CSE530 Distributed Systems: Concepts and Design
Mid-semester Exam (Winter 2024) - February 28, 2024
Max. Time: (1 hours 30 minutes) Max. Marks: 50

VERY VERY IMPORTANT NOTES:

1. Please ensure that the question paper consists of **X printed sides [Y pages].**
2. Calculator is allowed but Sharing of calculators is **NOT ALLOWED**.
3. For every question or part of a question, you are required to enter **ONLY** the **FINAL** answer in the space provided immediately after the question.
    a. For numerical type answers, just enter the final number without any units. The units will be mentioned in the question. For example, if your final answer is 30 milliseconds, please enter only **30** in the answer field for that question.
    b. For multiple choice questions, you may have more than one option correct. Please select/write all the correct answers from the options provided for that question. There is no partial marking for partially correct answers.
4. Google Form Link will be made available at the end of the exam. You will have 15 minutes to input all your answers in the google form.
5. **Google form answers will be taken as the FINAL ANSWERS.** The answers which you write in this question paper will just be a backup.
6. Please do not open your laptop without the permission of the invigilators.
7. Once you have submitted the google form, please stay seated in your place. **DO NOT LEAVE** the exam hall without handing over this question paper (filled with all your answers) to the invigilators.
8. Before starting to answer any question, please read it completely.
9. Exam is fully closed-book. No cheat sheet etc. is allowed.

QUESTION 1: Consider the following scenario for **Network Time Protocol (NTP)**: Client sends a request for the current time to server at $\mathbf{T_1}$=1150 make the client conclude that the server's clock is 50 seconds ahead of itself (the client). The modified response of the server is received by the client at **T_4**=1300 seconds (local time at client). Assume that there is no additional delay due to interception and processing by the attacker. Also, assume that the network delay from client to server (**$D_{c\text{-}s}$**) is one-fourth times the network delay from server to client (**$D_{s\text{-}c}$**) i.e. **$D_{s\text{-}c}$ = 4 * $D_{c\text{-}s}$**

Answer the following questions:

QUESTION 1 PART 1: Calculate the network delay from client to server (**$D_{c\text{-}s}$**) in seconds. **[2 marks]**

Solution: $D_{c\text{-}s}$ = [(1300-1150) - (1245-1195)]/5 = **20 seconds**

QUESTION 1 PART 2: Compute the actual (original) offset value in seconds. That is, if the attacker had not modified **T_2** and **T_3**, how many seconds would the client need to add to its own clock in order to sync with the server's clock? **[2 marks]**

Solution: Offset = T_3 + $D_{s\text{-}c}$ - T_4 = 1245 + 80 - 1300 = **25 seconds**

QUESTION 1 PART 3: Calculate the modified **T_2** value (modified by the attacker) in seconds. **[2 marks]**

Solution: 1195 + 50 - 25 = **1220 seconds**

QUESTION 1 PART 4: Calculate the modified **T_3** value (modified by the attacker) in seconds. **[2 marks]**

Solution: 1245 + 50 - 25 = **1270 seconds**

QUESTION 2: Consider a distributed system with four processes: P1, P2, P3, and P4. Assume that all the processes have an agreement on physical time at any moment. The following events occur in the system: [T refers to the timestamp according to physical time.]
Event A: At T = 1, P1 sends a message M1 to P2
Event B: At T = 2, P2 sends a message M2 to P3
Event C: At T = 2, P3 sends a message M4 to P4
Event D: At T = 3, P2 receives the message M1 from P1
Event E: At T = 4, P1 sends another message M3 to P2

Event J: At T = 10, P2 receives the message M3 from P1

Assume that the messages are sent or received by any process based on the specified T. Answer the following questions:

QUESTION 2 PART A: Which of the following message pairs may be causally dependent? **[2 marks]**

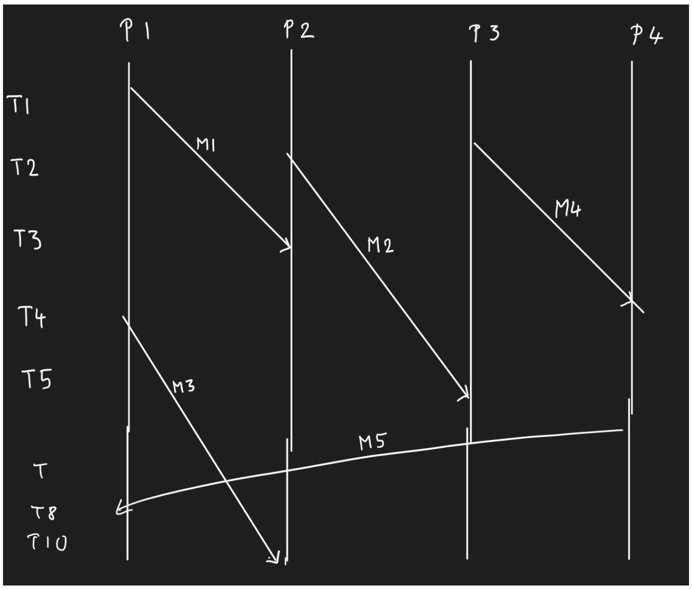
a. M1 & M3
b. M1 & M5
c. M4 & M3
d. M4 & M5

Solution: a and d

QUESTION 2 PART B: Which of the following message pairs are concurrent? **[2 marks]**

a. M4 & M5
b. M3 & M4
c. M3 & M5
d. M1 & M2

Solution: b, c and d



QUESTION 3: Consider a Token-Ring network with **N = 5** nodes. The network transmission time for transmitting the token between any two nodes is **2 seconds**. Initially, each node has an equal probability of **p = 0.7** for accessing the critical section. However, node 3 malfunctions, always requesting access to the critical section whenever it receives the token (regardless of its previous state). Determine the average amount of time (in seconds) it takes for the token to complete one full cycle around the ring. Assume that each node

a shared resource in a ring of 10 nodes. Nodes are numbered from 0 to 9. Node **k** passes messages to node **(k+1)%10**. In this variant, we have 10 tokens circulating in the ring instead of a single token in the original Token-Ring algorithm. A node can only pass one token in one message to the neighboring process. Assume that any node requires all the 10 tokens for accessing the shared resource. Consider a scenario wherein **Node 0** is currently having no tokens and it wants to access the shared resource. Answer the following questions:

QUESTION 4 PART 1: What is the minimum number of messages that need to be exchanged in the ring for **node 0** to access the shared resource? **[2 marks]**

Solution: 10 (all tokens are there with the node 9. So, it forwards these tokens to node 0 one by one)

QUESTION 4 PART 2: What is the maximum number of messages that can be exchanged in the ring before node 0 is allowed to access the shared resource? **[2 marks]**

Solution: 10 * 9 = **90** (all tokens are there with the node 1. So, for each token, the token needs to traverse the entire ring from node 1 to node 2….. to node 0)
89 is also accepted as a correct answer

QUESTION 5: Consider the decentralized algorithm for mutual exclusion having 7 replicas of the coordinator. Consider a scenario wherein there are two processes which simultaneously send out requests to all the 7 replicas for accessing the shared resource. What is the **LEAST** number of messages which need to be exchanged in the system until both processes are able to access the shared resource? **[3 marks]**

Solution: **56:** In the first round, let us assume Process 1 gets all the votes. So, process 1 will be involved in 21 messages (7 requests + 7 granted votes + 7 release messages). In this first round, Process 2 will be involved in 14 messages (it won't get any votes, so it does not have to release anything). Then in the second round, Process 2 will be involved in 21 messages. In total, **56** messages will be exchanged in the system. Any other scenario should also lead to 56 messages only as we have an odd number of replicas. So, at least one process out of the two processes will get majority votes in the first round.
Answer: 56 or 49 or 46 or 33 or 35

N_W is number of nodes in the write quorum. **[3 marks]**

Solution: 35
Subject to constraints $N_W>N/2$ and $N_R+N_W>N$, we can write the following cases:
$N_W=5$, $N_R=5, 6, 7, 8, 9$
$N_W=6$, $N_R=4, 5, 6, 7, 8, 9$
$N_W=7$, $N_R=3, 4, 5, 6, 7, 8, 9$
$N_W=8$, $N_R=2, 3, 4, 5, 6, 7, 8, 9$
$N_W=9$, $N_R=1, 2, 3, 4, 5, 6, 7, 8, 9$
Total=35
Answer: 35 or 45 (ignoring read-write conflicts**)**

QUESTION 7: Select the correct statements with respect to Primary Backup Remote-Write (PBRW) Protocol: **[2 marks]**

1. Blocking version of PBRW offers higher fault tolerance as compared to the non-blocking version of PBRW
2. Non-blocking version of PBRW offers higher fault tolerance as compared to the blocking version of PBRW
3. Blocking version of PBRW offers faster writes as compared to the non-blocking version of PBRW
4. Non-blocking version of PBRW offers faster writes as compared to the blocking version of PBRW

Solution: Options 1 and 4

QUESTION 8: Consider the below scenario in Raft where there are **seven nodes** (Node 1 to Node 7) in the system and **two clients** (Client A and Client B):

1. Initially, when the system starts up, each node starts as a follower and has **term = 0**.
2. Node 3 times out first and subsequently becomes the leader (without any further timeouts).
3. Now, **Client A** contacts the current leader and asks it to **set the value of variable x to 3**. The leader is able to commit the entry in its own state as well as propagate this update and commit to all the followers.
4. Then a network partition takes place. Nodes 1, 3, and 7 come in **Partition-1** and the rest go to **Partition-2**.
5. **Client A** again contacts the leader in **Partition-1** and asks it to add 2 to the current value of **x**. The leader then informs the followers in this partition about this operation.

current value of **x**.

8. A considerable amount of time passes by. So, updates have been communicated from the leader in both partitions to their respective followers.
9. Then the network partition gets resolved. No node times out. All the nodes arrive at a consensus about the final leader and the valid log entries.
10. **Client A** again contacts Node 3 and asks it to add 2 to the current value of x.
11. **Client B** contacts Node 2 and asks it to add 4 to the current value of x.
12. A considerable amount of time again passes by. So, updates have been communicated from the leader to all the followers.

Answer the following questions based on the above mentioned scenario:

QUESTION 8 PART 1: What is the value of term in Node 1 just before the network partition is resolved? **[1 mark]**

Solution: 1

QUESTION 8 PART 2: What is the state of **x** in Node 1 just before the network partition is resolved? **[1 mark]**

Solution: x=3 (addition of 2 remains in uncommitted state and hence, is not applied to the state)

QUESTION 8 PART 3: Which of the following is true for **Partition-1** after **Client A** again requested an update of the value of x (**Step 5 above**)? **[1 mark]**

1. The update operation was not appended to the log file.
2. The update operation was appended to the log file but left uncommitted.
3. The update operation was committed on the leader but not on other nodes in the partition.
4. The update operation was committed on the leader as well as other nodes in the partition.

Solution: Option 2 is correct. There is no majority in Partition-1, so the operation is only appended and not committed.

Node 2 just before the network partition is resolved? **[1 mark]**

Solution: x=6 (addition of 3 is committed and applied to the state as majority is present in Partition-2)

QUESTION 8 PART 6: What is the number of messages exchanged in **Partition-2** from the moment Node 2 and Node 4 time out (election timeout) to the moment Node 2 becomes the leader? **[2 marks]**

Solution: 3 RequestVote requests, each sent by Node 2 and Node 4 (Total 6 messages)
3 RequestVote responses, each received by Node 2 and Node 4 (Total 6 messages)
3 RequestVote requests, sent by Node 2 (Total 3 messages)
3 RequestVote responses, received by Node 2 (Total 3 messages)
3 empty append entries to declare node 2 as the leader from node 2 to nodes 4, 5, 6 (Total 3 messages)
3 responses from nodes 4, 5, 6 to node 2 accepting leadership (Total 3 messages)
So, the answer is 24.
Answer: 24 or 21 or 18

QUESTION 8 PART 7: What is the value of the term after the network partition is resolved and nodes have come to consensus about the final leader? **[1 mark]**

Solution: 3

QUESTION 8 PART 8: What is the state of **x** in Node 3 after the network partition has been resolved and the last two client updates have been propagated and committed throughout the system? **[1 mark]**

Solution: x=12 (3 + 3 + 2 + 4). The uncommitted entry before the network partition was resolved will be removed from Node 3 and other nodes.
Answer 12 or 10

QUESTION 8 PART 9: Which node is the final leader in the system after the network partition has been resolved and consensus has been reached? **[1 mark]**

Solution: Node 2

QUESTION 9: Consider a distributed system with four processes: P_1, P_2, P_3, and P_4. Assume that

1. $R_i(x)$ **a** [T_S, T_C] denotes a read operation on data item **x** by process P_i where
   a. **a** is the result of this read operation
   b. Read operation was issued at T_S by process P_i
   c. Read operation completed at T_C (i.e. P_i got the result **a** of read operation at T_C)
2. $W_i(x)$ **b** [T_S, T_C] denotes a write operation on data item **x** by process P_i where
   a. **b** is the value written to the data store for data item x
   b. Write operation was issued at T_S by process P_i
   c. Write operation completed at T_C (i.e. Until T_C, P_i was blocked waiting for the write request to return. At T_C , P_i was notified of the completion of the write request. Note that this does not necessarily mean that the write operation completed on the data store / replica.)

The variable **x** has been initialized to 0. After this initialization, the following events occur in the system:

1. **P_1:** W_1(x)10 [1, 5], W_1(x) 20 [6, 10]
2. **P_2:** W_2(x) 30 [3, 7], W_2(x) 40 [9, 13]
3. **P_3:** R_3(x) **B** [8, 11], R_3(x) **C** [12, 14]
4. **P_4:** R_4(x) **A** [2, 5.5], R_4(x) **D** [15, 16]

Answer the following questions:

QUESTION 9 PART 1: Select all the scenarios which are allowed by strict consistency? **[2 marks]**

1. A = 0, B = 20, C = 40, D = 40
2. A = 10, B = 20, C = 40, D = 40
3. A = 0, B = 30, C = 20, D = 40
4. A = 10, B = 30, C = 20, D = 40

Solution: **Option 2**

QUESTION 9 PART 2: Select all the scenarios which are allowed by linearizability consistency? **[2 marks]**

1. A = 0, B = 30, C = 20, D = 40
2. A = 10, B = 20, C = 40, D = 40
3. A = 10, B = 30, C = 20, D = 40
4. A = 10, B = 20, C = 30, D = 20

Solution: **Option 1** [using the understanding of what we discussed about linearizability in class]
Option 1, Option 2, Option 3 [using the actual/correct understanding of linearizability]

Solution: **Option 1, Option 4**

QUESTION 9 PART 4: Select all the scenarios which are allowed by causal consistency? **[3 marks]**

1. A = 10, B = 40, C = 40, D = 40
2. A = 20, B = 30, C = 30, D = 40
3. A = 10, B = 20, C = 20, D = 40
4. A = 10, B = 30, C = 10, D = 30

Solution: if all of the following two options are marked: **Option 1** and **Option 3**, then give full marks [3 marks]
if all of the following three options are marked: **Option 1**, **Option 3 and Option 4**, then give 1.5 marks

QUESTION 9 PART 5: Select all the scenarios which are allowed by eventual consistency? **[3 marks]**

1. A = 30, B = 20, C = 30, D = 40
2. A = 20, B = 30, C = 30, D = 40
3. A = 10, B = 20, C = 20, D = 40
4. A = 10, B = 30, C = 10, D = 30

Solution: if only **Option 3** is marked, then give full marks [3 marks]
if all of the following three options are marked: **Option 1**, **Option 3 and Option 4**, then give 1.5 marks