NATIONAL UNIVERSITY OF SINGAPORE

|  |
| --- |
| SCHOOL OF COMPUTING  ASSESSMENT FOR  Semester 1 AY2021/2022  CS5425/CS4225 – Big Data Systems for Data Science  Mock Test Paper  October 2021 Time Allowed: 1.5 Hours |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**INSTRUCTIONS TO STUDENTS:**

1. This assessment paper contains **FOUR (4)** questions and comprises **FOURTEEN (14)** printed pages, including this page.
2. Students are required to answer ALL the questions.
3. Write your answers within the space provided. Answers written on other parts of the answer script will not be graded unless you specify explicitly.
4. This is an **OPEN BOOK** examination, allowing any materials except discussion with others.
5. Please circle your class and write your matriculation number below.

**Class: CS5425 CS4225**

**Matriculation Number: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

This portion is for examiner’s use only

|  |  |  |
| --- | --- | --- |
| **Question** | **Marks** | **Remarks** |
| **Q1 [9 marks]** |  |  |
| **Q2 [12 marks]** |  |  |
| **Q3 [9 marks]** |  |  |
| **Q4 [10 marks]** |  |  |
| **Total [40 marks]** |  |  |

**QUESTION 1: Indicate whether the following statements are true or false. Each question is worth 1 mark. [9 marks]**

|  |  |
| --- | --- |
| Questions | Your Answer (True/False) |
| One of the key big data challenges is velocity, which means uncertainty of the data. | False |
| The RDD of Spark can also be stored in the disk. | True |
| Since DRAM at all levels in data centers has a much better performance than hard disk in terms of both latency and bandwidth, we should always put our data into DRAM. | False |
| The sample graph shown in Figure 1 has at least one dead end. | True |
| The sample graph shown in Figure 1 has at least one spider trap. | True |
| In a data center, the latency of Rack DRAM is higher than that of Local DRAM, due to the network latency. | True |
| The reliability of RDD in Spark is implemented with the same mechanism as Hadoop. | False |
| As described in lecture, min hashing is used to find candidate similar pairs, that we can further test to find which are actually similar pairs. | True |
| As described in lecture, the reservoir sampling algorithm only works if the total number of elements in the data stream in known in advance. | False |

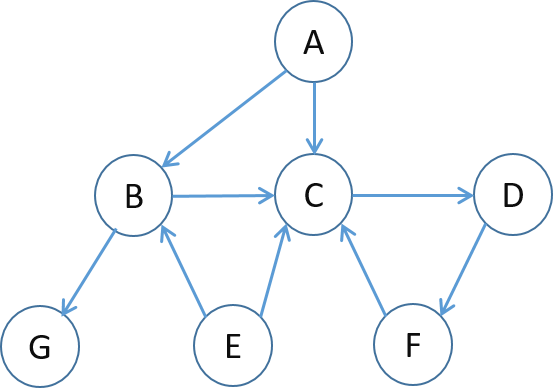
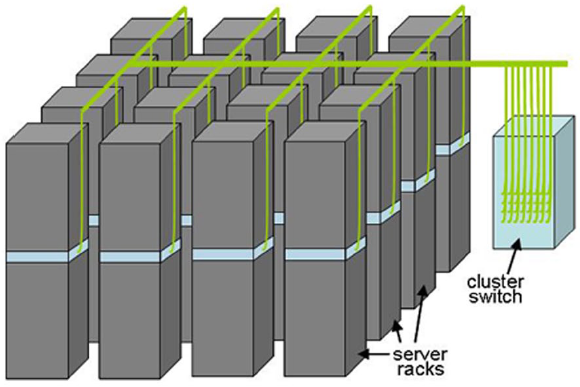


Figure 1. A sample graph for Question Q1.

**QUESTION 2: Big Data in Data Center [12 marks]**

The architecture of a commercial data center is illustrated in the below figure. The number on the top of each rack is the identifier of each rack. Users can run Hadoop or Spark jobs in the data center. Answer the following questions.



1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

Server A

(a) Suppose a program *P* is running on Server *A* in Rack 1. You find that the latency of *P* accessing the local disk of the server is similar to that of accessing the disk in other servers in the same rack. Why is this the case?

[2 marks]

Answer: the network in the Rack is relatively fast, and the disk performance becomes the bottleneck.

(b) Hadoop and HDFS are designed to work well together. What is the potential advantage for Hadoop task scheduling if HDFS exposes the locations of block replica to Hadoop?

[2 marks]

Answer:  
The MapReduce scheduler can arrange for Map tasks to execute on the same node that stores the data, avoiding a copy across the network.

(c) In HDFS, each chunk is replicated three times by default. Explain one strong point and one weak point compared to the Hadoop/HDFS system, if we replicate each chunk four times rather than three times.

[2 marks]

Answer:  
(good) system will be more robust to node failure  
(bad) storage overhead and slow write performance.

(d) Suppose that Hadoop schedules a Map task to execute on Server A. However, the execution time of that Map task is over twice the average time of Map tasks on other servers for the same job. Describe two potential reasons for this phenomenon.

[2 marks]

Answer:  
(a) the Map task requires processing a larger amount of data.

(b) the Map input is stored in a remote machine, increasing network I/O.  
(c) the machine is busy.

(e) Suppose the following Spark program will be run in the cluster. This program basically identifies the lines with “INFO” from the execution log generated from the previous Hadoop and Spark job executions. The log is huge in the tens of terabytes. Amend the program by only adding 1 code line to improve the efficiency of the program.

Line 1: lines = spark.textFile(“hdfs://log”)

Line 2: errors = lines.filter(lambda s: s.startswith(“INFO”))

Line 3: info = errors.map(lambda s: s.split(“\t”)[2])

Line 4: info.filter(lambda s: “hadoop” in s).count()

Line 5: info.filter(lambda s: “spark” in s).count()

[2 marks]

Answer:

Add info.cache() between Line 3 and Line 4. This will avoid repetitive work for computing “info” when running Lines 4 and 5.

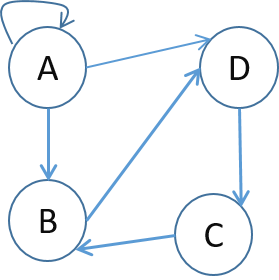
(f) In GFS/HDFS, the chunk size is set to 64MB by default. Explain the problems if we set this chunk size to be 1) much smaller (say, 4KB), or 2) much bigger (say, 1GB).

[2 marks]

Answer:  
Too small: high overhead OR large amount of disk and network I/O  
Too big: parallelism will degrade OR Cannot fit into memory.

**QUESTION 3: Calculations [9 marks]**

(a) Set up the PageRank equations for the below graph, assuming β = 0.85 (teleport probability = 1−β). Denote the PageRank of node *x* by *r(x)*.



[3 marks]

Answer:

r(A)=0.85\*(r(A)/3)+0.15/4

r(B)=0.85\*(r(A)/3+r(C))+0.15/4

r(C)=0.85\*(r(D))+0.15/4

r(D)=0.85\*(r(A)/3+r(B))+0.15/4

[Writing as the equivalent matrix equation is also fine. Please refer to the tutorial.]

(b) Compute the Jaccard similarity of the sets of 2-shingles for the following two text documents:

Document 1: “cat and dog”

Document 2: “the cat and mouse”.

[3 marks]

Answer:  
Set of 2-shingles of document 1={cat and, and dog}  
Set of 2-shingles of document 2 ={the cat, cat and, and mouse}

Intersection = {cat and}

Union = {the cat, cat and, and mouse, and dog}

Jaccard similarity = size of intersection / size of union = 1/4.

(c) Assume the following dataset of data points is given: (2,2), (4,4), (6,6), (10,10), (0,4), (4,0). k-means is run with k=3 to cluster the dataset. Euclidean distance is used as the distance function to compute distances between centroids (i.e. cluster centers) and points in the dataset, i.e., . Suppose that at some iteration, just before updating the centroids (i.e. just before the Update step), the cluster assignments C1, C2 and C3 of k-means are as follows:

C1: {(2,2), (4,4)}

C2: {(0,4), (4,0)}

C3: {(6,6), (10,10)}

If we run k-means to completion, what are the final cluster assignments and what are their centroids?

[3 marks]

Answer:

Assignments 🡪 Centroids:

C1: {(4,4)} 🡪(4, 4)

C2: {(0,4), (4,0), (2,2)}🡪(2,2)

C3: {(6,6), (10,10)} 🡪(8, 8)

*Note*: the question doesn’t ask for the step by step process, so showing the final output above is sufficient, but in case you want to check your steps, here are the steps:

Update step: centroids are set to (3,3), (2,2), (8,8)

Assignment step: point (2,2) is moved to cluster C2

Update step: centroids are set to (4,4), (2,2), (8,8). Then the algorithm converges.

**QUESTION 4: Analysis and Problem Solving [10 marks]**

(a) Jim is using the k-means algorithm in Hadoop to cluster a set of points. In his implementation, each iteration of k-means is implemented by one Hadoop job. Taking a point as input, the mapper outputs the point’s nearest cluster number as the key and the point itself as the value. The reducer computes the new centroid for each cluster, which is emitted as the value with the key being the cluster number.

1. Jim runs the Hadoop program in a single server. He finds that the performance is much lower than a k-means implementation written in Java from scratch. Explain why.

[2 marks]

A) Hadoop is designed for distributed execution. As such, it has additional overhead (e.g. related to scheduling, and parallelism-oriented designs).

OR:

B) Hadoop needs to repeatedly write to HDFS, leading to additional disk I/O.

1. Suppose that Jim runs the tasks on 32 servers. He finds that the time taken by the program does not significantly improve. Explain one possible reason why.

[2 marks]

A) Some clusters can be huge (e.g. containing the majority of the data points), which results in limited benefits of parallelism during the reduce stage (since the points in each cluster are only handled by 1 worker).

OR:

B) The bottleneck may be the latency of reading and writing from HDFS (e.g. consider the case where the data size is small). As such, increasing the number of workers does not improve efficiency.

(b) A programmer wants to develop a system to obtain the popular words from microblogs (e.g. tweets). The microblogs are generated and collected continuously at the speed of around 10,000 microblogs per second, each with 100 characters on average, and we want the results to be refreshed every second. Which framework (or type of framework) discussed in lecture is suitable for this situation?

[2 marks]

Answer: This task is a streaming task (since we need to be able to update our results upon receiving new data), so stream data processing systems (in particular, Storm as discussed in lecture) should be used.

(c) Suppose you have a large number of documents, and have stored k-grams (or k- shingles) of these documents. The shingles are stored one by one in the form of <*docID*, *kgram*>, where *kgram* is the string of the k-shingle, and *docID* is the document ID that the *kgram* belongs to. Show the pseudo code for how you would use MapReduce to compute a Min-Hash value for each of your documents, using a single hash function (which maps strings to integers); namely hash(x). You must follow the pseudo code of the below format.

map (docID, kgram) {  
/\* your pseudo code\*/  
/\* you need to output the map results by calling the API, emitIntermediate(specify your map output)\*/  
}

reduce (/\* specify your input to reducer \*/) {  
/\* your pseudo code \*/  
/\* you need to output the map results by calling the API, emitFinal(docID, minHash). Here, the key should be document ID and the value should be its Min-Hash value \*/  
}

(Note: there are multiple ways to solve this. You can choose to do more of the work in the map function or in the reduce function; both ways are acceptable).

[4 marks]

Answer:

The map function computes the hash value:

map(docID, kgram){

emitIntermediate(docID, hash(kgram));

}

The reduce function finds the minimum hash value for each document ID:  
  
reduce(docID, hashes[]){

minhash := INFINITY;

for each h in hashes {

if h < minhash {

minhash := h

}

}  
emitFinal(docID, minhash)

}

Note: replacing the entire reduce function with something like

reduce(docID, hashes[]){

emitFinal(docID, min(hashes))

}

is also acceptable. As long as what your code is doing is clear, and its steps can plausibly be considered as ‘single steps’, the code is acceptable.

It is also fine to do the hashing in the reduce function, in which case the map function can simply do nothing (i.e. emit its input tuples as output).

BLANK PAGE

END OF ASSESSMENT