CS425 Fall 2018 – Homework 4

(a.k.a. "The Interview")

Out: Nov 8, 2018. Due: Dec 6, 2018 (Start of Lecture. 2 pm US Central time.)

Topics: Lecture 19 onwards

Instructions:

1. **Attempt any 8 out of the 10 problems** in this homework (regardless of how many credits you're taking the course for). If you attempt more, we will grade only the first 8 solutions that appear in your homework (and ignore the rest). Choose wisely!
2. Please hand in **solutions that are typed** (you may use your favorite word processor. We will not accept handwritten solutions. Figures and equations (if any) may be drawn by hand (and scanned).
3. **All students (On-campus and Online/Coursera) –** Please submit PDF only! Please submit on Gradescope. [https://www.gradescope.com/]
4. Please **start each problem on a fresh page**, and **type your name at the top of each page**.
5. Homeworks will be **due at the beginning of class on the day of the deadline. No extensions. For DRES students only:** once the solutions are posted (typically a few hours after the HW is due), subsequent submissions will get a zero**. All non-DRES students must submit by the deadline time+date.**
6. Each problem has the same grade value as the others (10 points each).
7. Unless otherwise specified, the only resources you can avail of in your HWs are the provided course materials (slides, textbooks, etc.), and communication with instructor/TA via discussion forum and e-mail.
8. You can discuss lecture concepts and the questions on Piazza and with your friends, but you cannot discuss solutions or ideas. All work must be your own.

Prologue: After the success of your epic Mars mission and your celebrated return to Earth, you receive many offers of employment at top companies and universities. But before you get hired, you have to go through interviews at these places. Your goal is to attempt 8 interviews and ace them so that you have the maximum choice of where you want to (After all, if you're attached, the choice will likely be made by your significant other, so you want to give her/him the most choice! If you're not attached, you want to have the most choice anyway, right? ;).

The storylines, statements, events, things, and games in this homework are purely fictitious. Any resemblance to persons, places, or events, living or dead, past, present, or future, is purely coincidental.

Problems:

1. During an interview at IBM Research Almaden, they tell you that the relational database model was invented by E. F. Codd, so they love transactions. They give you a log of two transactions executed concurrently by two clients – T1 and T2 (a, b, c, d are objects at the server):
   T1: read(a, T1); read(b, T1); read(a, T1); write(c, foo, T1);
   T2: read(b, T2); write(d, bar, T2); write(a, baz, T2); read(c, T2);
   The transaction management system seems to think each of the following interleavings is serially equivalent. For each of the following interleavings, say if (and why/why not) it is serially equivalent:
   a. read(a, T1); read(b, T1); read(b, T2); write(d, bar, T2); read(a, T1); write(a, baz, T2); read(c, T2); write(c, foo, T1);
   b. read(b, T2); write(d, bar, T2); read(a, T1); read(b, T1); write(a, baz, T2); read(a, T1); read(c, T2); write(c, foo, T1);
   c. read(a, T1); read(b, T1); read(b, T2); write(d, bar, T2); read(a, T1); write(a, baz, T2); write(c, foo, T1); read(c, T2);
   d. read(b, T2); read(a, T1); read(b, T1); read(a, T1); write(d, bar, T2); write(a, baz, T2); write(c, foo, T1); read(c, T2);

2. When interviewing at Oracle, they tell you that they use a transaction management system that assigns unique word ids to objects, where the words are dictionary words (Oxford English dictionary only). The system uses exclusive locking of objects for concurrency control. The system restricts transactions to acquire locks only in lexicographically *decreasing* order of object id (i.e., if a word appears later in the dictionary it has a higher object id, and it will be locked first). They tell you that this system will not deadlock only if all locks are acquired at the start of the transaction. You think otherwise, and say that even if you acquire locks during the transaction (obeying the decreasing lexicographic rule), there will be no deadlocks. Who is right? Give an example or present a proof, as applicable. You can assume 2 phase locking for this problem.

3. In traditional two phase locking, as you've seen in lecture, a transaction cannot acquire (or promote) any locks after it has started releasing locks. Thus transactions have a growing phase and a shrinking phase. During one of your interviews you meet Gordon Gekko, who tells you that approach is too safe, and that greed is better. But Gordon has been tempered by his time in prison. He tells

you that he has a new algorithm called limited-greedy two phase locking. The idea is the following: there is still a growing phase where locks can only be acquired. However, once the shrinking phase starts, the transaction has an opportunity to acquire *at most one new* lock (objects are never locked more than once by one transaction).

   a. Does this limited-greedy two phase locking still satisfy serial equivalence? If yes, give a proof. If no, show a clear counterexample.
   b. Can this limited-greedy two phase locking deadlock? If yes, give a clear counter-example. If no, give a proof.

4. You cannot use the Web for this question. While you're interviewing at the top-secret E. Corp., they face a catastrophic attack from a group of hackers known as fsociety. The attack partitions their datacenter into three partitions – the NYC partition, the NJ (New jersey) partition, and the DC partition. Calmly, you tell your interviewers that this is called a partitioned system, you learnt this in CS425, and you've got this. You've got a few choices for how to handle this partition – these choices are listed below. For each of these choices, say if it: i) violates strong consistency (and why), ii) if it violates availability for reads (and why), and iii) if it violates availability for writes (and why).

   a. Don't allow any operations until partitions are repaired.
   b. Allow both NYC and NJ partitions to process writes, but only the DC partition to process reads.
   c. Allow all partitions to process reads, but only the DC partition to process writes.
   d. Allow only the partition that has at least a quorum number of servers (measured across all partitions) to execute writes; reads can be executed in all partitions.
   e. Allow only the partition that has at least a quorum number of servers (measured across all partitions) to execute reads; writes can be executed in all partitions.
   f. Allow reads and writes in a partition only if it has a quorum number of servers responsive (measured only within that partition).
   g. Until partitions are repaired, allow only reads but no writes.
   h. Allow only partitions with a quorum of servers (measured across all partitions) to execute writes and reads.

5. (You can use the Web as a resource for this question.) You interview at one of the biggest social media companies and they tell you they use Kafka to track hashtag trends. They say Kafka is a "publish-subscribe" system. In less than 100 words, say what a publish-subscribe system is. In your answer, be sure to include URLs/links pointing to specific characteristics, otherwise you may not

get points (these don't count in the word limit). Don't write too long, but don't write a short answer either.

6. (You can use the Web as a resource for this question.) In class, it was mentioned that Zookeeper is a system that uses a consensus-like protocol called ZAB (Zookeeper Atomic Broadcast). Summarize briefly (< 100 words) but concretely how ZAB works. Point out 3 key differences between ZAB and Paxos.

7. You cannot use the Web for this question. At Berkeley, where they invented the Mica Mote (true fact!), they say they are building a sensor network in the ocean to monitor movements of whales (and ensure that none of them are killed for sport). The sensors measure speed, but they are also mobile sensors because the whales are continuously swimming. The deployment spreads 2000 MICA motes over several thousands of square miles. They would like to get hourly measurements of the *average* speed across all your sensors (the data does not need to be seen immediately, but eventually they will need all hours). All base stations are on shore. How would you design the system? A few options are suggested to you: I) have every sensor node send its data directly to the base station, II) build a spanning tree with one base station at the root, III) don't build a spanning tree, but instead collect data from a whale only when it is near a base station.
   a. Give one advantage of each of these approaches (compared to the other two approaches).
   b. Which of these three approaches would you prefer? Why?
   c. To calculate the average via a spanning tree among the sensor nodes, what data would you pass along (up the tree)? Give precisely the calculation involved for aggregation.
   d. To calculate the first quartile value (25th percentile) via a spanning tree among the sensor nodes, what data would you pass along (up the tree)?

8. During your exciting interview at MIT, you find that they seem to like distributed shared memory. They ask you the following question involving 5 processes P1-P5 in a distributed shared memory system using invalidate. Process P3 wants to write a page. In each of the following cases, say what is the series of operations that needs to happen for P3 to be able to write (warning: there be tricks below!) -- You cannot use the Web for this question. If the setup seems wrong to you, you should point out ALL errors in it.
   a. P3 is the owner of the page, and is the only one holding it (in Read Mode)
   b. P3 is holding the page in Read mode and P4 is holding it in Write mode and P3 is the owner
   c. P4 is the owner and is holding the page in a Write mode
   d. P1 and P2 are each holding the page in a Write mode, and P3 is the owner

e. P1, P2, P3 are each holding the page in a Write mode, and P3 is the owner
f. P1, P2 and P3 are currently holding the page in Read mode, P2 is the owner
g. P2 and P4 are holding the page in Read mode, and P4 is the owner
h. P3 and P4 are both holding the page in write mode
i. P1 is holding the page in write mode
j. P4 and P5 are each holding the page in a Read mode, and P4 is the owner

9. You cannot use the Web for this question. You also interview at a top-secret spy agency, where of course they are concerned with security protocols. A noob there has designed a few new digital signature schemes that aim to be more efficient than existing ones. For each of these say if they are correct or not, and why/why not. (Hash is a strong and well-known cryptographic function).
    a. Sign a message M from Alice as: [M, $Hash(K_{Apriv}(M))$].
    b. Sign a message M from Alice as: [M, $Hash(K_{Apub}(M))$].
    c. Sign a message M from Alice as: [M, $K_{Apub}(Hash(M))$].
    d. Sign a message M from Alice as: [M, $K_{Apriv}(Hash(M))$].
    e. Sign a message M from Alice as: [M, $K_{Apriv}(M)$].
    f. Sign a message M from Alice as: [M, $K_{Apub}(M)$].
    g. Which of these above options would you choose (think: correctness and speed).

10. Your last interview is at Facebook. Your friend who works at Facebook, has in her free time, built a new distributed file system, which implements (at its low level) Unix file system read/write-like semantics, i.e., internally it maintains file descriptor data structures which contain an automated read-write pointer (at the server side). Your friend claims this makes it easier to program with, since Unix programs can use the same API. She would like to have the entire company using this design.
    a. Do you agree with this design? If yes, say why. If not, say what the problem with this design is (be specific)?
    b. How would improve this design given your knowledge of distributed file systems?
    c. (Optional, no points for answering, answer only if you want to) If you had offers from all the places listed in the questions of this homework (faculty at universities, starting positions at companies and government agencies), where would you join?
    d. PS (not a question): If you're wondering why Illinois/UIUC is not listed in the above places, it's because we encourage cross-pollination and would like our alumni to spread their knowledge everywhere around the world (true fact!).