreeColExtra
ct
drwxr-xr-x   - ec2-user supergroup          0 2020-10-03 01:42 /user/ec2-user/instead
[ec2-user@ip-172-31-48-241 apache-hive-2.0.1-bin]$ hadoop fs -ls /user/ec2-user/NewCols/
Found 1 items
-rwxr-xr-x   1 ec2-user supergroup     287749 2020-10-03 02:26 /user/ec2-user/NewCols/00000
0_0
```