Qtext :

a) Illustrate that the following statement is true : "We should make the sequential bottleneck (i.e. portion of code that cannot be parallelized) as small as possible. Increasing the cluster size alone may not result in a good speedup" [02Marks]

b) A hierarchical memory system that uses cache memory has cache access time of 50 nanoseconds and main memory access time of 300 nanoseconds. 75% of memory accesses are for read with cache hit-ratio of 0.8 for read access. The writes go directly to the main memory. What will be the average access time of the system specifically for read and write requests ? [03 Marks]

a) The data analytics team at an e-commerce company is setting up Hive for storing historical data. The data contains sales transactions with following details : product_category, purchase_month_year, productID, customerID, and amount. A sample record is ["Accessories", 05-2021, PID12321, CUST2341, 12000].

i) Explain a possible partitioning and bucketing strategy along with the SQL Query for CREATE TABLE.

ii) If the most common type of query is to find sales trends for a specific productID, what optimization can we do to quickly find the productID ? Update the CREATE TABLE statement in (a) to include the optimization. [04 Marks]

b) What are CRUD operations and how are they performed in MongoDB Discuss with example? [02 Marks]

Qtext :

a) Design an application using different services of AWS, the application should perform search of a very large set of records, attempting to identify records that satisfy a regular expression. The source of this search is a collection of document URLs produced by the Alexa Web Search, a software system that crawls the Web every night. The inputs to the applications are a regular expression and the large data set produced by the Web-crawling software; the output is the set of records that satisfy the expression. The user should be able to interact with the application and get the current status.The application should use message passing to trigger the activities and it should use Hadoop MapReduce for computation. [06 Marks]

b) What are the different cloud delivery and deployment models you will use in the following scenarios? [03 Marks]

i)A Financial organization wants to move the legacy code and data to Cloud, they are however concerned about data security in certain applications.

ii) An organization maintaining movie databases wants to move the application and data to cloud. They don't want to maintain any infrastructure management team.

iii) A startup is not sure if it will be successful, and they don't have initial capital for inhouse systems. They want to move their HR and Customer management from spreadsheet to industry standard processes without overhead of a big IT team.

Qtext :

a) Write RDD transformations for following: [03 Marks]

i) Your Base RDD contains lines from a text. Create another RDD containing the lines and its length

ii) Your Base RDD contains lines from a text. Convert lines into words and then count the word “RDD”

iii) Create a base RDD with dates (2,”Oct”,1975), (6,”Jan”, 2004), (2,”Oct”,1975) and create another RDD with unique dates

b) In the following scenarios, choose between the given types of storage services on AWS Cloud: Simple Storage Service (S3), Elastic Block Storage (EBS) or Elastic File System (EFS). In each case mention which one will you use and briefly explain why.

i) You need to setup a Big Data DB and query service on Cloud that is very I/O intensive and needs millisecond level latency on queries.

ii) A digital streaming media company stores interaction data from millions of users to be consumed by its personalized recommendation engine.

iii)An application running on multiple VM instances on Cloud needs to store data into a common repository that each instance can mount using NFS protocol to read or write data. [03 Marks]

Qtext :

a) Consider the following code snippet in Spark showing a set of operations:

Line 1: data = sc.parallelize([1,2,5,3,4,4,5,6,7])

Line 2: new_data = data.map(lambda x: (x, 2*x+5))

Line 3: filtered1 = new_data.filter(lambda x: x[1] > 15)

Line 4: filtered2 = new_data.filter(lambda x: x[1] <= 15)

Line 5: result = filtered2.reduceByKey(lambda a,b: a+b)

Line 6: sorted = result.sortByKey()

Line 7: print(sorted.collect())

Explain the concepts of Narrow, Wide transformations, Actions, and Lazy evaluation in the context of this code. What is the final output of the code? [05 Marks]

b) Write a query to insert a new column (new_col INT) into a hive table (h_table) at a position before an existing column (x_col). [02 Marks]

Qtext :

a) Write a pseudocode for spark program to check if a given keyword exists in a huge text file or not? [03Marks]

b) Write the pseudocode of Map and Reduce functions in Hadoop MapReduce philosophy to find average daily gain of each company from 01/01/2020 – 12/31/2021. The company data is a set of lines: {date, company, start_price, end_price } [04 Marks]

BDS-MIDSEM-Regular

1. JetFlix is the online movie streaming service. It maintains the database of the movies viewed by its users in the following format, where "1" indicates user has seen the movie. [4 + 2 + 1 = 07 Marks]

User Movies	M1	M2	M3	M4	M5
U1	1	1	0	0	1
U2	0	0	1	1	1
U3	1	0	1	1	1
U4	0	0	0	1	1
U5	1	1	1	0	0

1.

(A) Write pseudo map reduce functions code that will help to determine the most popular movie among the user

ANSWER KEY:

```
<key, value> map(user_id, user_record)
{
    For each movie in user_record
        Emit (movide_id, 1)
}
    Reduce (List [<movie_id, [list of counts of movie_id]>])
    {
        total_views = 0
        popular_movie_id = ""

        For each movie_id
            Count =  0
            For each value in list of counts of movie_id
            Count = count + value
            If Count >= total_views
                Total_views = Count
                popular_movie_id = movie_id

        emit(popular_movie_id, total_views)
}
```

02 marks for map function
02 marks for reduce function

(B) Show the outcome of the map phase for Users U1 and U5.

ANSWER KEY:

Outcome of Map phase

For User1 : {M1: 1}, {M2:1}, {M3:0}, {M4:0}, {M5:1}
For User5 : {M1: 1}, {M2:1}, {M3:1}, {M4:0}, {M5:0}
mark for each correct output

(C) Show the data structure that will be generated after shuffle is carried out.

ANSWER KEY:

Data structure that will be generated after shuffle is carried out
M1 : [1, 0, 1, 0, 1]
M2 : [1, 0, 0, 0, 1]
M3 : [0, 1, 1, 0, 1]
M4 : [0, 1, 1, 1, 0]
M5 : [1, 1, 1, 1, 0]
1 mark for the structure

2. Consider a bigdata analytics project for "*Building recommendation system by analyzing the user ratings and feedback*" illustrate different phases in lifecycle of this bigdata project. (06Marks)

ANSWER KEY:

Phases in Bigdata Analytics :

Business Case Evaluation; Data Identification; Data Acquisition & Filtering; Data Extraction; Data Validation & Cleansing; Data Aggregation & Representation; Data Analysis; Data Visualization; Utilization of Analysis Results

Student need to map given problem in to different phases

Explanation of each phase for the given problem statement : 01marks * atleast 06 phases

3. Answer the following Questions: (04+04+04 = 12Marks)

I. Ravi wants to run a simulation for his research and his supervisor advised him to run it for a fixed problem size. Ravi is successful in achieving 90% parallelism of the code, with 10% of it being sequential. What speedup can Ravi expect on 10 processors? What would be the maximum speedup on an infinite number of processors?

ANSWER KEY:

This is solved using Amdahl's Law.

$Sp = 1 / (f + (1 - f)/p) = 1 / (.1 + (.9)/10) = 1 / .19 = 5.26$

$\max Sp = 1 / (.1 + .9/\infty) = 1 / .1 = 10$

Calculating speedup 02 Marks

Calculating max speedup 02 Marks

II. In a Movie reservation system what are the trade-offs between consistency and availability you can think of for the following cases (applying CAP theorem)? Justify your answer:

(A) When most of seats are available

(B) When the cinema hall is close to be filled

ANSWER KEY:

A) It is ok to rely on somewhat out-of-date data, availability is more critical - 02 Marks

B) It needs more accurate data to ensure the cinema hall is not overbooked, consistency is more critical - 02 Marks

III. In each of the following cases, identify the motivation for high performance requirement:
(A) Airline scheduling
(B) Summary Statistics of Historic Sales Data of a Retailer
(C) Web Crawler
(D) Stockmarket analysis

ANSWER KEY:

Discussion of the Given cases are HPC problems.

Students need to justify why the given cases are HPC problems.

Discussion w.r.t their resource requirements.

04 cases * 01 mark for each case

4. Answer the following questions: (02 + 03 = 05 Marks)

I. *"For an organization if the latency is more critical than data"*, in this case weather to implement ACID or BASE data model? Justify your answer.

ANSWER KEY:

BASE would be the best option, otherwise, the ACID will help you to make the system as reliable as possible.

II. Justify the statement *"Hadoop Distributed File System (HDFS) provides fault tolerant service"*.

ANSWER KEY:

HDFS provide FT using replication default replication factor is 3. The blocks in the data nodes are replicated by a factor of 3 where one replica is stored in the same data node where the original file resides.

The second replica is stored in another node within the same cluster. The third replica is stored in another node within the same cluster or in another cluster if it is available.

Birla Institute of Technology and Science, Pilani

Work Integrated Learning Programmes Division Cluster
Programme - M.Tech in Data Science and Engg.
I Semester 2020-21 (Regular- ANSWERKEY)

Course Number: DSECL ZG522
Course Name: Big Data Systems
Exam. Mode: Open Book
Weightage: 30%
Duration: ~~90 Minutes~~
Date: ~~29/03/2020 2 – 3.30PM~~

Pages: 2
Questions: 4
Marks: 30

Instructions

1. Questions are to be answered strictly in the order in which they appear in the question paper and in the page numbers specified in the answer-book.
2. **All answers must be directed to the question in short and simple paragraphs or bullet points; use visuals/diagrams wherever necessary; use last sheets of your answer-book for thinking and rough-work if necessary, and write legibly the final answer on the designated pages**.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

Q1. Consider a function of the form for answering the sub-questions: **[2 + 2 + 2 + 4 = 10]**

```
Boolean matchWord(F, word) {
    open text file F for reading;
    test whether the given word occurs in F;
    if word is present
        return TRUE;
    otherwise
        return FALSE;
}
```

Assume that a large collection C of text files is given.

a) How would you verify whether there is at least one file in C that contains the given word using the approaches discussed in the class?

b) Assume that C has N documents and there is no limit on the number of computers one can use. How many computers are needed to obtain the maximum parallelism in testing whether there is at least one file that contains the word using the map-reduce model?

c) Assume that C has N documents and there is no limit on the number of computers one can use. Ignore the time taken to distribute the files in C to all computers and the time taken to communicate the results of the computation in this question. How much time will be taken for testing whether there is at least one file in C that contains the word using the map-reduce model?

d) Again consider a collection C of N text files stored – one each – in N computers in a network. Assume that the function extractWords(F) takes a file F and returns a list of words that occur in F – each word is listed only once in the output even if it occurs more than once in F. Simply explain whether it will be possible with the Map-Reduce approach or not? Why or Why not?

Answers:

a) How would you verify whether there is at least one file in C that contains the given word using the approaches discussed in the class?

Answer:

Applying the function “matchWord” parallely on C using the map construct

Then performing logical OR using the parallel reverse tree algorithm on all map outputs.
Or using the reducer that will take the input as array of the map results and find out whether any true appearing in it or not.

b) Assume that C has N documents and there is no limit on the number of computers one can use. How many computers are needed to obtain the maximum parallelism in testing whether there is at least one file that contains the word using the map-reduce model?
Answer:
N computers for the map phase and N/2 computers for the reduce phase
Map can process each of the N files in parallel – one in each computer if it has N computers; reduce will combine two Boolean values in one operation i.e. it can execute N/2 OR operations in one step, and then N/4 operations, and so on

c) Assume that C has N documents and there is no limit on the number of computers one can use. Ignore the time taken to distribute the files in C to all computers and the time taken to communicate the results of the computation in this question.How much time will be taken for testing whether there is at least one file in C that contains the word using the map-reduce model?
Answer:
Time taken for executing matchWord once + Time taken for executing OR operation log2 N times
Map executes N matchWord operations in parallel. So the time taken for all matchWord operations is the same as executing one of them. Reduce executes N/2 OR operations in the first step, N/4 operations in the second step and so on until it performs one OR operation to get a single Boolean result – this take logN steps.

d) Again consider a collection C of N text files stored – one each – in N computers in a network. Assume that the function extractWords(F) takes a file F and returns a list of words that occur in F – each word is listed only once in the output even if it occurs more than once in F. Simply explain whether it will be possible with the Map-Reduce approach or not? Why or Why not?
Answer:
Yes it's possible with Map-Reduce approach.
1) (map_extractWords(C)) will apply the function extractWords() parallelly on each individual doc. The execution of extractWords () on a single doc will return a list of words.
2) So, the overall output generated after the successful execution of extractWords () on each document is a list of list of words, say list1.
3) (reduce_ extractWords (list1)) will return the intersection of all lists in list1which will be a list of words that occur in all files in C.

Q2. Answer the following in brief: **[2*3 = 6 marks]**

a) "For an organization if the latency is more critical than data", in this case weather to implement ACID or BASE data model? Justify your answer.

Answer: BASE would be the best option, otherwise, the ACID will help you to make the system as reliable as possible.

b) In a Movie reservation system what are the trade-offs between consistency and availability you can think of for the following cases (applying CAP theorem)? Justify your answer: (06 Marks)
i) When most of seats are available
ii) When the cinema hall is close to be filled
Answer**:**
i) It is ok to rely on somewhat out-of-date data, availability is more critical

ii) It needs more accurate data to ensure the cinema hall is not overbooked, consistency is more critical

c) Justify the statement "Hadoop Distributed File System (HDFS) provides fault tolerant service".

Answer**:**
HDFS provide FT using replication default replication factor is 3. The blocks in the data nodes are replicated by a factor of 3 where one replica is stored in the same data node where the original file resides. The second replica is stored in another node within the same cluster. The third replica is stored in another node within the same cluster or in another cluster if it is available.

Q3. Consider a data storage for University students. Each student data, stuData, which is in a file of size less than 64 MB (1 MB = 2^{20}B). A data block (of default size 64MB) stores the full file data for a student of stuData_idN, where N =1 to 500. Assume that the cluster used to store these records consists of racks which has two DataNodes for processing each of 64 GB (1 GB = 2^{30}B) memory. Cluster consists of 120 racks. Replication factor 3. **[1 + 3 + 2 + 1 + 1 = 8]**

a) How the files of each student will be distributed at a cluster?
b) How many students' data can be stored at one cluster?
c) What is the total memory capacity of the cluster (in TB) and DataNodes in each rack?
d) How many racks will be required if only 5000 student's data needs to be stored?
e) What will be the changes when a stuData file size <= 128 MB?

Answers:

a) How the files of each student will be distributed at a cluster?
Answer: Data block default size is 64 MB. Each students file size is less than 64 MB. Therefore for each student file one data block will suffice.

b) How many students' data can be stored at one cluster?
Answer:
A data block is in a DataNode. Each rack has two DataNodes each of memory size 64 GB.
Each node can store 64 GB / 64 MB = 1024 data blocks = 1024 student files
Each rack thus store 2 * 1024 data blocks = 2048 student files

Each block default replicates three times in DataNodes.
Number of students whose data can be stored in the cluster
= number of racks * number of files / 3 = 120 * 2028 / 3 = 81920

c) What is the total memory capacity of the cluster (in TB)?
Answer:
Total memory capacity of cluster
= 120 * 128 GB = 15360 GB
= 15 TB

d) How many racks will be required if only 5000 student's data needs to be stored?
Answer:
One rack will store 2048 students' data
For storing 5000 students data = 5000 / 2048 = 2.44 racks = 3 racks will be required (without replication)
With replication = 3 * 3 = 9 racks will be required

e) What will be the changes when a stuData file size <= 128 MB?
Answer: Changes will be that each node will have half number of data blocks.

Q4. Consider an assembly consisting of three components with respective MTTF as 10, 20 and 15 minutes and same MTTR = 3 minutes. **[1 + 3 + 2 = 6]**

a) If these components are connected in serial assembly, what is the reliability of the system?

b) If these components are connected in parallel assembly, what is the reliability of the system?
c) What is the availability of the system in both cases?

Answers:

a) If these components are connected in serial assembly, what is the reliability of the system?
Answer =
Reliability = MTTF = 1 / 1/ M_a + 1 / M_b + 1 / M_c
= 1 / 1/10 + 1/20 + 1/15 = 1 / (0.1 + 0.05 + 0.06) = 4.61 minutes

b) If these components are connected in parallel assembly, what is the reliability of the system?
Answer:
Reliability = MTTF =

$$\text{System MTTF} = \frac{1}{\lambda_1} + \frac{1}{\lambda_2} - \frac{1}{\lambda_1 + \lambda_2} - \frac{1}{\lambda_2 + \lambda_3} - \frac{1}{\lambda_3 + \lambda_1} + \frac{1}{\lambda_1 + \lambda_2 + \lambda_3}$$

$\Lambda = 1 / MTTF_i$

$$MTTF_i = \frac{1}{\lambda}$$

λ 1 = 1 / 10 = 0.1
λ 2 = 1 / 20 = 0.05
λ 3 = 1 / 15 = 0.06
MMTF = 10 + 20 – 1/0.15 – 1 / 0.11 – 1 /0.16 + 1 / 0.21
= 30 – 10.05 – 9.09 – 6.25 + 4.76 = 9.37

c) What is the availability of the system in both cases?

Answer:
Availability = MTTF / (MTTF + MTTR)
For sequential case = Availability = 4.61 / (4.61 + 3) = 0.6
For parallel case = Availability = 9.37 / (9.37 + 3) = 0.75

Birla Institute of Technology and Science, Pilani

Work Integrated Learning Programmes Division Cluster
Programme - M.Tech in Data Science and Engg.
I Semester 2020-21 (Makeup- ANSWERKEY)

Course Number: DSECL ZG522
Course Name: Big Data Systems
Exam. Mode: Open Book
Weightage: 30%
Duration: ~~90 Minutes~~
Date: ~~29/03/2020 2 – 3.30PM~~

Pages: 4
Questions: 4
Marks: 30

Instructions

1. Questions are to be answered strictly in the order in which they appear in the question paper and in the page numbers specified in the answer-book.
2. **All answers must be directed to the question in short and simple paragraphs or bullet points; use visuals/diagrams wherever necessary; use last sheets of your answer-book for thinking and rough-work if necessary, and write legibly the final answer on the designated pages**.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

Q1. Consider a program for multiplying two large-scale N * N matrices, where N is matrix size. The sequential multiply time on a single server is $T_1 = cN^3$ minutes, where c is constant determined by the server used. A MPI-code parallel program requires $T_n = (cN^3 / n) + (dN^2 / n^{0.5})$ minutes to complete execution on n-server cluster system, where d is constant determined by MPI version used. Assume that program has a zero sequential bottleneck. The second term in T_n accounts for total message-passing overhead experienced by n servers.
Answer the following questions for a given cluster configuration with n = 64 servers, c = 0.8, d=0.1. Part (a, b) have a fixed workload corresponding to the matrix size N = 15000. Parts (c, d) have a scaled workload associated with an enlarged matrix size $N' = n^{1/3}N$. Assume the same cluster configuration to process both workloads. **[2 + 2 + 3 + 2 + 1 = 10]**

a) Using Amdahl's law, calculate the speedup of n-server cluster over single server.
b) What is the efficiency of the cluster system used in (a)?
c) Calculate the speedup in executing the scaled workload for enlarged N' * N' matrix on the same cluster configuration.
d) Calculate the efficiency of running the scaled workload in part(c).
e) Compare the above speedup and efficiency results and comment on their implications.

Answer:

bdsscan
05-Oct-2020 19-41-1

Q2. Answer the following questions in brief**: [2 * 4 = 8]**

a) Suppose that in a program 10% of the code cannot be parallelized i.e. it must be executed strictly sequentially. The other 90% of the code can be fully parallelized i.e. can be executed in data parallel mode. Suppose there is no constraint on the number of processors available. What is the maximum speedup that can be achieved?

Answer:
10
Amdahls' Law: Speedup is 1 / [f + (1−f)/p]

where f is the sequential part and p is number of processors.
If p is infinity (i.e. no constraint on processors available) then the maximum speedup is 1/f.
Here f is 0.1 (i.e. 10%), speedup = 1 / 0.1 = 10

b) Assume Sonu gave examination in Semester 1, 1999 in four subjects. She gave examination in five subjects in Semester 2, 1999. Another student, Monu gave examination in Semester 1, 2019 in three subjects, out of which one was theory and other two were practical. Assume the names of subjects and grades awarded for the same. Whether using JSON will be a better option for representing this data? Comment briefly.

Answer:
The structure of these two CSV files will be different as the attributes are different in them. If needs to be merged together for data processing, then need to arrive at decision how to handle missing values or columns. Whereas JSON provides us flexibility to represent this data in key-value pairs where just presence or absence of key will matter, no extra logic will be required to deal with missing key-values, so JSON will be a better representation for such a data which has different number of fields involved in it.

c) Consider an online food aggregation services available in the market. Identify two cases related to their data usage and classify those requirements using the data categories discussed in the class.
Answer:
1) Storing restaurant data, customer data, discounts etc – Structured data
2) Capturing the customer comments or feedbacks using text, image or videos – Unstructured data

d) Enlist four technologies that are making Big Data Analytics possible in today's era.
Answer:
- Batch processing technologies
- Stream processing technologies
- NoSQL databases
- Cluster setups on cloud

Each point = 0.5 marks, 0.5 * 4 = 2 marks

Q3. Apply the following Map-Reduce algorithm on the given data present in a file. Clearly show the outcome of map, sort and reduce phases. What is the objective of this task? **[2 + 1 + 2 + 1 = 6]**

```
Map(key, value)
{
        String str = value.toString();
        String part1 = str.substring(15,18);
        int i = 0;
        if(str.charAt(25) == '+')
        {
                i = str.substring(26,30).toInt();
        }
        emit( part1, i);
}
Reduce(key, value[])
{
        int i = 0, out1 = 0;
        for(each v in value)
        {
                if (v > out1 )
                 out1 = v;
        }
        emit (key, out1);
```

}
Data in the file -
0012345123412341950 5150N9+00001+99999
0012342342347841951 5151N9+00022+99998
0012456566896101950 5150N9+00023+99997
0012456566896101951 5150N9+00050+99996
0012456566896101952 5150N9+00012+99998
0012345123412341952 5150N9+00001+99999
0012342342347841953 5151N9+00022+99998
0012455668961019555 5150N9+00023+99997
0012456566896101958 5150N9+00050+99996
0012456566896101958 5150N9+00012+99998

Answer :
The line shows the weather information in cryptic information where year is embedded between places 15 to 18 followed by temperature values hidden between places 26 to 30. Students need to handwrite this algorithm to find out the outcome. Then looking at reduce phase they can figure out its listing the max temperature value for a given year. **It's okay if they are not able to get the context of weather data but should be able to tell that its finding out the max value from a given sort phase output.**
Map phase key will be line and value will be individual line in a file.
1950 ------ 00001
1951 ------ 00022
1950 ------ 00023
1951 ----- 00050
1952 ----- 00012
1952 -----00001
1953 ----- 00022
1955 ----- 00023
1958 ----- 00050
1958 -----00012
Sort phase outcome –
1950 ------ [00001, 00023]
1951 ----- [00022, 00050]
1952 ----- [00012, 00001]
1953 ----- [00022]
1955 ---- [00023]
1958 ---- [00050, 00012]
Reduce phase outcome ---
1950 ----- 00023
1951 ---- 00050
1952 ---- 00012
1953 ---- 00022
1955 ---- 00023
1958 --- 00050

Q4. Predictive Automotive Components Services (PACS) Company renders customer services for maintenance and servicing of (Internet) connected cars and its components. Assume that number of centers are 8192 ($=2^{13}$), number of car serviced by each center per day equals 32 ($=2^5$). Each car has 256 ($=2^8$) components, which requires maintenance or servicing in the Company's car. The service center also collects feedback after every service and send responses to customer requests. The feedback and responses text takes on average 128 B($=2^7$ B) and each service or responses records in a report of average 512 B ($= 2^9$) text. Company saves the centers data for maximum 10 years and follows last-in first-out data replacement policy. **[1 + 1 + 2 + 2 = 6]**

a) Justify why PACS is example of big data use case.
b) What are the big data system characteristics that you think will be important in this service?

c) How will the files of PACS be saved using big data file system like GFS or HDFS?
d) What shall be the minimum memory requirement (in MB) during 10 years?

Answer:

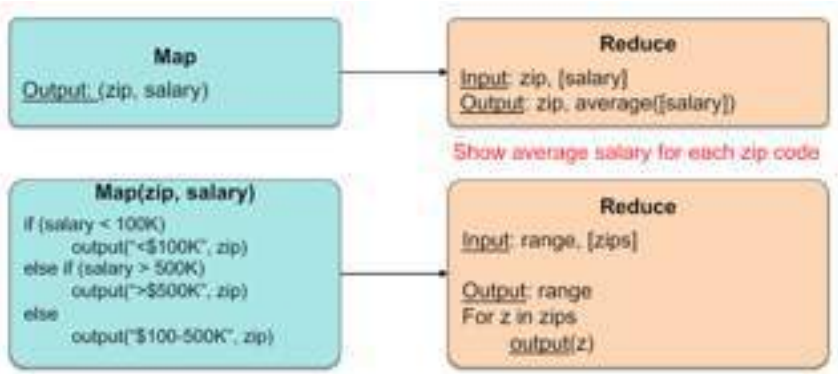
a) Its big data usecase as it caters to the "Volume" property of big data.
b) Big data system characteristics : scalability, reliability and maintainability
c) Data is stored in data blocks on the DataNodes. HDFS replicates those data blocks, usually 64MB in size, and distributes them so they are replicated within multiple nodes across the cluster.
   Here, Data files of size 512B (Service Report) and 128B (feedback) of each car requires total 640B (512B+128B). Therefore, each data block may store 64MB/640B, that is ~100K records.
   The default replication factor is 3, thus each block is replicated 3 times.
   There can be several blocks in a data node. Several data nodes can be present in Each Rack of cluster
d) The minimum memory requirement = [No. of centers * No. of cars serviced per day per center * No. of days in a year * 10 years * (Size of feedback + size of Service Report)] B
   = [$2^{13} * 2^{5} * 365 * 10 * 640$] B
   = 584000 MB

BDS-MIDSEM- MAKEUP

1.Write the logic of mapper and reducer classes to display zip codes where average salaries are in the ranges:
(1) < $100K (2) $100K … $500K (3) > $500K.
For the following data sets: {name, age, address, zip, salary}
(06 Marks)

ANSWER KEY:

Writing Mapper logic : 03 Marks
Writing Reducer logic: 03 Marks



2. Answer the following in brief: [2+2 = 04 marks]

I. Suppose that in a program 30% of the code cannot be parallelized i.e. it must be executed strictly sequentially. The other 70% of the code can be fully parallelized i.e. can be executed in data parallel mode. Suppose there is no constraint on the number of processors available. What is the maximum speedup that can be achieved?

ANSWER KEY:
Amdahls' Law: Speedup is 1 / [f + (1−f)/p]

where f is the sequential part and p is number of processors.
If p is infinity (i.e. no constraint on processors available) then the maximum speedup is 1/f.
Here f is 0.3 (i.e. 30%), speedup = 1 / 0.3 = 30%

II. Consider an online food aggregation services available in the market. Identify two cases related to their data usage and classify those requirements using the data categories discussed in the class.

ANSWER KEY:

Categorising data in to structured, semistructured, unstructured data in online food aggregation services available in the market. Justification for the same. 02 Marks

3. Answer the following Questions: (03+03+03 = 09Marks)

I. A GPS system need to provide the information on the real- time traffic and suggests taking an alternative route to reach the destination quickly. What type of big data analytical approach need to be performed in this scenario? Justify your answer.

ANSWER KEY:

In a GPS system, the prescriptive analytics provides information on the real- time traffic and suggests taking an alternative route to reach the destination quickly. Though it seems to be a overlap between the predictive and prescriptive analytics, real-time data is involved in the case of prescriptive analytics compared to predictive analytics scenario.

II. There is a need for storing transactional data generated by a card swapping machines. The data is to be stored in a tabular format. According to CAP theorem, which type of data store is to be used for this? Elaborate your answer.

ANSWER KEY:

CA

RDBMS systems will be required to store transactional data in structured format. As per CAP theorem, RDBMS systems support Consistency and Availability.

III. Which of the following problem can be solved using MapReduce? Why?

(A) Sorting
(B) Parallel search
(C) Fibonacci series
(D) Finding max in a list of numbers

ANSWER KEY:

All except (c) can be solved with map reduce as they can be done on separate systems as there is no state involved. For Fibonacci series calculation is dependent on previous state, hence cannot be done with MapReduce.

Identification of problems – 1 mark each, explanation one mark each

4. Explain big data lifecycle for online e-commerce platform. Suggest suitable big data platform for implementation. (05Marks)

ANSWER KEY:

Business Case Evaluation; Data Identification; Data Acquisition & Filtering; Data Extraction; Data Validation & Cleansing; Data Aggregation & Representation; Data Analysis; Data Visualization; Utilization of Analysis Results

Student need to map given problem in to different phases

Explanation of each phase for the given problem statement : 01marks * atleast 05 phases

5. Answer the following Questions: (02+02+02 = 06Marks)

I. An organisation is having high scalability requirement weather the organization need to adopt parallel computing environment or Distributed computing environment? Justify your answer.

ANSWER KEY:

Distributed computing

Justification for the answer

II. Consider a website for railways to provide information on the schedule of trains, a mechanism for online booking of tickets, and checking of the status on reservations and on current train timings. Identify the forms of data used (as in structured, semi structured, or unstructured).

ANSWER KEY:

All the raw data (i.e. schedule, online booking, reservation status, and train timings) are structured. [Additional descriptive information such as facilities information on stations, historical/ tourist information on locations, and images, video content etc. may be unstructured. Websites are likely to semi-structured: i.e. they may contain HTML and raw data (both structured) and other information that is unstructured.]

Talking about structured and semi structured data – 1 mark
Talking about unstructured data – 1 mark

III. You have been asked to design an "Open File System" which is derived from HDFS architecture with following variances ñ chunk size of 100 MB, block size of 1 MB. A file of size 1 TB needs to be stored on this file system.
How many chunks will be required to store the file and How many blocks will be required to store the file?

ANSWER KEY:

(01 Marks) Number of chunks needed

(01 Marks) Number of blocks needed

Qtext :

You are given a data set with the following fields in each record: name, age, address, zipcode, salary. You need to output zip codes where avg salary is in the range $100k- $500K using MapReduce. Show the logic of Mapper and Reducer functions to solve the given problem. **(04 Marks)**

User Answer :

Mapper 1 - Extracts a tuple of (zip, salary)

mapper1():

```
import sys

// Assume that the fields are comma separated (",")
for line in sys.stdin:
  data = line.strip().split(",")
  if len(data) == 5:
    // Ouputs (zip, salary)
    print(data[3], "\t", data[4])
```

Reducer 1 - Calculates average salary per zip as a tuple (zip, average(salary))

reducer1():

```
import sys

oldZip = None
salTotal = 0
salCount = 0

// Assume that the fields are tab separated
for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
    // assume something has gone wrong and skip

  zip, sal = data

  if oldZip and oldZip != zip:
    avgSal = salTotal/salCount
    print(oldZip, "\t", avgSal)
    oldZip = zip
    salTotal = 0
    salCount = 0

  oldZip = zip
  salTotal += float(sal)
  salCount += 1

if oldZip != None:
  avgSal = salTotal/salCount
  print(oldZip, "\t", avgSal)
```

Mapper 2 - Collects those zips whose salary is between 100K-500K

mapper2():

```
import sys:

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue

zip, sal = data

if ((sal>=100000) & (sal <= 500000)):
  print("100K-500K", "\t", zip)
else:
  print("Others", "\t", zip)
```

Reducer 2 - Print out all the zips with key as "100K-500K"

reducer2():

```
import sys

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue

  flag, zip = data
```

Qtext :

Write the pseudocode of mapper and reducer classes to find Call data records which has call type as 'sms' and call duration more than 10mins. Consider the datasets with the following fields: subscriber_phone_number, cell_id, timestamp, call_duration, phone_id, status and type_of_call. **(06 Marks)**

User Answer :

Mapper 1 - Extracts call records with type "sms"

mapper1():

```
import sys

for line in sys.stdin:
  # Assume data is comma separated (",")
  data = line.strip().split(",")
  if len(data)!=7:
    continue

  if (data[6] == "\"sms\""):
    for item in data:
      print(item, end="\t")
    print()
```

Reducer 1- Returns records with type "sms" and duration more than 10 mins

reducer1():

```
import sys

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=7:
    continue


  #(int(data[3])>=10):
    for item in data:
      print(item, end="\t")
    print()
```

Qtext :

"spotify, an on-demand music providing platform, uses Big Data Analytics, collects data from all its users around the globe, and then uses the analyzed data to give informed music recommendations and suggestions to every individual user".Illustrate different phases in the life cycle of this big data project. **(04 Marks)**

User Answer :

Given are the stages:

1. Business case evaluation

Spotify has to define a clear business case towards this project. In this case, it is to use the user information like music choices, follow-up songs, minutes played, etc to make a recommendation system to every individual user.
Justification - a good recommendation system will attract more users and retain existing users since good recommendations lead to more usage
Motivation - to build a large user base and be the most profitable music platform in the industry
Goals - To create a personalized recommendation system for each user based on the information collected from all users in the system

2. Data Identification

Almost all of the data sources will be internal since spotify will be able to get its users usage pattern to extract information needed for the recommendation system.
Here, identifying the relevant inputs is the most important. The inputs could be as follows:
- Minutes listened per song
- Which song was followed up after the current song
- Most listened genre
- Which decade of the songs most listened to?
- Songs liked/disliked

3. Data Acquisition and Filtering

In this phase, the data decided from the previous step is gathered and filtered. A basic data validity check is performed to see whether the data is useable or not.

4. Data Extraction

Most of the data here may not be in any strict structural format. This stage deals with converting all the unstructured data into a structured format which can then be passed to the model training.

5. Data Validation and Cleansing

In this stage, the extracted data undergoes a thorough integrity check. The data is checked for outliers, skewness, missing fields, etc.

6. Data Aggregation and Representation

In this stage, all the different structred data from the previous stage is collated into a single unified view. Any key related discrepancies are corrected and the data structure is made uniform.

7. Data Analysis

This stage contains the actual analysis that leads the feasible results. Here, a mix of descriptive, diagnostic and predictive analytics takes place in order to determine the users' behaviour and how that can be translated to getting a good score on their recommendation logic.
Once done, the test results are tested against any or all alternate hypotheses that may have been suggested during the course of the project. All the features are explored in order to see which features form the final feature list that define the recommendation system.

8. Data Visualization

In this stage, the results of the previous stage are visualized. These results could in the form of an accuracy index, a correlation plot, or something as simple as a probability matrix of the recommended music tracks based on the a test input.
This stage is important to explain the performance efficiency of the recommendation system to the business, since business does not necessarily have analytical skills to understand the numerical results.

9. Utilization of Analysis Results

In this final stage, the recommendation model is deployed to a small section of the user base to test out the feedback and the performance of their model. Once the results are in, stages 7-9 may be repeated to enhance the performance.
Finally the model is deployed for the entire user base.

Qtext :

It has been noted that there has been a random failure in one of the nodes of a cluster of 4 servers. The failure is observed always on the same node and is happening on the 25th of every month. An hour is taken to recover this node. Incidentally, the applications on these servers need an additional half hour to be started. Consider 30 days in a month. **(06 Marks)**

a) Calculate the availability of the cluster.
b) If the cost of downtime is $2k per hour, then what is the quarterly cost?

User Answer :

Q3

(a)
MTTR = 1 + 0.5 hours = 1.5 hours

Assuming, there are 30 days, the total hours = 30*24 = 720 hours
Therfore, since the node fails consistently on 25th of every month, we can say that the
MTTF = 720 hours

Therefore,
Availability = MTTF / (MTTF+MTTR) = 720 /721.5 = **0.99792 or 99.792%**

(b)
One quarter = 3 months = 90 days = 2160 hours
Downtime cost = $2000 per hour

Quarterly cost = 2000*1.5*2160/720 = **$9000**

Therfore, quarterly cost of downtime = $9000

Qtext :

You have a map-reduce program on a Hadoop cluster. If you run the program on a single node, it takes total 345 sec and 5% of the overall time is spent in a sequential reduce operation. The rest of the map reduce application code and runtime can be parallelized. **(06 Marks)**

(a) If you had a 10 node cluster with similar nodes for the same program and data set, how much time would you theoretically expect the program to take ?

(b) Is there any reason to expect higher time if you actually measured the program execution on the 10 node cluster ?

(c) Suppose you ran this program, with necessary modifications, but with a larger data set trying to accomplish more work with the 10 node cluster. What is a theoretical speed up you could target ?

User Answer :

Q2

T(1) = 345 seconds
f = 0.05

(a)
N = 10
Time taken by 10-node cluster = T(N) = f*T(1) + (1-f)*T(1)/N
T(10) = 0.05*345 + 0.95*345/10 = 17.25 + 32.775 = **50.025 seconds**
(Speedup = 6.896551724)

The program in the 10 node cluster would take 50.025 seconds.

(b)
The time of 50.025 seconds is a theoretical time that assumes fixed workload. Additionally, all system or communication overhead are ignored in this calculation. And finally, the I/O time or exception handling time is also not included the analysis. By factoring in these times, **it is definitely possible to get a higher time** with a 10 node cluster.

(c)
Using Gustafson's Law.
S(N) = f + (1-f)N

Therefore, S(10) = 0.05 + 0.95*10 = **9.55**

Therefore, theoretical speedup = 9.55

Qtext :

In the following types of application scenarios, point out what is most important - consistency or availability, when a system failure results in a network partition in the backend distributed DB. Explain briefly the reason behind your answer. **(04 Marks)**

(a) A limited quantity offer on a product for 100 items at an online retail store is almost 95% claimed

(b) An online signature campaign on a social media platform is gathering user input

(c) A hotel booking website is trying to sell rooms at a destination that is seeing very little uptake despite offers

(d) An online remote multi-player game needing a lot of sequential team coordination in executing a set of steps to reach a milestone.

User Answer :

Q1.

(a)
For a product with limited quantity, it is very important that the store does not sell the product to the customer when the product has actually been sold out. To ensure this, it is important that the DB is always updated to the latest numbers. Therefore, **consistency** is most important in this scenario since the all the partitions must report the same updated numbers.

(b)
For an online campaign that is collecting user input, the final tally of the scores is not necessary to be updated in real-time. In this scenario, it is more important to have the availability of the platform to the end-user so that they are encouraged to participate in the campaign.
Therefore, **availability** is most important in this scenario.

(c)
Since the hotel booking website is seeing very little uptake, it can be safe to assume that there is sufficient vacancy in the hotel. With this in mind, the priority of the website goes towards user experience towards booking rooms online. Therefore, **availability** will be more important. However, once most of the rooms have been booked, the system must quickly switch to *consistency* to avoid overbooking.

(d)
Since the key factor here is sequential team coordination, and the order of steps is very important, it is very important that the system prioritizes **consistency** over availability. That way the team will be working with the latest information and not have to make decisions based on obsolete data.

- Consider a website for railways to provide information on the schedule of trains, a mechanism for online booking of tickets, and checking of the status on reservations and on current train timings. Identify the forms of data used (as in structured, semi structured, or unstructured). [3]

- There is a need for storing transactional data generated by a card swapping machines. The data is to be stored in a tabular format. According to CAP theorem, which type of data store is to be used for this? Elaborate your answer. [2]

- There's a bag of differently numbered toys. The task is to find all the even numbered toys and transfer them to a different bag. For each toys the requirement is to test if the toys is having even number on it, and it is moved to a first bag; otherwise, it is moved to a second bag. Assume that there is function findAndMove which achieves this requirement. Based on this data, answer

the following questions:

i. Can this problem be solved using the Map construct? How?
ii. Consider the call map findAndMove(list), where list is a list of 50 toys. How many times is the findAndMove function called? Explain the logic behind it.
iii. A pseudocode for solving this problem is given below. Here, 'num' is the number on the toy, and 'bag-num' is the number on the bag.

```
//Toy is a class with field "num" denoting number on the toy
// returns a pair (num, bag-num)
findAndMove(Toy t)
??????????????????
        return (t.num, "FIRST");
else
        return (t.num, "SECOND")
```

Select the appropriate line that can fill the ??????????????????

iv. If toyList is a list of 100 Balls what is the output of the following call?
map findAndMove (toyList)

v. Let's assume the time taken for executing findAndMove once is 10 ns. Consider the call: map findAndMove (toyList), where toyList is a list of 1000 balls.
What is the ratio of time taken by sequential processing (i.e. map executes all calls to findAndMove - one after another - on one compute-node) over time taken by parallel processing with 1000 compute-nodes (i.e. map executes one call to findAndMove in each node)? Assume that any additional processing time required - other than calls to findAndMove - in both cases is negligible.

[2 + 2 + 2 + 2 + 3 = 11]

Q4. Attempt the following question related to system efficiency: **[4 + 2 + 2 + 2 + 2 = 12]**

i. Derive expression of system efficiency for fixed workload and scaled-workloads.
ii. If number of nodes in cluster is 100, sequential code is 40%, calculate the speedup achieved assuming fixed workload.
iii. If number of nodes in cluster is changed to 200 keeping other parameters same, calculate the speedup achieved assuming fixed workload.
iv. What's the conclusion drawn from the above two speedup numbers obtained in (ii) and (iii)?
v. Calculate the system efficiency for cluster consisting of 100 nodes and sequential code is 40%.

Suppose we have given two data files as follows.
movies.csv which contains three columns: -
Movie_ID: Unique ID for a movie.
Title: Title of the movie.
Year: Year of launch.
ratings.csv, which contains two columns: -
First_field: unique ID number for a movie
Second_field: IMDB rating of the movie

Write a simple Pig script to get a final Pig relation as shown below

Movie_ID	Title	Rating
1	ABC	4.5
2	XYZ	3.5

The dataset needs to be sorted in descending order w.r.t the rating column. **[10]**

Consider the below Java code to answer following Spark execution related questions:

[3 + 1 + 2 + 2 + 2+ 2 = 12]

a) What are the total number of transformations and actions operation performed on myRDD variable? List them out.

b) What will be the output of the first print statement?

c) What will be the output of the second print statement?

d) What will be the output from the last print statement?

e) If, instead of **"x -> x > 100 && x < 500 ? true : false"** we use **"x -> x > 100 && x < 500 ? true : true"** as the Boolean expression in the filter statement, then what will the call to the method "myRDD.count()" return?

f) Draw a lineage graph for the RDDs created.

```
public class Client
{
public static void main(String[] args)
{
System.setProperty("hadoop.home.dir", "d:\big_data");

SparkConf conf = new SparkConf().setAppName("MyFirstProgram").setMaster("local[*];
JavaSparkContext sc = new JavaSparkContext(conf);

JavaRDD<Integer> myRDD = sc.parallelize(Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9));

myRDD = myRDD.map(x -> x % 2 == 0 ? x * x : x * x * x);
System.out.println(myRDD.first()); //first print stmt

myRDD = myRDD.filter(x -> x > 100 && x < 500 ? true : false);
System.out.print(myRDD.count()); //second print stmt

List<Integer> list = myRDD.collect();
for (int i : list) {
        System.out.println(i);
} //end of for

sc.close();
} //end of main
} //end of class
```

1. a) If 10% of the program is sequential and 100 processors are used calculate the effective speedup (02Marks)

Answer:
1/(0.1 + 0.9/100) = 1/0.109 = 9.17

b) Consider a website for Airline reservation to provide information on the schedule of flights, a mechanism for online booking of tickets, and checking of the status on reservations and on current flight timings. Identify the forms of data used (as in structured, semi structured, or unstructured). (03Marks)

Answer:
All the raw data (i.e. schedule, online booking, reservation status, and flight timings) are structured. [Additional descriptive information such as facilities information on stations, historical/tourist information on locations, and images, video content etc. may be unstructured. Websites are likely to semi-structured: i.e. they may contain HTML and raw data (both structured) and other information that is unstructured.]

Talking about structured data – 1 mark
Talking about unstructured data – 1 mark
Talking about semi structured data – 1 mark

2.a) Take the following tuples as an example and group by MongoDB aggregation. Select all the attributes and groups by name throughout the records. (02 Marks)

{Name: Neil, x: 5, y: 3}
{Name: Nitin, z: 9}
{Name: Mukesh, x: 12, y: 2}

Answer:

We can do MongoDB aggregation as follows:

```
db.example.aggregate(
  {
    $group:{
      _id:'$name',
          x: {$addToSet: "$x"    },
          y: {$addToSet: "$y"    },
          z: {$addToSet: "$z"    },
     }
});
```

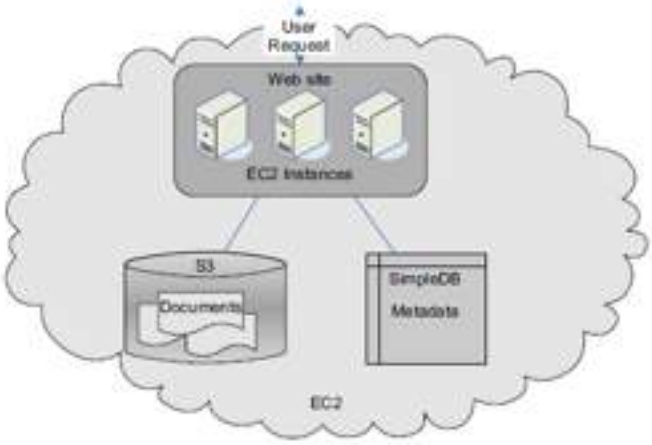
b) You have a file personal_data.txt in the HDFS directory with 100 records. You want to see only the first 5 records from the employee.txt file. How will you do this in Pig? (03 Marks)

Answer:

For getting only 5 records from 100 records we use limit operator.
First load the data in Pig:
personal_data = LOAD "/personal_data.txt" USING PigStorage(',') as (parameter1, Parameter2, …);
Then Limit the data to 5 records:
limit_data = LIMIT personal_data 5;

3.A) Design a book publishing portal by considering usage of following AWS services EC2, S3 and SimpleDB. (07 Marks)

ANSWER:



Student should demonstrate the usage of EC2, S3 and SimpleDB to build book publishing portal

b) Following points are related to Consistency. Provide True / False with justification (03 Marks)

a. Consistency has a same meaning across theorems

b. If a transaction is eventual consistent it implies it is casually consistent

c. In a distributed data system, consistency can be improved by adding more nodes

ANSWER:

Provide True / False with justification

a. Consistency has a same meaning across theorems - False

b. If a transaction is eventual consistent it implies it is casually consistent. - False

c. In a distributed data system, consistency can be improved by adding more nodes - False

4.a) Consider an online shopping site whose transaction workload increases during festive season like Christmas. So for this specific period of time, the resources need spike up. In order to handle this kind of situation what is the preferable choice? Is it cloud elasticity or cloud scalability give the reasons. (04 Marks)

ANSWER:

Ans: the Elasticity refers to the ability of a cloud to automatically expand or compressed the infrastructural resources on a sudden-up and down in the requirement so that the workload can be managed efficiently.

Scalability is commonly used where the persistent deployment of resources is required to handle the workload statically.

Hence we can go for Cloud-Elasticity service rather than Cloud Scalability. As soon as the season goes out, the deployed resources can then be requested for withdrawal.

b) In distributed DBs, such as Amazon DynamoDB and Cassandra, an architectural technique is used to handle temporary failures of nodes that prevent the failed node to receive a replica write request for some time. What is the technique and give a conceptual explanation of how it works with any relevant diagrams to explain data item and node mapping.

(Marks: 4)

Solution:
The technique is called hinted handoff or a “sloppy quorum” where first N healthy nodes are replica targets rather than N nodes in exact sequence. (1 mark)

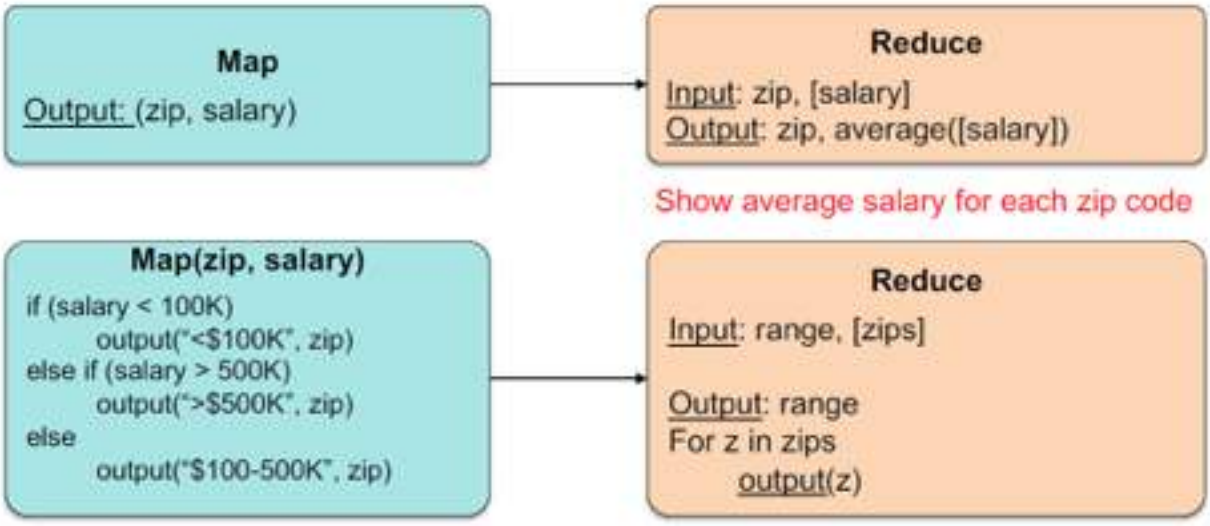
All data items and nodes are hashed into a common key space. Given a replication factor, e.g. 3, a data item, e.g. K, is mapped to 3 nodes (e.g. A, B, C) that follow the hashed value of the data item. However when A fails and cannot accept a replica, the data item is stored on the next following node D along with a hint that this item was actually designated for A. This is called a hinted handoff.
As part of group membership protocol, when D realises that A is up again, it forwards the data item to A, thus restoring the originally planned mapping. (3 marks)

5.a) Write the pseudocode of mapper and reducer classes to display zip codes where average salaries are in the ranges: (06 Marks)

(1) < $100K (2) $100K … $500K (3) > $500K.

For the following data sets: {name, age, address, zip, salary}

ANSWER:



b) Create a Hive table by name employees with the following fields and write the Hive query to solve the below requirements: (06 Marks)

i. Insert at least 6 rows into the table.

ii. Find all the employee names beginning with S

EMP_ID	EMP_NAME	DEPT	SAL
1023	Raj	Sales	25,000
1034	Ramesh	Marketing	30,000

6.a) For each of the following scenarios, pick a database that you will use with adequate reasons. Your choices are: MongoDB, Cassandra, Hive, Neo4j (or a similar graph DB). If you have picked a DB for a scenario avoid using it for another one unless there is no other option.

(a) An e-commerce site needs a DB to store all browse and buy interactions for each user. It wants to use the same DB technology but can tweak the configuration for different consistency options for the different types of transactions. E.g. buy and browse transactions can use different instances of the same DB technology with different configurations.
(b) A bank wants to store last 5 years worth of user transaction data in a warehouse for regulatory requirements as well as for historical data analysis.
(c) A system stores weather sensor monitoring data as key-value pairs collected across various regions in the country. The data is consumed into a streaming analysis platform to provide near real-time forecasts of local weather changes.
(d) An IT Network management system needs a DB technology to store the device network information along with various attributes of the devices. Each device can have different types of attributes. The key requirement is to enable root cause trouble shooting using the network data where one device or link failure can impact other devices and links. There can be thousands of devices of different types.

(Marks: 1.5 + 1.5 + 1.5 + 1.5)

Solution:

(a) MongoDB, because browse transactions can be stored fast on instance using eventual consistency configuration (e.g. read=write < majority nodes to survive failures but not requiring majority of nodes to acknowledge) whereas buy transactions can be stored on instance using causal consistency configuration (read=write=majority).
(b) Hive because it provides cost-effective warehouse data storage on Hadoop / HDFS and a SQL like query interface on structured data for regulatory compliance reporting and user data analysis.
(c) Cassandra because it is a geographically scalable key-value store that uses eventual consistency model that is useful for real time systems. Also has integration with Spark etc. for streaming data analysis.
(d) A graph DB like Neo4j will be suitable because each node can be a device or a device component and edges can be device connections (links) or relationships between various devices. Both devices and links can have variable sets of attributes. Root cause analysis of a problem needs graph traversal to navigate edges and nodes to find the problem.

Mention of correct DB name (1 mark) and reasonable explanation (0.5 marks).

b) Spark Core uses RDD as a base data structure. Explain the resiliency aspect of an RDD with specific details of how a computation that is a sequence of transformations and actions, can survive a node / executor crash or what happens when a node runs low on memory.

(Marks: 4)

Solution:

An RDD is partitioned between multiple nodes. Each time a transformation operation is applied on an RDD, a new RDD is created. Depending on a Narrow or a Wide transformation the partition of the new RDD is either created locally on the same node or is a result of data shuffling from multiple partitions across nodes. However, the lineage of a new RDD is stored across all transformations. This is called a DAG.

Only when an action operation is specified, the lineage is used to create the intermediate RDDs and the final non-RDD value. In the event of a node failure, the same DAG can be used to reconstruct the RDD to perform the action and get the value.

A value is always sent back to the driver program. An intermediate result in an RDD can be sent to the driver using collect() operation. (3 marks)

Further, when a node runs out of memory for local RDD partitions, the data can be spilled over onto disk automatically this avoiding out-of-memory issues. (1 mark)

1. An online retail company's product recommendation system runs in a Hadoop cluster of 100 nodes with a Master node. A cluster node on the average can fail every 150 hours. The Master is configured with a Secondary node that has to be brought up manually, but it takes 15 min using a script. Also, if a failure happens and once Secondary Master node is started, the analysis needs to be restarted and re-initialized, which takes about 45 min. **(04 Marks)**

a) What is the availability of this recommendation system ?

b) If the recommendation system is down then every hour estimated $50,000 is lost as business impact. What is the yearly cost of this downtime ?

ANSWER:

((02Marks + 02 Marks)

MTTF = 150 hours

Availability:
MTTF = 150 hours
MTTR = 8 hours
Availability = 150/ (158) = 94.9%

2. What is your choice for designing a distributed database by considering CAP theorem for the following cases. Justify your design choice in each case. **(04 Marks)**

a) tiger.com is a stock brokerage service company. They have requirement to display most recent price of stocks to its users. Display of older price is not desirable. If the prices are not consistent then some of the triggers may not hit and the user may suffer a serious loss!

b) fromkart.com is an ecommerce company that maintains database of millions of products. At the backend a bulk update of product prices need to be done, these processes will take an hour. Uses of fromkart.com have to wait for all prices to be updated before any read query takes place. This is not desirable as the users of the company will face a downtime resulting in loss of business. Company have to return the partially updated data till the full upload is done.

ANSWER:

(02Marks + 02 Marks)

i. Pick anCP database

ii.Pick an AP database.

3.

i. Suppose there is a file having size of 514MB is stored in a Hadoop (Hadoop 2.x) by using the default size-configuration of 128 MB block and also by default replication-factor. Then, how many blocks will be created in total and what will be the size of each block? **(04 Marks)**

ii. A program can achieve 67% parallelism. Assuming a fixed workload, draw a speedup graph considering different number of processors – 4; 16; 64; 256 What is the maximum theoretical speedup this program can ever achieve? **(04 Marks)**

ANSWER:

i.

The default replication factor is 3 and the default block-size is 128MB in Hadoop 2.x.
Thus, the 514MB of file can be split into:

A	B	C	D	E
128MB	128MB	128MB	128MB	2MB

•The default block size is 128MB
•Number of blocks: 514MB128MB=4.05 ≈5 blocks
•Replication factor: 3
•Total blocks: 5*3=15
•Total size: 514*3=1542MB

ii.

$S(p) = 1/ (f + (1-f)/p)$

No. of Processors	4	16	64	256
Speed Up	2.01005	2.689076	2.937127	3.006459

4. In each of the following scenarios, point out and give a brief reason what type of multi-processor computer one would use. The choices are (SIMD, SISD, MIMD, MISD).
(04 Marks)

(a) A scientific computing application does a sin(a) + tan(a) transformation for every variable a.

(b) An image processing application where multiple images are merged at corresponding pixel level with a certain operator F to create the result image. E.g. image3 = F(image1(i, j), image2(i,j), ...) for all pixels (i,j) in the input images.

(c) A Hadoop system for data analysis

(d) A development system with application code and a MongoDB running on a single multi-core laptop.

ANSWER:

(a)MISD (b) SIMD (c) MIMD (d) MIMD

(1+1+1+1 marks)

5. Figures (a) and (b) show two independent sequences of reads and writes for data items in a distributed DB across various application nodes P1, P2, P3, P4 on a time line. “R: A=V” at P1 means read operation on data item A returns value V at application node P1 and “W: B=V” at P2 means data item B is written with value V by application node P2. In each case, do you think the sequence demonstrates strict consistency or eventual consistency ? Give reasons. **(04 Marks)**

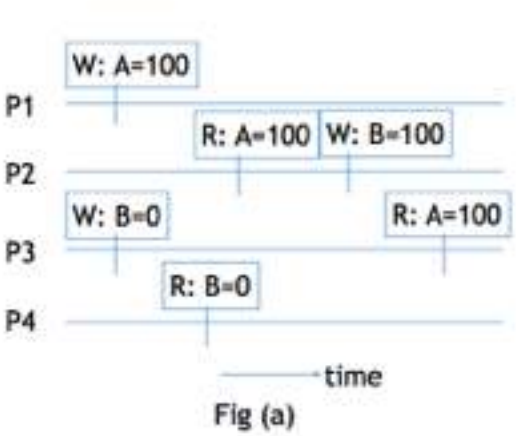
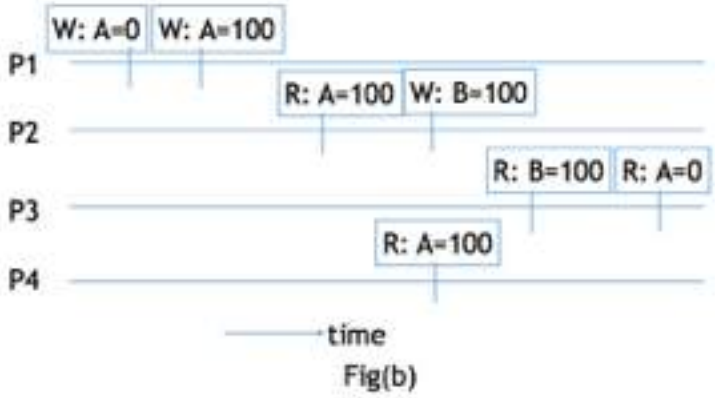


Fig (a)



Fig(b)

ANSWER:

(a)Strict (b) Eventual

(02Marks+ 02 Marks)

6. An online book retailer site recommends book titles by their rating. The historical data from book title reviews is stored in a file as show below where a book title can be reviewed by many users and a user can review many book titles. **(06 Marks)**

user	title	rating
U1	T1	2
U3	T2	2
U2	T1	3
U1	T2	4

...

The recommendation system using MapReduce programming creates Top 5 books by their max rating.

(a) Show the pseudo-code logic of the mapper
(b) Show the sample output format of mapper
(c) Show the pseudo-code logic of the reducer
(d) Show the output of shuffle sort that is fed to the reducer

Answer:

(2 + 1 + 2 + 1 marks)

Answer:
(a)

```
class Mapper {

    TreeMap topN; // stores a sorted list of topN rating records
for each key

    <key, value> map (user, record) {

        value = record[rating]
        key = record[title] + '_' + record[category]
        topN.put(value, key)
        if topN.size() > 5:
            topN.remove(topN.firstKey()
    }
    at the end of each Mapper instance {
        for each entry <k,v>  in topN:
            emit(k, v)
    }
}
```

(b)

```
map 1:   map2:    …
<T1, 3>  <T2, 4>
<T1, 2>  <T2, 2>
…        …
```

(c)

```
class Reducer {

    TreeMap topN; // stores a sorted list of topN rating records
for each key

    <key, value> reduce (user, record) {

        value = record[rating]
        key = record[title] + '_' + record[category]
        topN.put(value, key)
        if topN.size() > 5:
            topN.remove(topN.firstKey()
    }
    at the end of each Mapper instance {
        for each entry <k,v>  in topN:
            emit(k, v)
    }
}
```

(d)

```
<T1, [3, 2, ..]>
<T2, [4, 2, ..]>
…
```

Qtext :

In the following types of application scenarios, point out what is most important - consistency or availability, when a system failure results in a network partition in the backend distributed DB. Explain briefly the reason behind your answer. **(04 Marks)**

(a) A limited quantity offer on a product for 100 items at an online retail store is almost 95% claimed

(b) An online signature campaign on a social media platform is gathering user input

(c) A hotel booking website is trying to sell rooms at a destination that is seeing very little uptake despite offers

(d) An online remote multi-player game needing a lot of sequential team coordination in executing a set of steps to reach a milestone.

User Answer :

Q1.

(a)

For a product with limited quantity, it is very important that the store does not sell the product to the customer when the product has actually been sold out. To ensure this, it is important that the DB is always updated to the latest numbers. Therefore, **consistency** is most important in this scenario since the all the partitions must report the same updated numbers.

(b)

For an online campaign that is collecting user input, the final tally of the scores is not necessary to be updated in real-time. In this scenario, it is more important to have the availability of the platform to the end-user so that they are encouraged to participate in the campaign.
Therefore, **availability** is most important in this scenario.

(c)

Since the hotel booking website is seeing very little uptake, it can be safe to assume that there is sufficient vacancy in the hotel. With this in mind, the priority of the website goes towards user experience towards booking rooms online. Therefore, **availability** will be more important. However, once most of the rooms have been booked, the system must quickly switch to *consistency* to avoid overbooking.

(d)

Since the key factor here is sequential team coordination, and the order of steps is very important, it is very important that the system prioritizes **consistency** over availability. That way the team will be working with the latest information and not have to make decisions based on obsolete data.

Qtext :

You have a map-reduce program on a Hadoop cluster. If you run the program on a single node, it takes total 345 se and 5% of the overall time is spent in a sequential reduce operation. The rest of the map reduce application code ar runtime can be parallelized. **(06 Marks)**

(a) If you had a 10 node cluster with similar nodes for the same program and data set, how much time would you theoretically expect the program to take ?

(b) Is there any reason to expect higher time if you actually measured the program execution on the 10 node cluster ?

(c) Suppose you ran this program, with necessary modifications, but with a larger data set trying to accomplish more work with the 10 node cluster. What is a theoretical speed up you could target ?

User Answer :

Q2

T(1) = 345 seconds
f = 0.05

(a)
N = 10
Time taken by 10-node cluster = T(N) = f*T(1) + (1-f)*T(1)/N
T(10) = 0.05*345 + 0.95*345/10 = 17.25 + 32.775 = **50.025 seconds**
(Speedup = 6.896551724)

The program in the 10 node cluster would take 50.025 seconds.

(b)
The time of 50.025 seconds is a theoretical time that assumes fixed workload. Additionally, all system or communication overhead are ignored in thi calculation. And finally, the I/O time or exception handling time is also not included the analysis. By factoring in these times, **it is definitely possible to get a higher time** with a 10 node cluster.

(c)
Using Gustafson's Law:
S(N) = f + (1-f)N

Therefore, S(10) = 0.05 + 0.95*10 = **9.55**

Therefore, theoretical speedup = 9.55

Qtext :

It has been noted that there has been a random failure in one of the nodes of a cluster of 4 servers. The failure is observed always on the same node and is happening on the 25th of every month. An hour is taken to recover this node. Incidentally, the applications on these servers need an additional half hour to be started. Consider 30 days in a month. **(06 Marks)**

a) Calculate the availability of the cluster.
b) If the cost of downtime is $2k per hour, then what is the quarterly cost?

User Answer :

Q3.

(a)
MTTR = 1 + 0.5 hours = 1.5 hours

Assuming, there are 30 days, the total hours = 30*24 = 720 hours
Therfore, since the node fails consistently on 25th of every month, we can say that the
MTTF = 720 hours

Therefore,
Availability = MTTF / (MTTF+MTTR) = 720 /721.5 = **0.99792 or 99.792%**

(b)
One quarter = 3 months = 90 days = 2160 hours
Downtime cost = $2000 per hour

Quarterly cost = 2000*1.5*2160/720 = **$9000**

Therfore, quarterly cost of downtime = $9000

Close

Qtext :

”spotify, an on-demand music providing platform, uses Big Data Analytics, collects data from all its users around the globe, and then uses the analyzed data to give informed music recommendations and suggestions to every individual user".illustrate different phases in the life cycle of this big data project. **(04 Marks)**

User Answer :

Given are the stages:

1. Business case evaluation

Spotify has to define a clear business case towards this project. In this case, it is to use the user information like music choices, follow-up songs, minutes played, etc to make a recommendation system to every individual user.
Justification - a good recommendation system will attract more users and retain existing users since good recommendations lead to more usage
Motivation - to build a large user base and be the most profitable music platform in the industry
Goals - To create a personalized recommendation system for each user based on the information collected from all users in the system

2. Data Identification

Almost all of the data sources will be internal since spotify will be able to get its users usage pattern to extract information needed for the recommendation system.
Here, identifying the relevant inputs is the most important. The inputs could be as follows:
- Minutes listened per song
- Which song was followed up after the current song
- Most listened genre
- Which decade of the songs most listened to?
- Songs liked/disliked

3. Data Acquisition and Filtering

In this phase, the data decided from the previous step is gathered and filtered. A basic data validitiy check is performed to see whether the data is useable or not.

4. Data Extraction

Most of the data here may not be in any strict structural format. This stage deals with converting all the unstructured data into a structured format which can then be passed to the model training.

5. Data Validation and Cleansing

In this stage, the extracted data undergoes a thorough integrity check. The data is checked for outliers, skewness, missing fields, etc.

6. Data Aggregation and Representation

In this stage, all the different structred data from the previous stage is collated into a single unified view. Any key related discrepancies are corrected

Goals - To create a personalized recommendation system for each user based on the information collected from all users in the system

2. Data Identification
Almost all of the data sources will be internal since spotify will be able to get its users usage pattern to extract information needed for the recommendation system.
Here, identifying the relevant inputs is the most important. The inputs could be as follows:
- Minutes listened per song
- Which song was followed up after the current song
- Most listened genre
- Which decade of the songs most listened to?
- Songs liked/disliked

3. Data Acquisition and Filtering
In this phase, the data decided from the previous step is gathered and filtered. A basic data validitiy check is performed to see whether the data is useable or not.

4. Data Extraction
Most of the data here may not be in any strict structural format. This stage deals with converting all the unstructured data into a structured format which can then be passed to the model training.

5. Data Validation and Cleansing
In this stage, the extracted data undergoes a thorough integrity check. The data is checked for outliers, skewness, missing fields, etc.

6. Data Aggregation and Representation
In this stage, all the different structred data from the previous stage is collated into a single unified view. Any key related discrepancies are corrected and the data structure is made uniform.

7. Data Analysis
This stage contains the actual analysis that leads the feasible results. Here, a mix of descriptive, diagnostic and predictive analytics takes place in order to determine the users' behaviour and how that can be translated to getting a good score on their recommendation logic.
Once done, the test results are tested against any or all alternate hypotheses that may have been suggested during the course of the project. All the features are explored in order to see which features form the final feature list that define the recommendation system.

8. Data Visualization
In this stage, the results of the previous stage are visualized. These results could in the form of an accuracy index, a correlation plot, or something as simple as a probability matrix of the recommended music tracks based on the a test input.
This stage is important to explain the performance efficiency of the recommendation system to the business, since business does not necessarily have analytical skills to understand the numerical results.

9. Utilization of Analysis Results
In this final stage, the recommendation model is deployed to a small section of the user base to test out the feedback and the performance of their model. Once the results are in, stages 7-9 may be repeated to enhance the performance.
Finally the model is deployed for the entire user base.

Qtext :

Write the pseudocode of mapper and reducer classes to find Call data records which has call type as ‘sms’ and call duration more than 10mins. Consider the datasets with the following fields: subscriber_phone_number, cell_id, timestamp, call_duration, phone_id, status and type_of_call. **(06 Marks)**

User Answer :

Mapper 1 - Extracts call records with type "sms"

mapper1():

```
import sys

for line in sys.stdin:
  // Assume data is comma separated (",")
  data = line.strip().split(",")
  if len(data)!=7:
    continue

  if (data[6] == '\'sms\''):
    for item in data:
      print(item, end='\t')
    print()
```

Reducer 1- Returns records with type "sms" and duration more than 10 mins

reducer1():

```
import sys

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=7:
    continue


  if(int(data[3]>=10):
    for item in data:
      print(item, end='\t')
    print()
```

Qtext :

You are given a data set with the following fields in each record: name, age, address, zipcode, salary. You need to output zip codes where avg salary is in the range $100k- $500K using MapReduce. Show the logic of Mapper and Reducer functions to solve the given problem. **(04 Marks)**

User Answer :

Mapper 1 - Extracts a tuple of (zip, salary)

mapper1():

```
import sys

// Assume that the fields are comma separated (",")
for line in sys.stdin:
  data = line.strip().split(",")
  if len(data) == 5:
    // Ouputs (zip, salary)
    print(data[3], "\t", data[4])
```

Reducer 1 - Calculates average salary per zip as a tuple (zip, average(salary))

reducer1():

```
import sys

oldZip = None
salTotal = 0
salCount = 0

// Assume that the fields are tab separated
for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
    // assume something has gone wrong and skip

  zip, sal = data

  if oldZip and oldZip != zip:
    avgSal = salTotal/salCount
```

Reducer 1 - Calculates average salary per zip as a tuple (zip, average(salary))

reducer1():

```
import sys

oldZip = None
salTotal = 0
salCount = 0

// Assume that the fields are tab separated
for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
    // assume something has gone wrong and skip

  zip, sal = data

  if oldZip and oldZip != zip:
    avgSal = salTotal/salCount
    print(oldZip, "\t", avgSal)
    oldZip = zip
    salTotal = 0
    salCount = 0

  oldZip = zip
  salTotal += float(sal)
  salCount += 1

if oldZip != None:
avgSal = salTotal/salCount
print(oldZip, "\t", avgSal)
```

Mapper 2 - Collects those zips whose salary is between 100K-500K

mapper2():

```
import sys:

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
```

```
  salCount += 1

 if oldZip != None:
 avgSal = salTotal/salCount
 print(oldZip, "\t", avgSal)
```

Mapper 2 - Collects those zips whose salary is between 100K-500K

mapper2():

```
import sys:

 for line in sys.stdin:
   data = line.strip().split("\t")
   if len(data)!=2:
     continue

 zip, sal = data

 if ((sal>=100000) & (sal <= 500000)):
   print("100K-500K", "\t", zip)
 else:
   print("Others", "\t", zip)
```

Reducer 2 - Print out all the zips with key as "100K-500K"

reducer2():

```
 import sys

 for line in sys.stdin:
   data = line.strip().split("\t")
   if len(data)!=2:
     continue

   flag, zip = data

   if flag == "100K-500K":
     print(zip) // prints the final required zips
```

BDS EC2R solutions

Question 1

(a) Eventual consistency. Reason: US1 reads CT and ST and then updates ST and CT. So there is a causal dependency inferred between these variables where write of ST will require the latest value of ST and CT. On US2, there is a write for CT but it does not have the latest value of CT or ST. Hence causal consistency is violated and this can only be eventual consistency.

(b) Given the network may fail, P (partition tolerance) is always important and C (consistency) is required for limited seats to avoid conflict. Hence CP combination if required.

(c) Given enough seats, consistency of the remaining seats is not important for a reservation. So A (availability) takes precedence so that users are able to book. Hence AP combination is required.

Question 2

1. S(100) = 1 / (0.17 + (1-0.17)/100) = 5.6 using Amdahl's Law
2. S(inf) = 1/0.17 = 5.9 using Amdahl's Law
3. S(100) = 0.1 + (1-0.1)*100 = 90.1 using Gustafson-Barsis Law

Question 3

map.py

```
# import csv
import csv

# initialize rows
rows=[]

# csv file name
filename = "Agri.csv"

# reading csv file
with open(filename, 'r') as csvfile:
      # creating a csv reader object
      csvreader = csv.reader(csvfile)

      # extracting each data row one by one
      for row in csvreader:
            rows.append(row)

# map function produces k, v pairs with locid as key
# so that all entries of a locid go to the same reducer
for row in rows:
      print (row[0],row[1],row[2])
```

reduce.py

```
# takes key-value pairs (locid as key) and averages the entries
# outputs top and bottom locids based on SFI

import sys
locations = []
row = []
temp = {}

def checkLocationValue(row,locations):
    for location in locations:
        if(location[0] == row[0]):
            location[2] = int(location[2]) + int(row[2])
            #temp[location[0]] = temp[location[0]] + 1
            location[1] = location[1]+1
            return 1
    return 0

for line in sys.stdin:
    line=line.strip()
    row=line.split()
    if checkLocationValue(row,locations) == 0:
        row[1]=1
        locations.append(row)

print("LocID \t SFI")
for row in locations:
    if (float(row[2])/int(row[1])) > 80:
        print(row[0],"\t",float(row[2])/int(row[1]))
    elif(float(row[2])/int(row[1])) < 20:
        print(row[0],"\t",float(row[2])/int(row[1]))
```

sample data

```
450   125   90
450   126   90
451   130   4
451   130   6
452   145   26
453   155   36
453   156   89
454   166   23
455   178   3
450   127   90
450   128   90
```

Output from sample data by simulating a map-reduce runtime. You could write Java code or python code that runs using Hadoop streaming. But the ask was to present good pseudo code that has this kind of structure. Should show how key-value pairs are generated by mapper with locid as key and then reducer processes a set of values for a key and produces output.

```
$ python map.py
450 125 80
450 126 85
451 130 4
451 130 6
452 145 26
453 155 36
453 156 89
```

```
454 166 23
455 178 3

$ python map.py | python reduce.py
LocID SFI
450 90.0
451 5.0
455 3.0
```

Question 4

(a) 10 = h * 5 + (1-h) * (30 + 5), h = 0.83 or 83%. Since data is always accessed from cache, miss penalty is 30+5.

(b) First get a historical query trace. Study 3 things from the data accessed by the queries: (1) temporal locality, i.e. do subsequent queries access the same data within a time window (2) Spatial locality, i.e. do subsequent queries use data in the neighborhood or adjacent data. Spatial and temporal locality will help to determine if there is any LoR in the queried data to exploit cache. (3) What is the working set size S of the queries, i.e. if we cache S amount of data then it will lead to a substantial cache hit from temporal or spatial locality. This will help decide the cache size.

Question 5

(a)

MTTF = 120 days = 120*24 hrs

MTTR = 30 + 15 = 45 min = 0.75 hrs

Availability = MTTF*100 / (MTTF + MTTR)=120*24*100/(120*24+0.75) = 99.97 %

(b) 2 points to improve Availability: (1) Increase MTTF - which can only be achieved up to a limit. E.g. Master Node hardware and software can be improved to an extent. (2) Reduce MTTR - which can be done through replication, standby nodes esp with an active/active config so that repair of failed node can be done in the background while the system is available.

(c) Cost = ((1-99.97) / 100) * 365 * 24 * 3000 = USD 7884

Question 6

(a) A fair scheduling policy may be preferred because

- A fair scheduler using weights avoids min-max config and provides a simpler control using a single number to share resources where numbers don't need to add up to 100.

- It avoids starvation of shorter jobs when long jobs cannot be preempted in a capacity scheduler.

(b) Preemption config can be done for queue1 with no such config for queue2. Both queues can have equal weights. In queue1 we can configure a threshold and a timeout to ensure that when queue1 is at 90% util for at least 1min, resources can be taken from queue2 and allocated to queue1 by preempting jobs in queue2. Also, for optimization, the scheduler will try to make sure that specific jobs are selected that have not progressed long, e.g. young applications or map tasks, so that large computation effort is not wasted.

(c) User limit factor can be set to 2 so that max allocation of a user can be within 2 * 10% (min) = 20%.

BDS EC2 Makeup solutions

Question 1

(a) Eventual consistency because Y is dependent on X (because P2 reads X and then writes W) but P3 violates the causal order in reads because it reads new value of Y but old value of X.
(b) Causal consistency because no causal order is violated (there is no causal dependency between X and Y). Not strict consistency because P3 reads the old value of Y as it is getting written (it is linearizable but not strict).

Question 2

(a) S(4) = 1 / (.35 + 0.65/4) = 1.95, S(16) = 2.56
(b) S(inf) = 2.8
(c) S(64) = .35 + .65 * 64 = 41.95
(d) 50 = 25 * h + (1-h) * (100+25) = .75 or 75%

Question 3

mapper.py —

```
import csv
rows=[]

# csv file name
filename = "Errors.csv"

# reading csv file
with open(filename, 'r') as csvfile:
        # creating a csv reader object
        csvreader = csv.reader(csvfile)

        # extracting each data row one by one
        for row in csvreader:
                rows.append(row)

# map function
for row in rows:
        print '%s\t%s\t%s' % (row[1],row[2],row[3])
```

reducer.py —

```
# import sys to read and write data
import sys
# initializing machines and rows
machines = []
row = []

# to check whether machine value appended to machines
def checkMachineValue(row,machines):
   for machine in machines:
        # check whether the vale is already existing in the list
        if(machine[0] == row[0] and machine[1] == row[1]):
                # add the error value to existing value
                machine[2] = int(machine[2]) + int(row[2])
                return 1
```

```
    return 0


# reading data from stdin
for line in sys.stdin:
        # to remove leading and trailing whitespace
        line=line.strip()
        # split the line into words
        row=line.split()

        # call the function to check whether the row is already in the list
        if checkMachineValue(row,machines) == 0:
                # if not already exists, append to machines
                machines.append(row)

# check whether the machines "at-risk"
for row in machines:
        # if total value of HIGH >100 : print the details of machine
        if ( row[1] == 'HIGH' and  int(row[2]) > 100 ):
                print(row)
        # if total value of LOW >500: print the details of machine
        elif ( row[1] == 'LOW' and int(row[2]) > 500 ):
                print(row)
```

You could write Java code or Python code that runs using Hadoop streaming. But the ask was to present good pseudo code that has this kind of structure even if you have not used Hadoop libs.

Question 4

1. AP because no consistency is required with sufficient seats remaining
2. AP because no consistency is required because feedback posts are independent
3. CA because consistency is needed and less chance of failures at local venue DB but availability is required
4. CA because consistency is needed and less chance of failures at centralized DB but availability is required
5. CP because consistency needed for counting limited coupons for item and large scale systems needs partition tolerance

Question 5

1. 1 / (1/(50*3) + 1/(100*2)) = 85.7 days

2. Cluster availability = 1 - (Cluster unavailability) = 1- (node 1 unavailability * node 2 unavailability) = 1 - 0.02 * 0.02 = 0.9996 or 99.96%

3. (3.5/100) * 365 * 24 * 20000 = USD 6,132,000

Question 6

(a) Total data set size = 800M records
Each Mapper does age > 21 filter and output of 2 Mappers contain total 600M key-value records of size 1KB each.
There are 2 nodes. Mapper runs on node 1 and node 2. Say, Reducer runs on node 2. So node 1 Mapper data has to be sent to node 2.

In the average case, there are 300M records on node 1 and 300M on node 2 if filtered data (age>21) is uniformly distributed across all 800M records. So data sent to the Reducer : 300M x 1KB or 300GB.
In the worst case, there are 600M records on node 1 and no records produced on node 2. So data sent to the Reducer : 600M x 1KB or 600GB. However, this case is unlikely because Hadoop runtime should rather run the reducer on node 1 in the extreme case.

(b)

```
data = LOAD 'PERSONS.CSV' using PigStorage(',') as
(name:chararray, age:int, ward:chararray, education_level:int);
data2 = FILTER data by (age>21);
data3 = GROUP data2 by ward
result = FOREACH data3 GENERATE $0, AVG(data2.education_level)
dump result
```

1. An online retail company's product recommendation system runs in a Hadoop cluster of 100 nodes with a Master node. A cluster node on the average can fail every 150 hours. The Master is configured with a Secondary node that has to be brought up manually, but it takes 15 min using a script. Also, if a failure happens and once Secondary Master node is started, the analysis needs to be restarted and re-initialized, which takes about 45 min. **(04 Marks)**

a) What is the availability of this recommendation system ?

b) If the recommendation system is down then every hour estimated $50,000 is lost as business impact. What is the yearly cost of this downtime ?

ANSWER:

((02Marks + 02 Marks)

MTTF = 150 hours

Availability:
MTTF = 150 hours
MTTR = 8 hours
Availability = 150/ (158) = 94.9%

2. What is your choice for designing a distributed database by considering CAP theorem for the following cases. Justify your design choice in each case. **(04 Marks)**

a) tiger.com is a stock brokerage service company. They have requirement to display most recent price of stocks to its users. Display of older price is not desirable. If the prices are not consistent then some of the triggers may not hit and the user may suffer a serious loss!

b) fromkart.com is an ecommerce company that maintains database of millions of products. At the backend a bulk update of product prices need to be done, these processes will take an hour. Uses of fromkart.com have to wait for all prices to be updated before any read query takes place. This is not desirable as the users of the company will face a downtime resulting in loss of business. Company have to return the partially updated data till the full upload is done.

ANSWER:

(02Marks + 02 Marks)

i. Pick anCP database

ii.Pick an AP database.

3.

i. Suppose there is a file having size of 514MB is stored in a Hadoop (Hadoop 2.x) by using the default size-configuration of 128 MB block and also by default replication-factor. Then, how many blocks will be created in total and what will be the size of each block? **(04 Marks)**

ii. A program can achieve 67% parallelism. Assuming a fixed workload, draw a speedup graph considering different number of processors – 4; 16; 64; 256 What is the maximum theoretical speedup this program can ever achieve? **(04 Marks)**

ANSWER:

i.

The default replication factor is 3 and the default block-size is 128MB in Hadoop 2.x.
Thus, the 514MB of file can be split into:

A	B	C	D	E
128MB	128MB	128MB	128MB	2MB

•The default block size is 128MB
•Number of blocks: 514MB128MB=4.05 ≈5 blocks
•Replication factor: 3
•Total blocks: 5*3=15
•Total size: 514*3=1542MB

ii.

S(p) = 1/ (f + (1-f)/p)

No. of Processors	4	16	64	256
Speed Up	2.01005	2.689076	2.937127	3.006459

4. In each of the following scenarios, point out and give a brief reason what type of multi-processor computer one would use. The choices are (SIMD, SISD, MIMD, MISD).
(04 Marks)

(a) A scientific computing application does a sin(a) + tan(a) transformation for every variable a.

(b) An image processing application where multiple images are merged at corresponding pixel level with a certain operator F to create the result image. E.g. image3 = F(image1(i, j), image2(i,j), ...) for all pixels (i,j) in the input images.

(c) A Hadoop system for data analysis

(d) A development system with application code and a MongoDB running on a single multi-core laptop.

ANSWER:

(a)MISD (b) SIMD (c) MIMD (d) MIMD

(1+1+1+1 marks)

5. Figures (a) and (b) show two independent sequences of reads and writes for data items in a distributed DB across various application nodes P1, P2, P3, P4 on a time line. “R: A=V” at P1 means read operation on data item A returns value V at application node P1 and “W: B=V” at P2 means data item B is written with value V by application node P2. In each case, do you think the sequence demonstrates strict consistency or eventual consistency ? Give reasons. **(04 Marks)**

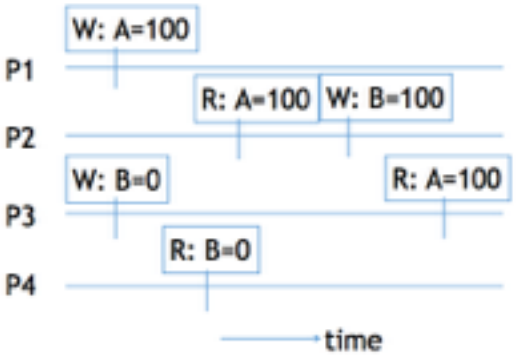
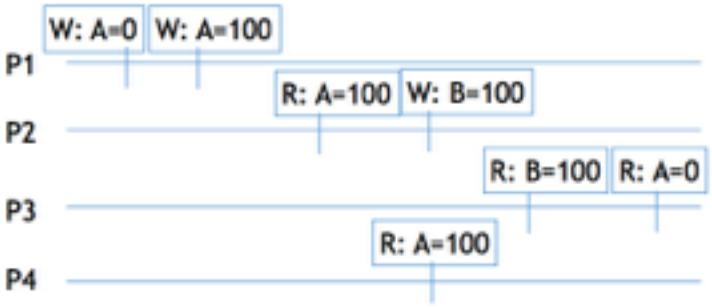


Fig (a)



Fig(b)

ANSWER:

(a)Strict (b) Eventual

(02Marks+ 02 Marks)

6. An online book retailer site recommends book titles by their rating. The historical data from book title reviews is stored in a file as show below where a book title can be reviewed by many users and a user can review many book titles. **(06 Marks)**

user	title	rating
U1	T1	2
U3	T2	2
U2	T1	3
U1	T2	4

...

The recommendation system using MapReduce programming creates Top 5 books by their max rating.

(a) Show the pseudo-code logic of the mapper
(b) Show the sample output format of mapper
(c) Show the pseudo-code logic of the reducer
(d) Show the output of shuffle sort that is fed to the reducer

Answer:

(2 + 1 + 2 + 1 marks)

Answer:
(a)

```
class Mapper {

    TreeMap topN; // stores a sorted list of topN rating records
for each key

    <key, value> map (user, record) {

        value = record[rating]
        key = record[title] + '_' + record[category]
        topN.put(value, key)
        if topN.size() > 5:
            topN.remove(topN.firstKey()
    }
    at the end of each Mapper instance {
        for each entry <k,v>  in topN:
            emit(k, v)
    }
}
```

(b)

map 1:	map2:	…
<T1, 3>	<T2, 4>	
<T1, 2>	<T2, 2>	
…	…	

(c)

```
class Reducer {

    TreeMap topN; // stores a sorted list of topN rating records
for each key

    <key, value> reduce (user, record) {

        value = record[rating]
        key = record[title] + '_' + record[category]
        topN.put(value, key)
        if topN.size() > 5:
            topN.remove(topN.firstKey()
    }
    at the end of each Mapper instance {
        for each entry <k,v>  in topN:
            emit(k, v)
    }
}
```

(d)

<T1, [3, 2, ..]>
<T2, [4, 2, ..]>
…

1. a) If 10% of the program is sequential and 100 processors are used calculate the effective speedup (02Marks)

Answer:
1/(0.1 + 0.9/100) = 1/0.109 = 9.17

b) Consider a website for Airline reservation to provide information on the schedule of flights, a mechanism for online booking of tickets, and checking of the status on reservations and on current flight timings. Identify the forms of data used (as in structured, semi structured, or unstructured). (03Marks)

Answer:
All the raw data (i.e. schedule, online booking, reservation status, and flight timings) are structured. [Additional descriptive information such as facilities information on stations, historical/tourist information on locations, and images, video content etc. may be unstructured. Websites are likely to semi-structured: i.e. they may contain HTML and raw data (both structured) and other information that is unstructured.]

Talking about structured data – 1 mark
Talking about unstructured data – 1 mark
Talking about semi structured data – 1 mark

2.a) Take the following tuples as an example and group by MongoDB aggregation. Select all the attributes and groups by name throughout the records. (02 Marks)

{Name: Neil, x: 5, y: 3}
{Name: Nitin, z: 9}
{Name: Mukesh, x: 12, y: 2}

Answer:

We can do MongoDB aggregation as follows:

```
db.example.aggregate(
  {
    $group:{
      _id:'$name',
          x: {$addToSet: "$x"    },
          y: {$addToSet: "$y"    },
          z: {$addToSet: "$z"    },
    }
});
```

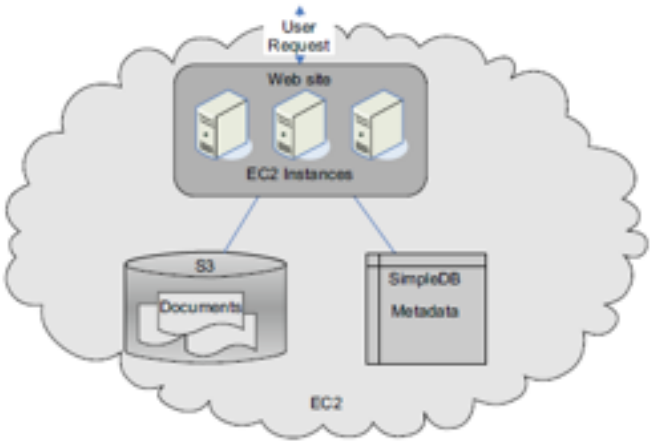
b) You have a file personal_data.txt in the HDFS directory with 100 records. You want to see only the first 5 records from the employee.txt file. How will you do this in Pig? (03 Marks)

Answer:

For getting only 5 records from 100 records we use limit operator.
First load the data in Pig:
personal_data = LOAD "/personal_data.txt" USING PigStorage(',') as (parameter1, Parameter2, …);
Then Limit the data to 5 records:
limit_data = LIMIT personal_data 5;

3.A) Design a book publishing portal by considering usage of following AWS services EC2, S3 and SimpleDB. (07 Marks)

ANSWER:



Student should demonstrate the usage of EC2, S3 and SimpleDB to build book publishing portal

b) Following points are related to Consistency. Provide True / False with justification (03 Marks)

a. Consistency has a same meaning across theorems

b. If a transaction is eventual consistent it implies it is casually consistent

c. In a distributed data system, consistency can be improved by adding more nodes

ANSWER:

Provide True / False with justification

a. Consistency has a same meaning across theorems - False

b. If a transaction is eventual consistent it implies it is casually consistent. - False

c. In a distributed data system, consistency can be improved by adding more nodes - False

4.a) Consider an online shopping site whose transaction workload increases during festive season like Christmas. So for this specific period of time, the resources need spike up. In order to handle this kind of situation what is the preferable choice? Is it cloud elasticity or cloud scalability give the reasons. (04 Marks)

ANSWER:

Ans: the Elasticity refers to the ability of a cloud to automatically expand or compressed the infrastructural resources on a sudden-up and down in the requirement so that the workload can be managed efficiently.

Scalability is commonly used where the persistent deployment of resources is required to handle the workload statically.

Hence we can go for Cloud-Elasticity service rather than Cloud Scalability. As soon as the season goes out, the deployed resources can then be requested for withdrawal.

b) In distributed DBs, such as Amazon DynamoDB and Cassandra, an architectural technique is used to handle temporary failures of nodes that prevent the failed node to receive a replica write request for some time. What is the technique and give a conceptual explanation of how it works with any relevant diagrams to explain data item and node mapping.

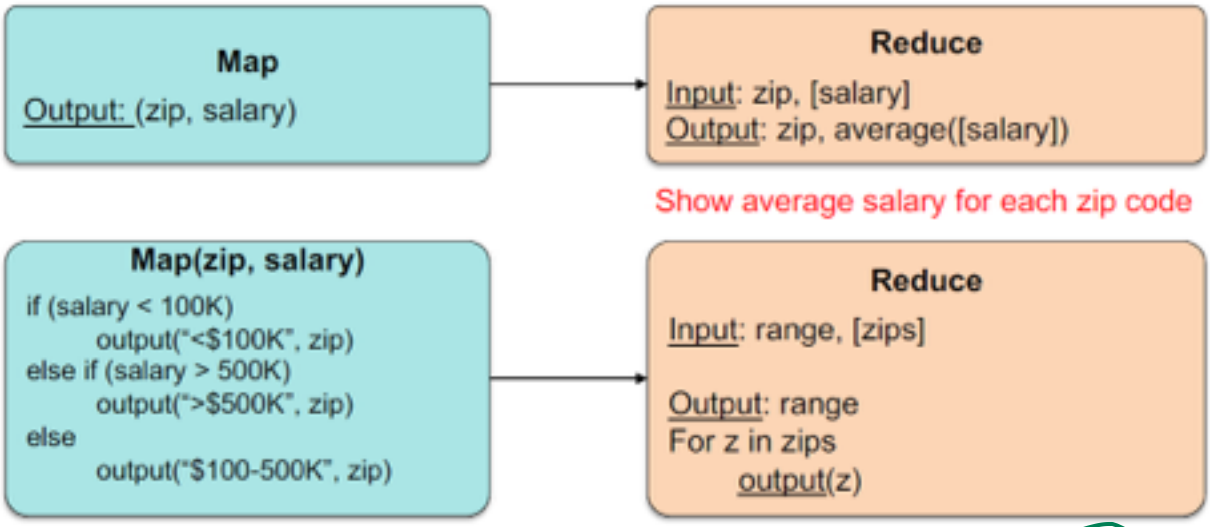
(Marks: 4)

Solution:
The technique is called hinted handoff or a “sloppy quorum” where first N healthy nodes are replica targets rather than N nodes in exact sequence. (1 mark)

All data items and nodes are hashed into a common key space. Given a replication factor, e.g. 3, a data item, e.g. K, is mapped to 3 nodes (e.g. A, B, C) that follow the hashed value of the data item. However when A fails and cannot accept a replica, the data item is stored on the next following node D along with a hint that this item was actually designated for A. This is called a hinted handoff.
As part of group membership protocol, when D realises that A is up again, it forwards the data item to A, thus restoring the originally planned mapping. (3 marks)

5.a) Write the pseudocode of mapper and reducer classes to display zip codes where average salaries are in the ranges: (06 Marks)
(1) < $100K (2) $100K … $500K (3) > $500K.
For the following data sets: {name, age, address, zip, salary}

ANSWER:



b) Create a Hive table by name employees with the following fields and write the Hive query to solve the below requirements: (06 Marks)

i. Insert at least 6 rows into the table.

ii. Find all the employee names beginning with S

EMP_ID	EMP_NAME	DEPT	SAL
1023	Raj	Sales	25,000
1034	Ramesh	Marketing	30,000

6.a) For each of the following scenarios, pick a database that you will use with adequate reasons. Your choices are: MongoDB, Cassandra, Hive, Neo4j (or a similar graph DB). If you have picked a DB for a scenario avoid using it for another one unless there is no other option.

(a) An e-commerce site needs a DB to store all browse and buy interactions for each user. It wants to use the same DB technology but can tweak the configuration for different consistency options for the different types of transactions. E.g. buy and browse transactions can use different instances of the same DB technology with different configurations.
(b) A bank wants to store last 5 years worth of user transaction data in a warehouse for regulatory requirements as well as for historical data analysis.
(c) A system stores weather sensor monitoring data as key-value pairs collected across various regions in the country. The data is consumed into a streaming analysis platform to provide near real-time forecasts of local weather changes.
(d) An IT Network management system needs a DB technology to store the device network information along with various attributes of the devices. Each device can have different types of attributes. The key requirement is to enable root cause trouble shooting using the network data where one device or link failure can impact other devices and links. There can be thousands of devices of different types.

(Marks: 1.5 + 1.5 + 1.5 + 1.5)

Solution:

(a) MongoDB, because browse transactions can be stored fast on instance using eventual consistency configuration (e.g. read=write < majority nodes to survive failures but not requiring majority of nodes to acknowledge) whereas buy transactions can be stored on instance using causal consistency configuration (read=write=majority).
(b) Hive because it provides cost-effective warehouse data storage on Hadoop / HDFS and a SQL like query interface on structured data for regulatory compliance reporting and user data analysis.
(c) Cassandra because it is a geographically scalable key-value store that uses eventual consistency model that is useful for real time systems. Also has integration with Spark etc. for streaming data analysis.
(d) A graph DB like Neo4j will be suitable because each node can be a device or a device component and edges can be device connections (links) or relationships between various devices. Both devices and links can have variable sets of attributes. Root cause analysis of a problem needs graph traversal to navigate edges and nodes to find the problem.

Mention of correct DB name (1 mark) and reasonable explanation (0.5 marks).

b) Spark Core uses RDD as a base data structure. Explain the resiliency aspect of an RDD with specific details of how a computation that is a sequence of transformations and actions, can survive a node / executor crash or what happens when a node runs low on memory.

(Marks: 4)

Solution:

An RDD is partitioned between multiple nodes. Each time a transformation operation is applied on an RDD, a new RDD is created. Depending on a Narrow or a Wide transformation the partition of the new RDD is either created locally on the same node or is a result of data shuffling from multiple partitions across nodes. However, the lineage of a new RDD is stored across all transformations. This is called a DAG.

Only when an action operation is specified, the lineage is used to create the intermediate RDDs and the final non-RDD value. In the event of a node failure, the same DAG can be used to reconstruct the RDD to perform the action and get the value.

A value is always sent back to the driver program. An intermediate result in an RDD can be sent to the driver using collect() operation. (3 marks)

Further, when a node runs out of memory for local RDD partitions, the data can be spilled over onto disk automatically this avoiding out-of-memory issues. (1 mark)

Qtext :

In the following types of application scenarios, point out what is most important - consistency or availability, when a system failure results in a network partition in the backend distributed DB. Explain briefly the reason behind your answer. **(04 Marks)**

(a) A limited quantity offer on a product for 100 items at an online retail store is almost 95% claimed

(b) An online signature campaign on a social media platform is gathering user input

(c) A hotel booking website is trying to sell rooms at a destination that is seeing very little uptake despite offers

(d) An online remote multi-player game needing a lot of sequential team coordination in executing a set of steps to reach a milestone.

User Answer :

Q1.

(a)

For a product with limited quantity, it is very important that the store does not sell the product to the customer when the product has actually been sold out. To ensure this, it is important that the DB is always updated to the latest numbers. Therefore, **consistency** is most important in this scenario since the all the partitions must report the same updated numbers.

(b)

For an online campaign that is collecting user input, the final tally of the scores is not necessary to be updated in real-time. In this scenario, it is more important to have the availability of the platform to the end-user so that they are encouraged to participate in the campaign.
Therefore, **availability** is most important in this scenario.

(c)

Since the hotel booking website is seeing very little uptake, it can be safe to assume that there is sufficient vacancy in the hotel. With this in mind, the priority of the website goes towards user experience towards booking rooms online. Therefore, **availability** will be more important. However, once most of the rooms have been booked, the system must quickly switch to *consistency* to avoid overbooking.

(d)

Since the key factor here is sequential team coordination, and the order of steps is very important, it is very important that the system prioritizes **consistency** over availability. That way the team will be working with the latest information and not have to make decisions based on obsolete data.

Qtext : 2

You have a map-reduce program on a Hadoop cluster. If you run the program on a single node, it takes total 345 se and 5% of the overall time is spent in a sequential reduce operation. The rest of the map reduce application code ar runtime can be parallelized. **(06 Marks)**

(a) If you had a 10 node cluster with similar nodes for the same program and data set, how much time would you theoretically expect the program to take ?

(b) Is there any reason to expect higher time if you actually measured the program execution on the 10 node cluster ?

(c) Suppose you ran this program, with necessary modifications, but with a larger data set trying to accomplish more work with the 10 node cluster. What is a theoretical speed up you could target ?

User Answer :

Q2

T(1) = 345 seconds
f = 0.05

(a)

N = 10
Time taken by 10-node cluster = T(N) = f*T(1) + (1-f)*T(1)/N
T(10) = 0.05*345 + 0.95*345/10 = 17.25 + 32.775 = **50.025 seconds**
(Speedup = 6.896551724)

The program in the 10 node cluster would take 50.025 seconds.

(b)

The time of 50.025 seconds is a theoretical time that assumes fixed workload. Additionally, all system or communication overhead are ignored in thi calculation. And finally, the I/O time or exception handling time is also not included the analysis. By factoring in these times, **it is definitely possible to get a higher time** with a 10 node cluster.

(c)

Using Gustafson's Law:
S(N) = f + (1-f)N

Therefore, S(10) = 0.05 + 0.95*10 = **9.55**

Therefore, theoretical speedup = 9.55

Qtext : 3

It has been noted that there has been a random failure in one of the nodes of a cluster of 4 servers. The failure is observed always on the same node and is happening on the 25th of every month. An hour is taken to recover this node. Incidentally, the applications on these servers need an additional half hour to be started. Consider 30 days in a month.

(06 Marks)

a) Calculate the availability of the cluster.
b) If the cost of downtime is $2k per hour, then what is the quarterly cost?

User Answer :
Q3.

(a)
MTTR = 1 + 0.5 hours = 1.5 hours

Assuming, there are 30 days, the total hours = 30*24 = 720 hours
Therfore, since the node fails consistently on 25th of every month, we can say that the
MTTF = 720 hours

Therefore,
Availability = MTTF / (MTTF+MTTR) = 720 /721.5 = **0.99792 or 99.792%**

(b)
One quarter = 3 months = 90 days = 2160 hours
Downtime cost = $2000 per hour

Quarterly cost = 2000*1.5*2160/720 = **$9000**

Therfore, quarterly cost of downtime = $9000

Close

Qtext : 4

"spotify, an on-demand music providing platform, uses Big Data Analytics, collects data from all its users around the globe, and then uses the analyzed data to give informed music recommendations and suggestions to every individual user".illustrate different phases in the life cycle of this big data project. **(04 Marks)**

User Answer :

Given are the stages:

1. Business case evaluation

Spotify has to define a clear business case towards this project. In this case, it is to use the user information like music choices, follow-up songs, minutes played, etc to make a recommendation system to every individual user.
Justification - a good recommendation system will attract more users and retain existing users since good recommendations lead to more usage
Motivation - to build a large user base and be the most profitable music platform in the industry
Goals - To create a personalized recommendation system for each user based on the information collected from all users in the system

2. Data Identification

Almost all of the data sources will be internal since spotify will be able to get its users usage pattern to extract information needed for the recommendation system.
Here, identifying the relevant inputs is the most important. The inputs could be as follows:
- Minutes listened per song
- Which song was followed up after the current song
- Most listened genre
- Which decade of the songs most listened to?
- Songs liked/disliked

3. Data Acquisition and Filtering

In this phase, the data decided from the previous step is gathered and filtered. A basic data validitiy check is performed to see whether the data is useable or not.

4. Data Extraction

Most of the data here may not be in any strict structural format. This stage deals with converting all the unstructured data into a structured format which can then be passed to the model training.

5. Data Validation and Cleansing

In this stage, the extracted data undergoes a thorough integrity check. The data is checked for outliers, skewness, missing fields, etc.

6. Data Aggregation and Representation

In this stage, all the different structred data from the previous stage is collated into a single unified view. Any key related discrepancies are corrected

4

Goals - To create a personalized recommendation system for each user based on the information collected from all users in the system

2. Data Identification
Almost all of the data sources will be internal since spotify will be able to get its users usage pattern to extract information needed for the recommendation system.
Here, identifying the relevant inputs is the most important. The inputs could be as follows:
- Minutes listened per song
- Which song was followed up after the current song
- Most listened genre
- Which decade of the songs most listened to?
- Songs liked/disliked

3. Data Acquisition and Filtering
In this phase, the data decided from the previous step is gathered and filtered. A basic data validitiy check is performed to see whether the data is useable or not.

4. Data Extraction
Most of the data here may not be in any strict structural format. This stage deals with converting all the unstructured data into a structured format which can then be passed to the model training.

5. Data Validation and Cleansing
In this stage, the extracted data undergoes a thorough integrity check. The data is checked for outliers, skewness, missing fields, etc.

6. Data Aggregation and Representation
In this stage, all the different structred data from the previous stage is collated into a single unified view. Any key related discrepancies are corrected and the data structure is made uniform.

7. Data Analysis
This stage contains the actual analysis that leads the feasible results. Here, a mix of descriptive, diagnostic and predictive analytics takes place in order to determine the users' behaviour and how that can be translated to getting a good score on their recommendation logic.
Once done, the test results are tested against any or all alternate hypotheses that may have been suggested during the course of the project. All the features are explored in order to see which features form the final feature list that define the recommendation system.

8. Data Visualization
In this stage, the results of the previous stage are visualized. These results could in the form of an accuracy index, a correlation plot, or something as simple as a probability matrix of the recommended music tracks based on the a test input.
This stage is important to explain the performance efficiency of the recommendation system to the business, since business does not necessarily have analytical skills to understand the numerical results.

9. Utilization of Analysis Results
In this final stage, the recommendation model is deployed to a small section of the user base to test out the feedback and the performance of their model. Once the results are in, stages 7-9 may be repeated to enhance the performance.
Finally the model is deployed for the entire user base.

Qtext :

Write the pseudocode of mapper and reducer classes to find Call data records which has call type as ‘sms’ and call duration more than 10mins. Consider the datasets with the following fields: subscriber_phone_number, cell_id, timestamp, call_duration, phone_id, status and type_of_call. **(06 Marks)**

User Answer :

Mapper 1 - Extracts call records with type "sms"

mapper1():

```
import sys

for line in sys.stdin:
  // Assume data is comma separated (",")
  data = line.strip().split(",")
  if len(data)!=7:
    continue

  if (data[6] == '\'sms\''):
    for item in data:
      print(item, end='\t')
    print()
```

Reducer 1- Returns records with type "sms" and duration more than 10 mins

reducer1():

```
import sys

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=7:
    continue


  if(int(data[3]>=10):
    for item in data:
      print(item, end='\t')
    print()
```

Qtext : 6

You are given a data set with the following fields in each record: name, age, address, zipcode, salary. You need to output zip codes where avg salary is in the range $100k- $500K using MapReduce. Show the logic of Mapper and Reducer functions to solve the given problem. **(04 Marks)**

User Answer :

Mapper 1 - Extracts a tuple of (zip, salary)

mapper1():

```
import sys

// Assume that the fields are comma separated (",")
for line in sys.stdin:
  data = line.strip().split(",")
  if len(data) == 5:
    // Ouputs (zip, salary)
    print(data[3], "\t", data[4])
```

Reducer 1 - Calculates average salary per zip as a tuple (zip, average(salary))

reducer1():

```
import sys

oldZip = None
salTotal = 0
salCount = 0

// Assume that the fields are tab separated
for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
    // assume something has gone wrong and skip

  zip, sal = data

  if oldZip and oldZip != zip:
    avgSal = salTotal/salCount
```

Reducer 1 - Calculates average salary per zip as a tuple (zip, average(salary))

reducer1():

```
import sys

oldZip = None
salTotal = 0
salCount = 0

// Assume that the fields are tab separated
for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
    // assume something has gone wrong and skip

  zip, sal = data

  if oldZip and oldZip != zip:
    avgSal = salTotal/salCount
    print(oldZip, "\t", avgSal)
    oldZip = zip
    salTotal = 0
    salCount = 0

  oldZip = zip
  salTotal += float(sal)
  salCount += 1

if oldZip != None:
avgSal = salTotal/salCount
print(oldZip, "\t", avgSal)
```

Mapper 2 - Collects those zips whose salary is between 100K-500K

mapper2():

```
import sys:

for line in sys.stdin:
  data = line.strip().split("\t")
  if len(data)!=2:
    continue
```

```
  salCount += 1

 if oldZip != None:
 avgSal = salTotal/salCount
 print(oldZip, "\t", avgSal)
```

Mapper 2 - Collects those zips whose salary is between 100K-500K

mapper2():

```
import sys:

 for line in sys.stdin:
   data = line.strip().split("\t")
   if len(data)!=2:
     continue

  zip, sal = data

  if ((sal>=100000) & (sal <= 500000)):
    print("100K-500K", "\t", zip)
  else:
    print("Others", "\t", zip)
```

Reducer 2 - Print out all the zips with key as "100K-500K"

reducer2():

```
  import sys

  for line in sys.stdin:
    data = line.strip().split("\t")
    if len(data)!=2:
      continue

    flag, zip = data

    if flag == "100K-500K":
      print(zip) // prints the final required zips
```

1. An online retail company's product recommendation system runs in a Hadoop cluster of 100 nodes with a Master node. A cluster node on the average can fail every 150 hours. The Master is configured with a Secondary node that has to be brought up manually, but it takes 15 min using a script. Also, if a failure happens and once Secondary Master node is started, the analysis needs to be restarted and re-initialized, which takes about 45 min. **(04 Marks)**

a) What is the availability of this recommendation system ?

b) If the recommendation system is down then every hour estimated $50,000 is lost as business impact. What is the yearly cost of this downtime ?

ANSWER:

((02Marks + 02 Marks)

MTTF = 150 hours

Availability:
MTTF = 150 hours
MTTR = 8 hours
Availability = 150/ (158) = 94.9%

2. What is your choice for designing a distributed database by considering CAP theorem for the following cases. Justify your design choice in each case. **(04 Marks)**

a) tiger.com is a stock brokerage service company. They have requirement to display most recent price of stocks to its users. Display of older price is not desirable. If the prices are not consistent then some of the triggers may not hit and the user may suffer a serious loss!

b) fromkart.com is an ecommerce company that maintains database of millions of products. At the backend a bulk update of product prices need to be done, these processes will take an hour. Uses of fromkart.com have to wait for all prices to be updated before any read query takes place. This is not desirable as the users of the company will face a downtime resulting in loss of business. Company have to return the partially updated data till the full upload is done.

ANSWER:

(02Marks + 02 Marks)

i. Pick anCP database

ii.Pick an AP database.

3.

i. Suppose there is a file having size of 514MB is stored in a Hadoop (Hadoop 2.x) by using the default size-configuration of 128 MB block and also by default replication-factor. Then, how many blocks will be created in total and what will be the size of each block? **(04 Marks)**

ii. A program can achieve 67% parallelism. Assuming a fixed workload, draw a speedup graph considering different number of processors – 4; 16; 64; 256 What is the maximum theoretical speedup this program can ever achieve? **(04 Marks)**

ANSWER:

i.

The default replication factor is 3 and the default block-size is 128MB in Hadoop 2.x.
Thus, the 514MB of file can be split into:

A	B	C	D	E
128MB	128MB	128MB	128MB	2MB

•The default block size is 128MB
•Number of blocks: 514MB128MB=4.05 ≈5 blocks
•Replication factor: 3
•Total blocks: 5*3=15
•Total size: 514*3=1542MB

ii.

$S(p) = 1/ (f + (1-f)/p)$

No. of Processors	4	16	64	256
Speed Up	2.01005	2.689076	2.937127	3.006459

4. In each of the following scenarios, point out and give a brief reason what type of multi-processor computer one would use. The choices are (SIMD, SISD, MIMD, MISD).
(04 Marks)

(a) A scientific computing application does a sin(a) + tan(a) transformation for every variable a.

(b) An image processing application where multiple images are merged at corresponding pixel level with a certain operator F to create the result image. E.g. image3 = F(image1(i, j), image2(i,j), ...) for all pixels (i,j) in the input images.

(c) A Hadoop system for data analysis

(d) A development system with application code and a MongoDB running on a single multi-core laptop.

ANSWER:

(a)MISD (b) SIMD (c) MIMD (d) MIMD

(1+1+1+1 marks)

5. Figures (a) and (b) show two independent sequences of reads and writes for data items in a distributed DB across various application nodes P1, P2, P3, P4 on a time line. “R: A=V” at P1 means read operation on data item A returns value V at application node P1 and “W: B=V” at P2 means data item B is written with value V by application node P2. In each case, do you think the sequence demonstrates strict consistency or eventual consistency ? Give reasons. **(04 Marks)**

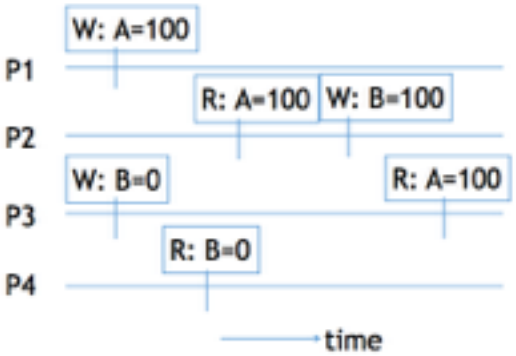
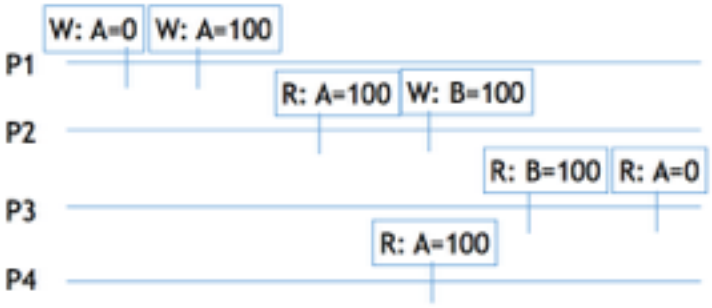


Fig (a)



Fig(b)

ANSWER:

(a)Strict (b) Eventual

(02Marks+ 02 Marks)

6. An online book retailer site recommends book titles by their rating. The historical data from book title reviews is stored in a file as show below where a book title can be reviewed by many users and a user can review many book titles. **(06 Marks)**

user	title	rating
U1	T1	2
U3	T2	2
U2	T1	3
U1	T2	4

...

The recommendation system using MapReduce programming creates Top 5 books by their max rating.

(a) Show the pseudo-code logic of the mapper
(b) Show the sample output format of mapper
(c) Show the pseudo-code logic of the reducer
(d) Show the output of shuffle sort that is fed to the reducer

Answer:

(2 + 1 + 2 + 1 marks)

Answer:
(a)

```
class Mapper {

    TreeMap topN; // stores a sorted list of topN rating records
for each key

    <key, value> map (user, record) {

        value = record[rating]
        key = record[title] + '_' + record[category]
        topN.put(value, key)
        if topN.size() > 5:
            topN.remove(topN.firstKey()
    }
    at the end of each Mapper instance {
        for each entry <k,v>  in topN:
            emit(k, v)
    }
}
```

(b)

```
map 1:   map2:    …
<T1, 3>  <T2, 4>
<T1, 2>  <T2, 2>
…        …
```

(c)

```
class Reducer {

    TreeMap topN; // stores a sorted list of topN rating records
for each key

    <key, value> reduce (user, record) {

        value = record[rating]
        key = record[title] + '_' + record[category]
        topN.put(value, key)
        if topN.size() > 5:
            topN.remove(topN.firstKey()
    }
    at the end of each Mapper instance {
        for each entry <k,v>  in topN:
            emit(k, v)
    }
}
```

(d)

```
<T1, [3, 2, ..]>
<T2, [4, 2, ..]>
…
```