Name:

CSCI/ECEN 5673: Distributed Systems
Spring 2022
<u>Final Exam</u>
05/04/2022: 4:30 – 6:00 PM

- This is a 1 hour 30 minutes, closed book, closed notes exam.
- There are seven questions. Answer all questions.
- Be concise and to-the-point in your answers. Answer all questions in the space provided.

Name:

Name:

1. **[20 Points]** Choose one correct answer for each of the following:
(+2 for correct answer, -1 for incorrect answer, 0 for no answer).

a. A gossip protocol
(i) doesn't guarantee that all non-faulty nodes that are connected will receive a multicast messages.
(ii) requires an overlay network to be built before multicast can begin.
(iii) uses either positive-ack or negative-ack technique to recover from message losses.
(iv) elects a leader to coordinate the dissemination of a multicast message.

b. Resilient Distributed Datasets
(i) can be modified only through coarse-grained operations called transformations.
(ii) can be partitioned across machines based on a key in each record.
(iii) do not allow parallel operations on them.
(iv) (i) and (ii) but not (iii).

c. Which of the following is true for early peer-to-peer networks?
(i) One limitation of Gnutella is that it relies on a central index server.
(ii) KaZaA architecture is based on a two-tier node hierarchy, super nodes and ordinary nodes.
(iii) Search function in Napster is based on flooding a query with a TTL.
(iv) None of the above are correct.

d. In Zookeeper,
(i) it is possible for a client write request to be executed more than once.
(ii) it is possible for a read operation to return stale data.
(iii) clients can get notified of a change to the data they are watching.
(iv) all of the above.

e. Chubby lock service
(i) is designed for managing small amount of critical data such as system configuration, state information, etc.
(ii) uses Paxos for electing master in a Chubby cell.
(iii) uses a hierarchical file namespace where locks are specified via directories and files are used to store data.
(iv) (i) and (ii) but not (iii).

f. Cassandra design is based BigTable as follows:
(i) Nodes communicate with each other using a Gossip protocol.
(ii) Data is stored across nodes using a Distributed Hash Table.
(iii) Data is written to an in-memory structure and then to disk once the memory structure is full.
(iv) All of the above.

Name:

g. BigTable
(i) supports transactions, which can be used to perform atomic read-modify-write sequences on data in the table.
(ii) uses DHTs to locate data based on a row key.
(iii) supports nested column families.
(iv) None of the above.

h. Chubby is used in BigTable to
(i) store Bigtable schema information such as column family information of each table.
(ii) store access control lists.
(iii) discover tablet servers.
(iv) All of the above.

i. In FreeNet,
(i) the total bandwidth usage to transfer a file is same as in Bit Torrent.
(ii) a key is stored at a node that is closest to it in the entire network.
(iii) attackers can insert junk files under existing keys.
(iv) new files are cached on nodes that have already stored files with similar keys.

j. Small-World networks exhibit following characteristics.
(i) Shortest path may not be easy to find.
(ii) Average shortest path is logarithmically small.
(iii) To route to a peer, simply route to the closest peer at each step
(iv) All of the above.

Name:

2. Given a log file in HDFS, consider the following Spark code to compute the number of segmentation fault error messages in foo and bar:

```
lines = spark.textfile(“hdfs://namenodedns:port/user/hdfs/folder/logfile”)
errors = lines.filter(_.startsWith(“ERROR”))
messages = errors.map(_.split(‘\t’)(2))
cachedMsgs = messages.cache()
fooMsgs = cachedMsgs.filter(_.contains(“foo”))
barMsgs = cachedMsgs.filter(_.contains(“bar”))
segFaultBar = barMsgs.filter(_.contains(“segmentation fault”))
segFaultBar.count
fooMsgs.filter(_.contains(“segmentation fault)).count
```

(a) **[2 Points]** Write down the names of all RDDs created in this code.

(b) **[2 Points]** What RDDs exist at the end of this code segment?

(c) **[2 Points]** Suppose segFaultBar is lost due to a failure. How can you recover it?

(d) **[4 Points]** Instead of Spark, if you use MapReduce, show the sequence of MapReduce tasks you will use.
Note: You do not need to provide code for Map or Reduce functions, just a brief one-line description is sufficient.

(e) **[3 Points]** What are the sources of inefficiency in your MapReduce solution that will make it run slower than the given Spark solution.

Name:

3. In an Amazon's Dynamo deployment, each key is stored in three nodes, uses a 32-bit hashing mechanism, and uses R = 2 and W = 2 as its read, write quorum values respectively. Physical nodes 1, 2, 3, 4, 5 and 6 are mapped to virtual nodes 10000, 18000, 34000, 50000, 58000 and 70000 respectively, and are successive nodes in the Dynamo ring. Each node stores all its (key, value) pairs consecutively in a sorted (hashed) key order in three different GFS like file system with chunk size 1000 (key, value) pairs. For each physical node *a*, file f_a stores all (key, value) pairs for which *a* is the coordinator node; and files a_y and a_z store the (key, value) pairs whose coordinator nodes are nodes *y* and *z*. At present, the following key and their corresponding values are stored in the Dynamo system:

Hashed Key Range	**Number of keys stored**
0 - 2000	300
2001 - 10000	3900
10001 – 12000	800
12001 – 14000	1200
14001 – 16000	1000
16001 – 18000	300
18001 – 34000	3400
34001 – 40000	600
40001 – 42000	800
42001 – 50000	3000
50001 – 58000	4000
58001 – 70000	6000
70001 – 80000	4000
80001 – 90000	2000

a. **[1 Points]** At what physical node(s) a key k with hash value 10368 would be stored?

b. **[2 Points]** How many (key, value) pairs node 4 is storing in each of its files at present? Explain your answer.

c. **[3 Points]** A user wants to retrieve the value corresponding to a key k' with hash value 34000. Provide the step-by-step sequence of actions that would be performed to retrieve this value.

Name:

d. **[3 Points]** A user wants to store the (key, value) pair (k'', v'') where the hash value of k'' is 18492 and the hash value of v'' is 50621. Provide the step-by-step sequence of actions that would be performed to store this (key, value) pair.

e. Suppose node 4 has fallen behind in updates due to an overload and the goal is to update its state to the latest state. Assume all other nodes are completely upto-date.

(i) **[1 Points]** Which nodes would node 4 exchange messages with? Explain your answer.

(ii) **[3 Points]** Suppose node 4's state is upto-date after all, how many messages would be exchanged and what would be the payload size of each of these messages? Explain your answer.

(iii) **[6 Points]** Suppose node 4's state is behind in the following hashed key values: 10282, 14289 and 40728. How many messages would be exchanged and what would be the payload size of each of these messages? Explain your answer.

Name:

4. Google file system is used to store files F1 and F2. Every chunk of F1 is replicated at servers S1, S2 and S3 (S2 is primary), and every chunk for F2 is replicated at servers S2, S3 and S4 (S3 is primary). Initially, contents of file F1 are 'P' and contents of fille F2 are 'Q'. Two clients C1 and C2 are running concurrently and execute the following sequence of operations:

C1	C2
C1(1): read 20K bytes from F2	C2(1): append 'X' to F2
C1(2): append 'A' to F1	C2(2): append 'Y' to F2
C1(3): read 5K bytes from F1	C2(3): append 'Z' to F1

(a) **[5 Points]** For each of the following operations, provide a sequence of node interactions that will take place.
Use the following notation: if C sends a message to S_b and then S_c sends a message S_d, denote that by C $\rightarrow S_b$, $S_c \rightarrow S_d$; if C sends a message to S_b, and in parallel, S_c sends a message S_d, denote that by {C $\rightarrow S_b$, $S_c \rightarrow S_d$}. You do not need to specify the content of the message.

C1(1):
C1(2):
C1(3):
C2(1):
C2(2):
C2(3):

(b) **[2 Points]** After the six operations have been executed, how many different snapshots of files F1 and F2 are possible? Explain your answer.

(c) **[2 Points]** For the following four snapshots of F1 and F2, answer YES or NO to indicate whether that snapshot is possible or not after the six operations have been executed

Snapshot 1: F1: PAZ	F2: QYX	YES	NO
Snapshot 2: F1: PZA	F2: QXY	YES	NO
Snapshot 3: F1: PAAZZ	F2: QXXY	YES	NO
Snapshot 4: F1: PAZZ	F2: QXYXY	YES	NO

(d) **[4 Points]** Suppose server S2 crashes. Provide a sequence of actions that will take place as a result of this crash.

Name:

5. A chord system uses a 9-bit hash function, where the hash function is a simple mod function (hash (x) = x mod 512). Let there be ten nodes with ids 306, 350, 592, 640, 768, 924, 1024, 1212, 1408 and 1576.

(a) **[3 Points]** Draw a ring and mark the locations of these nodes in the ring.

(b) **[3 Points]** Which node will the following <key, value> pair be stored?

(i) <420, 100>: __________
(ii) <960, 32>: ___________
(iii) <2589, 76>: ____________

(c) **[6 Points]** Use the table below to fill in the finger table for node 1024. Assume clockwise data assignment.

start	**interval**	**Succ.**

(d) **[2 Points]** A client sends a request to node 1024 to retrieve the value associated with key 2589. What path would this request take?

Name:

6. In a BitTorrent network, suppose there are 30 peers; peer ids are 0, 1, …, 29. Peers 0, …, 7 have high bandwidth connections, while other peers have low bandwidth connections. A file F is being shared in this network and is partitioned into 16 chunks, F_0, F_1, …, F_{15}. Peer 4 is the tracker and peer 6 is a web index server. At present, the distribution of this file is as follows:

   Peer 0 has chunks F_0, F_1, ..., F_6.
   Peer 1 has chunks F_4, …, F_{10}.
   Peer 2 has chunks F_5, …, F_{12}.
   Peer 3 has chunks F_0, … F_4 and F_{10}, … F_{15}.
   Peer 5 has chunks F_0, …, F_6 and F_{11}, …, F_{15}.
   Peer 7 has chunks F_3, … F_9 and F_{13}, …, F_{15}.
   Peers 8, 9 and 10 have chunks F_0, …, F_8.
   Peers 14, 17, 20 and 21 have chunks F_3, … F_{12}.
   Peers 22, 24 and 29 have chunks F_0, … F_5 and F_8, ..., F_{15}.

   (a) **[2 Points]** Suppose all the peers that have some file chunks are interested in downloading the remaining chunks. What file exchange pattern would you expect to observe among the peers in this system?

   (b) Suppose peers 3, 5, 7 and 8 are connected with one another and exchanging chunks among them.
   (i) **[2 Points]** What chunk would peer 8 download from peer 3?

   (ii) **[2 Points]** What chunk would peer 7 upload to peer 5?

   (b) Peer 15 now wants to download the file.

   (i) **[3 Points]** Provide the sequence of step-by-step messages that would be exchanged before it can start downloading chunks.

   (ii) **[2 Points]** Suppose it gets connected to peers 0, 8, 17 and 24. What chunk would it download first and from which peer?

   (iv) **[2 points]** What prevents peer 15 from only downloading chunks but never uploading?

Name:

7. Consider two photo storage systems, one using NFS with one file for each photo in a directory structure that is up to 8 nodes deep and the other using Facebook's Haystack. Both systems use CDNs and caching.

   (a) **[4 Points]** Show the sequence of actions that will take place when a client sends a request to download a photo in NFS-based storage.

   (b) **[4 Points]** Show the sequence of actions that will take place when a client sends a request to download a photo in Haystack storage.