Question 1 - Hadoop [25 Points]

a) In at most one sentence, what is HDFS? [2 points]

Hadoop Distributed file system, used to store and access large files in a distributed manner.

[2 points] If it in anyways mentions a distributed file system or that it stores large files in a distributed fashion.

b) What mechanism does Hadoop use to achieve fault tolerance? [4 points]

Replication of data/data blocks

[4 points] For mentioning that it replicates data.

c) What is the difference between a NameNode and a DataNode in HDFS? [4 points]

Name node is a master server that manages the filesystem namespace, tracks metadata, and regulates client access to files.

Data node is usually one per node in a cluster that manages storage attached to their node. Serves read/write requests, file creation/deletion, and replication.

[2 points] For a valid description of what a NameNode is/does
[2 points] For a value description of what a DataNode is/does

d) Name three possible common failures that HDFS must handle. [6 points]

DataNode failure
NameNode failure
Network partition
Rack Failure

[2 points] Per correct answer, max of 6 points

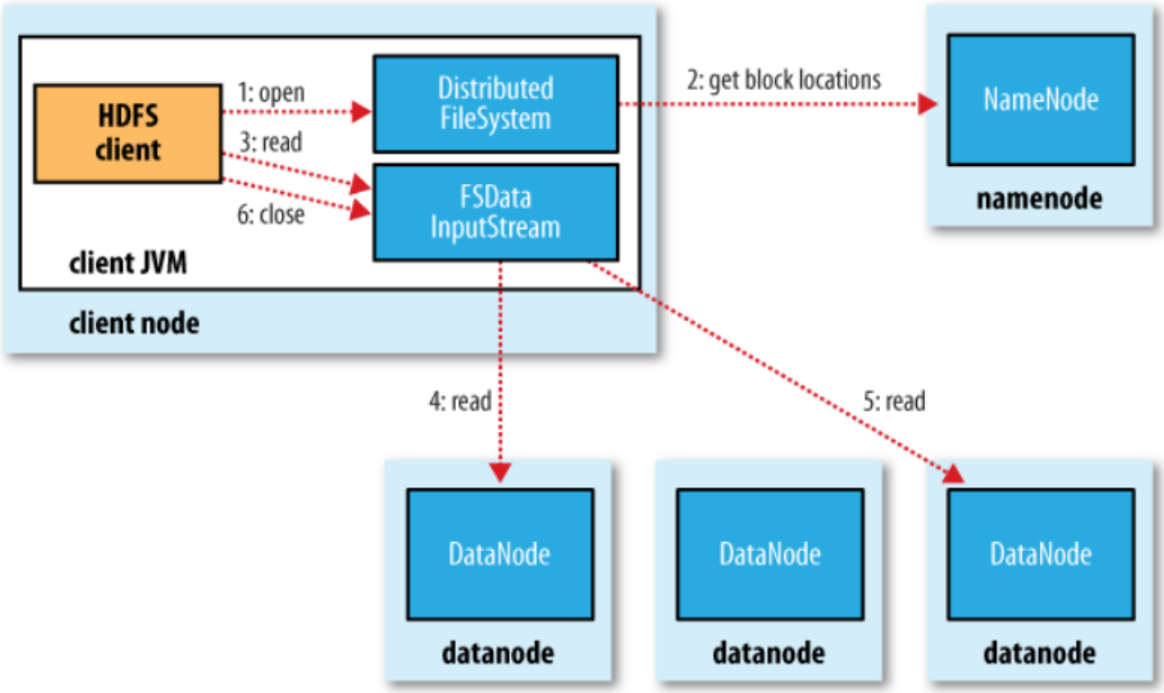
e) Lists two functions of JobTracker and TaskTracker in HDFS. [4 points]

Job tracker :
Scheduler service in the Hadoop system
Manages Resources

Task tracker:
Accepts tasks (Map, Reduce, Shuffle, etc) from JobTracker
Informed the JobTracker with the task status

[1 point] For each correct job (max 2 per JobTracker and TaskTracker)

f) Draw a diagram of a client reading data from a HDFS. [5 points]



[2 points] Diagram includes a client, a NameNode, and at least one DataNode
[1 point] Includes an arrow from HDFS client to DFS
[1 point] Includes an arrow from DFS to NameNode to get block locations
[1 point] Includes an arrow from DFS to at least one DataNode for the read

Question 2 - MapReduce Concepts [25 Points]

a) What is the primary data type that MapReduce applications deal with? [4 points]

key-value pairs

[4 points] Correct answer

b) What is the property of the data discussed in class that allows MapReduce to work on the data in parallel without worrying about synchronization or dependencies? [4 points]

WORM (write once, read many), the data is READ ONLY

[4 points] Correct answer

c) In a MapReduce application, what do we mean by intermediate data? [4 points]

The key-value pairs that are output by the mappers while processing the input data.

[4 points] Correct answer

d) Explain why intermediate data affects the performance of a MapReduce application in terms of I/O costs? In terms of communication costs? [5 points]

After mapping, or during spills, the intermediate data must be written to disk. More intermediate data = more data written.
After mapping, the key value pairs are sent to reducers, so more intermediate data = more data communicated.

[2 points] For a reason WHY more intermediate data means more I/O (it is written to disk during a spill or after mapping)
[2 points] For a reason WHY more intermediate data means more network cost (it is sent across the network to reducers after mapping)
[1 point] If both are correct

e) What is a "spill" in the context of MapReduce? [4 points]

A spill is when we run out of memory for the intermediate data a mapper has generated so we must write it to disk.

[4 points] Correct answer

f) Name one benefit we see in Mappers by making block size larger. [4 points]
Name one benefit we see in Mappers by making block size smaller.

Larger block size means less relative mapper overheads.
Smaller block size means we are less likely to overflow memory/spill.

[2 points] Understanding that large block size = low overhead
[2 points] Understanding that small block size = fewer spills/memory concerns

Question 3 - MapReduce Application [25 Points]

For the following question, consider the k-mer counting MapReduce application from class. Given an input sequence of DNA, containing the characters A,C,G,T, k-mer counting involves counting the number of occurrences of each sequence of length k that occurs in the string. For example, if k = 3 the input string is ACTTC contains one instance of ACT, one instance of CTT, and one instance of TTC.

a) For k = 5, if we assume that every possible sequence of length 5 occurs in our input string, then what is the size of our keyspace. For subsequent parts of this question, you may refer to this quantity as **K**, as needed. [2 points]

K = 1024

[2 points] Correct answer

b) For a given input DNA string containing 1 billion characters, and k = 5, how many total key value pairs will be output by the mappers? [2 points]
(Hint: Consider how many key-value pairs each character will be a part of)

1 billion - 4, or ~1 billion
(we will also accept ~5 billion due to miscommunication on Piazza)

[2 points] Correct answer

c) Considering the answer to the previous two questions, if we decide to add in-mapper aggregation to reduce the amount of intermediate data, what is the maximum number of key-value pairs that will be output **by each mapper** (assuming that the mappers only output key-value pairs after processing their input in its entirety). [4 points]

K (or 1024 or whatever answer they put for a)

[4 points] Correct answer

d) If we now run our application in a cloud environment with 32 sites and 16 mappers per site, how many total key-value pairs are involved in the shuffle-sort communication? [4 points]

1024 * 32 * 16 = 524,288

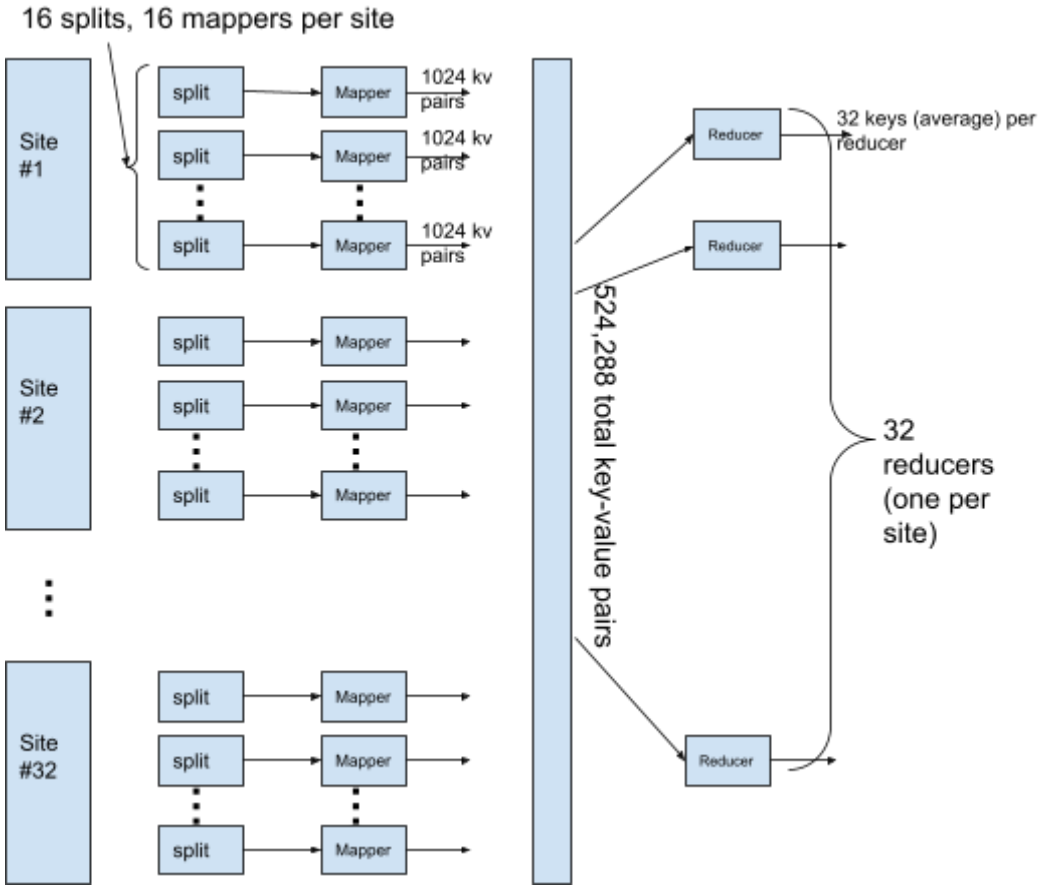
[4 points] Correct answer (or correct in terms of answer to a)

e) Assume we have 1 reducer per site. How many unique **keys** (not key-value pairs) will each reducer be responsible for reducing? [4 points]

1024 / 32 = 32

[4 points] Correct answer (or correct in terms of answer to a)

f) Draw a data flow diagram as shown in class. In your diagram, label the number of sites, number of mappers, number of reducers, and your quantities computed in parts c, d, e. [9 points]



[3 points] For structure of the diagram. Award three and deduct a point if:

- **There is not one split of data per mapper**
- **The barrier does not exist or does not cover all of the mappers**
- **Combiners are used (the problem specified that we did in-mapper aggregation instead)**

[1 point] For correctly labeling number of sites
[1 point] For correctly labeling number of mappers per site (or total number of mappers)
[1 point] For correctly labeling number of reducers
[1 point] For labeling at least one of the arrows leaving a mapper with the quantity they computed in C
[1 point] For labeling the barrier in some way stating the total number of kv pairs being shuffled is their quantity from D
[1 point] For labeling the number of keys each reducer handles (as they computed in part E)

Scrap Paper