CS425 Fall 2017 – Homework 4
(a.k.a. “The Interview”)

Out: Nov 9, 2017. Due: Dec 7, 2017 (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 hardcopy 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. Online students can email solutions, and MCS-DS students must upload on Coursera. If you’re not online/MCS-DS, and are traveling, please make other arrangements – we don’t accept email submissions.
3. Please start each problem on a fresh sheet (not just page), and type your name at the top of each sheet. Staple all your sheets together.
4. 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.
5. Each problem has the same grade value as the others (10 points each).
6. 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.
7. 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 go (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 are objects at the server):
   T1: read(a, T1); write(b, caz, T1); read(b, T1); write(c, foo, T1);
   T2: read(b, T2); write(b, 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, T2); write(b, bar, T2); write(b, caz, T1); write(a, baz, T2); read(c, T2); read(b, T1); write(c, foo, T1);
   b. read(a, T1); write(b, caz, T1); read(b, T2); read(b, T1); write(c, foo, T1); write(b, bar, T2); write(a, baz, T2); read(c, T2);
   c. read(b, T2); write(b, bar, T2); write(a, baz, T2); read(a, T1); write(b, caz, T1); read(b, T1); write(c, foo, T1); read(c, T2);
   d. read(b, T2); read(a, T1); write(b, caz, T1); read(b, T1); write(c, foo, T1); write(b, bar, T2); write(a, baz, T2); read(c, T2);

2. When interviewing at Oracle, they tell you that they use a transaction management system where each object is assigned an ID that is a vector timestamp. All vector timestamps have the same number of elements in the vector (fixed length). The vector timestamp assigned to an object is fixed, because it refers to the vector timestamp at the time the object was first created. The system currently uses exclusive locking of objects for concurrency control, and relies on 2-phase locking (as discussed in class). The system restricts transactions to acquire locks only in *increasing* order of vector timestamp: given two objects Oa and Ob, if TS(Oa) < TS(Ob) then lock Oa before Ob; else if TS(Ob) < TS(Oa) then lock Ob first before Oa; else pick randomly between Oa and Ob to lock first. They tell you they are facing deadlocks, but have no idea why.
   a. Give a counter-example to show how a deadlock may occur.
   b. If their scheme does not work, modify it slightly so that it works. Prove formally that your new scheme will avoid deadlocks. This must be a formal proof.
3. 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), and ii) if it violates availability (and why).
   a. Don’t allow any operations until partitions are repaired.
   b. Allow only reads on all partitions, writes in none.
   c. Allow writes in the partition that has a quorum of nodes (from the original system). Allow reads in all partitions.
   d. Allow reads and writes in all partitions.
   e. Allow reads in the partition that has a quorum of nodes (from the original system). Allow writes in all partitions.
   f. Allow writes only in the DC partition, allow reads in all partitions.
   g. Allow writes only in the DC and NYC partitions, but allow reads in all partitions.

4. (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.

5. (You can use the Web as a resource for this question.) While interviewing at a high-profile startup in the Bay Area, you see they use Storm and Heron but don’t seem to have heard of Spark Streaming. Describe clearly what Spark Streaming does and how this differs from the stream processing systems just mentioned. Limit to answer to 100 words. 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.) Your next interview is at Harvard University. Here they are proud of their alumnus George Kingsley Zipf (true fact!). They present you different frequency distribution plots (number of accesses vs. file popularity rank). The labels on both the axes are equi-spaced. For each of them, say whether the distribution is logarithmic, exponential, Zipf,
linear, none of these, or whether there is insufficient information. Say why in each case.

a. The Y axis labels are 10, 20, 30, 40, … The X axis labels are 30, 90, 270, 810. The line drawn in the plot curves downwards.

b. The Y axis labels are 10, 20, 30, 40, … The X axis labels are 1KB, 1MB, 1GB, and so on. The line is a straight downwards sloping line.

c. The Y axis labels are power of 2 … The X axis labels are powers of 5. The line is a curved line that slopes downwards.

d. The Y axis labels are power of 2 … The X axis labels are powers of 5. The line is a straight downwards sloping line.

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 on an African preserve to monitor movements of lions (and ensure that none of them are killed for sport). The sensors measure sound. The deployment spreads 2000 MICA motes over several hundreds of square miles. They would like to periodically (every minute) measure the average sound across all your sensors. You have two options: either having the sensor nodes route all their sound measurements to a base station (routed via other sensor nodes) which in turn then calculates the average temperature reading, or have the sensor nodes talk to each other and calculate the average amongst each other. Answer the following questions:

a. Which of these two options would you choose? Give at least one major reason why you chose that option.

b. 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.

c. To calculate the median (50th percentile) via a spanning tree among the sensor nodes, what data would you pass along (up the tree)?

8. You cannot use the Web for this question. During your exciting interview at MIT, you find that they too seem to like distributed shared memory. They ask you the following question. In a distributed shared memory system using invalidate, a 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 holding the page in Read mode and P4 is holding it in Write mode and P3 is the owner

b. P4 is the owner and is holding the page in a Write mode

c. P1 and P2 are each holding the page in a Write mode, and P4 is the owner

d. 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 10 places listed in 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!).