CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

Note:

1. **Write your name and GT number on each page.**
2. The test is **CLOSED BOOK** and **NOTES.**
3. Please provide the answers in the space provided. You can use scratch paper (provided by us) to figure things out (if needed) but you get credit **only** for what you put down in the space provided for each answer.
4. For conceptual questions, **concise bullets** (**not wordy sentences**) are preferred.
5. While it is NOT REQUIRED, where appropriate use figures to convey your points (a figure is worth a thousand words!)
6. **Illegible answers are wrong answers.**
7. **DON'T GET STUCK ON ANY SINGLE QUESTION…FIRST PASS: ANSWER QUESTIONS YOU CAN WITHOUT MUCH THINK TIME; SECOND PASS: DO THE REST.**

Good luck!

Question number		Points earned	Running total
0 (0 minute)	(Max: 1 pts)		
1 (5 minutes)	(Max: 8 pts)		
2 (5 minutes)	(Max: 10 pts)		
3 (5 minutes)	(Max: 9 pts)		
4 (5 minutes)	(Max: 10 pts)		
5 (8 minutes)	(Max: 10 pts)		
6 (3 minutes)	(Max: 6 pts)		
7 (10 minutes)	(Max: 16 pts)		
8 (10 minutes)	(Max: 18 pts)		
9 (10 minutes)	(Max: 12 pts)		
Total (61 minutes)	(Max: 101 pts)		

0. **(1 point, 0 minute)**(you get 1 point regardless of your answer)
This test format:
(a) increases my anxiety for test taking
(b) reduces my anxiety for test taking
(c) allows me to learn the material much better
(d) makes me procrastinate getting ready for the test

CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

Lesson 5: Distributed Systems

1. **(8 points, 5 minutes) (Lamport's logical clock)**

A student has implemented a distributed algorithm using Lamport's happened-before relationship to timestamp the events. She is in the middle of debugging the program. She observes the following events for each processor. Each event is tagged with the local timestamp recorded for the event on that processor following Lamport's logical clock.

P1's activities	P2's activities	P3's activities
1: msg-send (to ???)		1: local event
	2: msg-receive (from ??)	2: msg-send (to ??)
	3: msg-receive (from ??)	3: local event
	4: msg-send (to ??)	4: local event
5: msg-receipt(from ???)	5: msg-send (to ??)	
		6: msg-receive (from ??)

Please help her by identifying who is the sender/receiver for each message (namely, the sender, receiver, and the logical timestamp associated with that event.

In answering the question use the following format:
<processor-number>: <local-time>: msg-<send/receive> (<from>/<to>)
(e.g., P1: 20: msg-receive(P3) would mean P1 received a message from P3 at local time 20)

Answer:
P1: 1: msg-send(P2)
P1: 5: msg-receive(P2)
P2: 2: msg-receive(P1)
P2: 3: msg-receive(P3)
P2: 4: msg-send(P1)
P2: 5: msg-send(P3)
P3: 2: msg-send(P2)
P3: 6: msg-receive(P2)

1 point each

(Kishore's note: should we worry about double jeopardy?)

2. **(10 points, 5 minutes) (Lamport's M.E. Algorithm)**
(i) (5 points)
Lamport's ME algorithm requires two conditions to be satisfied for a node to assume that it has the lock:
(1) Its request is at the top of its request queue.
(2) It has received ACKs from all the other nodes for its request.
Explain how the algorithm could work correctly even without the second condition being satisfied.

CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

Answer:
If a node N1 has received a lock request from another node (say N2) that has a later timestamp than its own lock request then N1 knows that it will get the lock before N2, so even if N1 has not received an ACK from N2 it can correctly conclude that it has the lock if the first condition is satisfied.

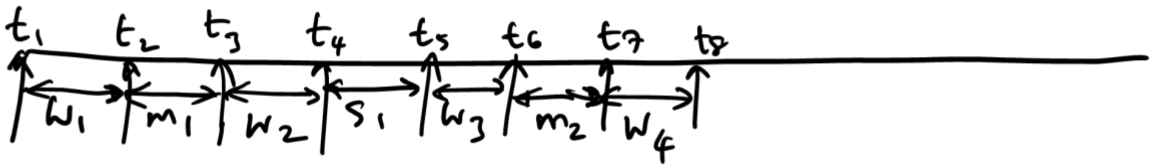
(all or nothing)

(ii) (5 points)
Lamport's ME algorithm relies on the fundamental assumption that no message is lost, and messages go in order between any two nodes. Let's assume that no LOCK/UNLOCK requests are lost but ACKs may sometimes be lost. Let's assume also that there is LIVENESS in the system (i.e., every processor is guaranteed to make LOCK requests periodically).
Answer with justification whether the ME algorithm will work or fail under these circumstances.

Answer:

1. **Let's suppose N1 makes a lock request.**
2. **Since lock requests are NOT lost, every node (say N2 as an example) would have received N1's lock request. (+1)**
3. **Let's assume that N2's ACK in response to this lock request is lost**
4. **The liveness guarantee (every processor will eventually make a lock request) ensures that N1 will receive a lock request from N2 at some future time. (+1)**
5. **Due to Lamport's logical clock, it is guaranteed that this lock request from N2 will have a later timestamp than N1's (follows from step 2). (+1)**
6. **The new lock request from N2 (with a later timestamp) serves as an implicit ACK for N1's lock request. (+1)**
7. **Thus the ME exclusion algorithm will work correctly so long as LOCK and UNLOCK messages are never lost. (+1 only if there is accompanying justification even if it is flawed)**

3. **(9 points, 5 mins) (Latency Reduction in RPC)**
(i) (5 points) Consider the following timeline:



Legend:

CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

t1: Client makes an RPC call
t2: protocol processing on client side kernel (shown as w1 = t2-t1) for marshaling arguments of the call is complete
t3: message transmission (shown as m1 = t3-t2) to the server node complete
t4: protocol processing to receive the message (shown as w2 = t4-t3) on server node complete
t5: context switching to the server process, unmarshalling, and server procedure execution (shown as s1 = t5-t4) complete
t6: protocol processing on server side for marshalling the results (shown as w3 = t6-t5) on server node complete
t7: message transmission (shown as m2 = t7-t6) to the client node complete
t8: protocol processing to receive the message (shown as w4 = t8-t7) on client node complete

Let X be the context switch time at client node. Under what condition will it make sense for the kernel on the client node to spin awaiting the response for the RPC call. Explain your answer.

Answer:

It would make sense to spin-wait if the following condition is satisfied
X > m1+w2+s1+w3+m2 (Eq.1)

- **(m1+w2+s1+w3+m2) represents the time that would elapse before the response is received from the client-side kernel** **(+5: basically 1 point for each of the above 5 components)**
- **explanation as below for why w1 NOT in Eq.1** **(+2)**
  - **w1 is the work to be done on behalf of the client RPC before the kernel can transmit the message to the server.**
  - **So, t2 is the earliest time for the client-side kernel to make a decision on spin or context switch**
- **explanation as below for why w4 NOT in Eq.1** **(+2)**
  - **t7 is the time at which the response is received by the client-side kernel and the kernel can go to work to prepare the results for the client that made the RPC call**
  - **w4 is the work to be done by the client-side kernel before the client can be scheduled to run**

(ii) (4 points)
What are the opportunities for overlapping computation with communication in an RPC on the client and the server sides?

Answer:

CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

- **context switching the client during the RPC call overlapped with communication to the server** **(+2)**
- **buffering the results of the RPC call for potential retransmission (anticipating message loss) overlapped with communication of the results to the client** **(+2)**

Lesson 6: Distributed Objects and Middleware

4. **(10 points, 5 minutes) (Spring OS)**

You are managing a subset of nodes in a cluster. You have chosen to use Spring as the network OS for the cluster. You must host the following services:

1. A PostGRES database server which is replicated on 3 nodes
2. A Web server on 3 nodes
3. A Web-server-load-balancer that does ensures equitable CPU utilization on all the servers for the client requests.
4. A PostGRES-load-balancer that does round-robin allocation of the servers for the client requests.

Each of the above servers and the load balancers are hosted on distinct nodes on the LAN. The clients are all on the same LAN and are expected to make requests to both the POSTGRES and Web service.

(i) (6 points) For the above deployment to work as envisioned above,
(a) list the subcontracts needed on the client machines.
Subcontract for web server load balancer and postGRES load balancer **(+2)**
subcontract for web server or postGRES is included **(-1)**
(b) list the subcontracts needed on the Web-server-load-balancer Subcontract for web server **(+2)**

(c) list the subcontracts needed on the PostGRES-load-balancer
Subcontract for postGRES **(+2)**

(ii) (2 points) You decide to beef up your web server with 2 more nodes. What changes will you need to make to ensure that the client requests can utilize the two new nodes?
We only need to change the subcontract on the web server load balancer to add the two new machines **(+2)**
Other stubs/subcontracts additionally change **(-1)**

CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

(iii) (2 points) You now need to access the PostGRES database from the web service. What changes do you need to make to the system?

We need to add asubcontract for the PostGRES load balancer from the web service nodes. **(+2)**

Other stubs/subcontracts additionally change **(-1)**

any other answer **(-2)**

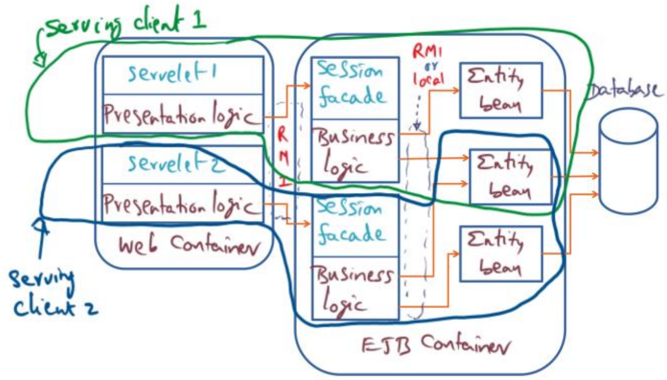
5. **(10 points, 8 minutes) (EJB)**

It is circa 2002. Yelp and Google Reviews don't exist yet. You're a developer and a foodie. You decide to build a restaurant review website that has the following functionalities:

1. Accept a restaurant name or a cuisine as input and display a list of restaurants with their ratings.
2. If a user clicks on a restaurant, they will be shown the reviews for the restaurant.
3. The user should be able to sort restaurants by distance from their location, and average review score.
4. Allow a user to post a review about a restaurant and store it in the database, along with some keywords (e.g., cuisine, ambience, etc.), which may or may not explicitly be tagged with the user-name.

Now you realize that restaurant searches are hyper-local, so you only need to show the user the restaurants which are within a 15-mile radius. So, you decide to use the user's GPS location to filter results.

You decide to implement the system with the state of the art, i.e., EJB entity beans as shown below:



CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

(i) (6 points) For the functionality that you need on the website, describe concisely what components go into the presentation logic, business logic and the entity beans.

Presentation logic:

1. The logic to display the restaurants and reviews **(+1)**
2. Accepting user location and user review text and rating **(+1)**

Business logic:

1. Searching/sorting reviews by rating and location **(+1)**
2. Massaging user review text (keyword extraction, metadata addition, timestamping) to the review to be written to the database. **(+1)**

Entity Beans:

1. User/Review Bean to read/write reviews for the restaurant **(+1)**
2. Restaurant Bean to query the database by keyword and restaurant name and distance from the user **(+1)**

(ii) (2 points) How would you optimize for latency for concurrent requests from users in the same location?

Add a cache in the EJB container for popular queries to the database and their corresponding results **(+2)**

(or any other reasonable answer)

(iii) (2 points) You decide to open shop in India, where there are 22 official languages. You realize that restricting yourself to English might be a problem. Where in this framework would you add the functionality to render content in different languages? Justify your answer (No points without justification).

Add a module in the presentation logic for translation. (+1 if there is justification)

The functionality is independent from business logic, so it can safely be added to the presentation layer. (+1)

(or)

The storage demand on the database would significantly increase if we choose to translate all the reviews at write time. Instead we should do the translation on demand. (+1)

6. **(6 points, 3 minutes) (Java RMI)**

(i) (4 points)

Java RMI evolved from the Spring Subcontract mechanism. Name one similarity and one difference in the implementation of the two systems.

Answer:

- **Similarity: Both RMI and subcontract hide the detail of the physical location of the client and server involved in the client-server interaction (+2)**

- **Difference: RMI exploits the semantics of the Java language in marshalling/unmarshalling while subcontract is language-agnostic relying on IDL for marshalling/unmarshalling** **(+2)**

(ii) (2 points)
Java allows object references to be passed as parameters during object invocation. What is the difference in parameter passing (when a local object reference is passed as a parameter) while invoking a remote object using Java RMI?

Answer:

- **The parameter is passed using value/result** **(+2)**

Lesson 7: Distributed Subsystems

7. **(16 points, 10 minutes) (GMS)**

(i) (2 points)
Upon a page fault, GMS converts the VPN to a UID. The UID includes the IP-ADDR of an NFS server. Why?

Answer:

- **Every virtual page (whether it comes from the virtual memory subsystem or the file system) is backed up by the storage server (i.e., NFS server). So including the IP-ADDR of the NFS server is a convenient way to generate the UID from the VPN** **(+2)**

(ii) (2 points)
Your friend suggests that the UID could be generated by simply prefixing the faulting VPN with the IP-ADDR of the source node and the PID of the faulting process. Will that work? Credit only if there is justification for your answer.

Answer:

- **No.** **(+1 if there is justification)**
- **Once a process terminates, the associated PID may get recycled at a given node by the OS for a different process.** **(+1 if the justification is valid)**

(iii) (2 points) (Answer True/False with justification)
For the geriatrics algorithm, it is straightforward to record the time of access to the virtual pages visited by a process during its execution.

Answer:

- **No.** **(+1 if there is justification)**
- **During process execution as long as there is no page fault, the order in which the process visits its mapped pages during its time quantum on the processor is not visible to the OS. The best bet is the TLB**

which contains the recently visited pages (still not exact since it does not contain the order in which the pages were visited) **(+1 enough if they say page access by a process not visible to the OS)**

(iv) (6 points)
Assume that there is a designated node (which never fails) in charge of additions/churns of the nodes participating in GMS. A new node joins the GMS.
(a) List the actions that ensue.

Answer:

- **The designated node (master) recomputes the partitions of the UID space for each node (including the new node)** **(+1)**
- **It creates a new POD and distributes it to all the nodes**
- **Each node reconstructs its GCD based on the assignment in the POD**
- **Each node sends the GCD entries that it is no longer responsible for to the new node that has come online** **(+1)**

(b) What data structures of GMS will get modified as a result? Why?

Answer:

- **POD because of the reassignment by the master** **(+1)**
- **GCD since each node is responsible for managing a different subset** **(+1)**

(c) What data structures of GMS will not get modified as a result? Why?

Answer:

- **PFD because the node that is currently housing a page in its DRAM does not change due to the addition of a node** **(+2)**

(v) (4 points)
Your friend suggests implementing GMS with a replicated table in each node that gives the mapping UID -> <nodeid, pframe>.

(a) What work would need to be done on each page fault?

Answer:

- **Convert the VPN to UID and broadcast the UID to all the nodes. Get back the node id (say N1) that has the PFD for that UID from whichever node happens to have that information** **(+1)**
- **Get the page from the PFD in N1** **(+1)**

(b) What work would need to be done on each page eviction from a node.

Answer:

- **Broadcast the change (UID, <new-home>) for the evicted page (+1)**
- **Update the UID table on all the nodes (+1)**
- **Update the PFDs on the old-home and new-home nodes**

8. **(18 points, 10 minutes) (DSM)**
(i) (2 points) (Answer True/False with justification)
For correctness of a multiprocess application using DSM running on a LAN cluster, a given virtual page should be mapped to the same physical frame number in each node.

Answer:

- **No (+1 if there is justification)**
- **DSM provide address equivalence at virtual page level (+1)**

(ii) (2 points) (Answer True/False with justification)
For correctness of a multiprocess application using DSM running on a LAN cluster, the size of physical memory on each node of the cluster should be the same.

Answer:

- **No (+1 if there is justification)**
- **(+1 for either of the following thought expressed in the answer)**
- **The size of physical memory at each node has no bearing on the correctness since DSM provides address equivalence for virtual memory.**
- **If there is less DRAM at a node it just means that the application component at that node may only have a subset of the virtual pages of its working set.**

(iii) (2 points)
Consider a DSM application that has no data races. It properly uses synchronization with mutual exclusion locks to safeguard access to shared data structures. The underlying DSM uses SC memory model with page level granularity for coherence. The application experiences poor performance. Explain why.

Answer:

- **SC memory model cannot distinguish between synchronization accesses and normal data read/write accesses. (+1)**

- **Every data write has to ensure that the coherence action is fully completed in the entire cluster for the SC memory model. (+1)**

(iv) (4 points)
The data structures protected by a lock is in the purview of the application and the DSM system has no knowledge of this association. LRC ensures the coherence maintenance of the data structures protected by a lock to the point of lock acquisition. How is coherence maintenance achieved in LRC?

Answer:

- **During the execution of a critical section protected by a lock, the DSM system will get notified by the OS when a process attempts to write to a page. (+1)**
- **This is implicit knowledge that DSM infers that the data structure protected by the lock is contained in that page. (+1)**
- **(+2 if the following thought is expressed in some fashion)**
- **A twin of the page is created.**
- **The process writes to the original page**
- **The DSM records diffs of changes to pages modified in the critical section at the release point.**
- **At the next lock acquisition diffs will be applied to a page modified by previous critical sections that used the same lock, when a process tries to access that page.**

(v) (8 points)
In Treadmarks DSM system the following critical section is executed at a node N1:

```
Lock(L1);
   Write to Page X;
   // Assume that the page is not present at this node;
   // Assume that there are three diff files for page X
   // named, $X_d^2$, $X_d^3$, and $X_d^4$ in nodes N2, N3, and N4,
   // respectively.
   // Assume the sync causality for the lock L1 is
   // N3 -> N2 -> N1 (i.e, this is the order of lock acquisition).
Unlock (L1);
```

a) (2 points)
What actions would be carried out by Treadmarks at Node 1 before the critical section above is executed by N1?

Answer:

- **DSM will fetch a pristine copy of the page from the owner for that page. (+1)**
- **It will apply the diffs X_d^3 and X_d^2 IN THAT ORDER to page(X) (+1)**

(no credit if the order is wrong; no credit if X_d^4 is also applied)

b) (2 points)
Upon exiting the critical section what action would be carried out by Treadmarks at Node N1?

Answer:

- **DSM will record the diff for X (call it X_d^1) at this node N1 (+1)**
- **With the OS help, it will mark the modified page X as read-only at node N1 (+0.5)**
- **With the OS help it will get rid of the twin that was created for page(X). (+0.5)**

A little while later, Node N1 executes the following critical section. Assume that no other node acquired the lock L1 in the interim.

```
Lock(L1);
  Read page X;
  do some computation without changes to any pages;
Unlock(L1);
```

(c) (2 points)
What would be the action carried out by Treadmarks at Node 1 before the critical section above is executed by N1?

Answer:

- **No action by DSM since the page(X) is available already locally for reading (+2)**

d) (2 points)
Upon exiting the critical section what would be the action by Treadmarks at Node N1?

Answer:

- **None (+2)**

9. **(12 points, 10 minutes) (DFS)**
Inspired by xFS, you and your classmate decide to implement a true distributed FS. In your implementation similar to xFS, the location of the files on the disks remain fixed (i.e., they are never migrated). However, you periodically assess the meta-data server activity on each node and redistribute the meta-data management to balance the load in the entire cluster. In your implementation you are using the same data structures as in the original xFS.

CS 6210 Fall 2022 Test 2 Solutions (should take 60 mins; however, you have 120 minutes)

Name:______CS 6210 Teaching Team_________GT Number:

(i) (4 points)
You observe that there is a load imbalance in the system. You carry out a load re-distribution.
List the steps in your meta-data load re-distribution algorithm.

Answer:

- **Let's assume that a designated node (manager) performs the re-distribution algorithm periodically (epoch based with a system defined parameter T for the period).**
- **Let's assume that each node informs the manager the activity (i.e., meta-data server load) at that node at the beginning of each epoch.**
- **Here are the steps of the re-distribution algorithm:**
    - **Freeze all the nodes from making changes and ask them to send their respective imap data structures (+1)**
    - **Recompute the mmap taking the load statistics into account (+1)**
    - **Recompute the imap for each node based on the new mmap (+1)**
    - **Re-distribute the new mmap and imap to all the nodes and unfreeze the nodes to resume normal action (+1)**

(ii) (4 points)
What data structures change as a result of the load re-distribution algorithm? Why?

Answer:

- **Mmap and imap data structures (+2)**
- **mmap gives the association between i-number and the node that manages the meta-data for that i-number (+1)**
- **imap gives the location of the inode for a given i-number (+1)**

(iii) (4 points)
What data structures do not change as a result of the load re-distribution algorithm? Why?

Answer:

- **Except for mmap and imap none of the other data structures change (+2)**
- **File-dir: association of file-name to i-number fixed at creation time and does not change (+1)**
- **Stripe-group map: files don't move so this does not change (+1)**