CS425 Fall 2022 – Homework 2

(a.k.a. "Once upon a time in Distributed Hollywood")

Out: Sep 21, 2022. Due: 2 pm US Central on Oct 3, 2022 (It's a Monday!)

Topics: Key-value Stores, Time and Ordering (Lectures 9-12)

Some corrections made on Sep 28 2022 (in red below, and also on Piazza "HW2 Errata")

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 (e.g., timeline questions) 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**. **And on Gradescope please tag each page with the problem number!**
5. Homeworks will be **due at time and date noted above. 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 in the question, 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 on Piazza.

Prologue: You have just been made the technical head in a production company that is producing a new Hollywood movie. The movie is sure to be a blockbuster, with a lot of well-known actors and actresses hired to star in it. Amazingly many of them know distributed systems! You run into them every day on the set. Here is what ensues.

All characters and their actions used in this homework are meant to make the homework fun! Any resemblance of their actions or opinions to real events, or places, is purely coincidental. Any stories involving real actors or actresses are fictional.

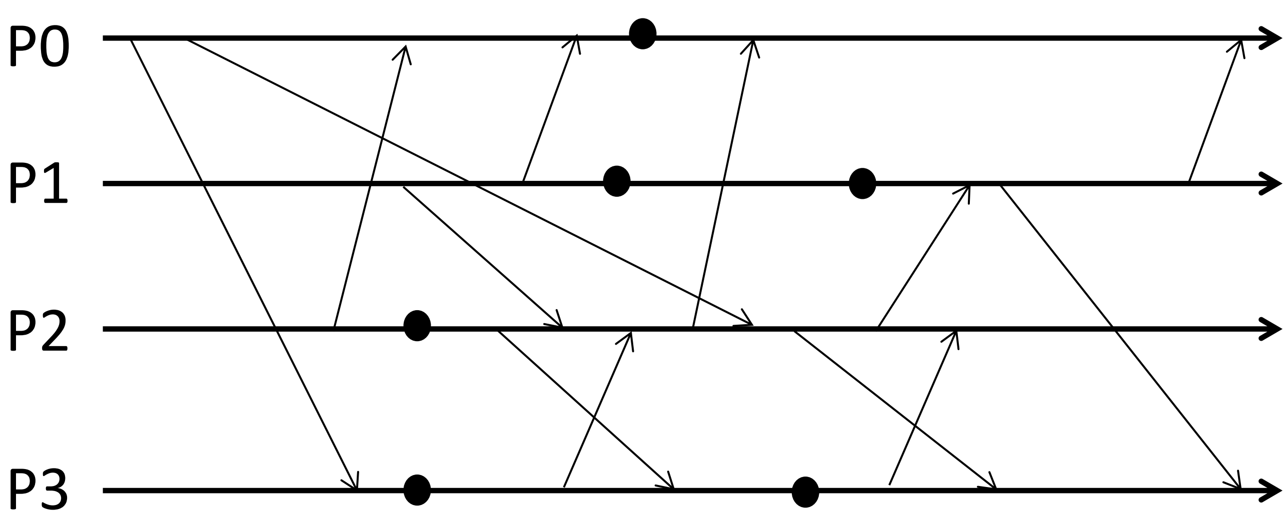
Problems:

1. One of the Producers, Leo Bloom, likes Bloom filters. But being a producer, he wants to create a new Bloom filter-based data structure. A (regular) Bloom filter's false positive rate is given as $\left(1 - e^{-\frac{kn}{m}}\right)^k$ where *k* is the number of hash functions, *n* is the size of the input set and *m* is the size of the Bloom filter in bits. Leo says that instead of using a single Bloom filter B with 2048 bits and 4 hash functions, his new datastructure, called Leo Bloom filter, uses 4 Bloom filters B1, B2, B3, and B4, each with 512 bits, and each using 4 hash functions (each hash function different from each other, and different from the above 4 hash functions). When checking for an item, it returns true only if the item is present in *all* of B1, B2, B3, and B4. When inserting an item it is inserted into *all* of B1, B2, B3, and B4. Which of the above two approaches—original using only B vs. Leo's Bloom filter using B1-B4—gives better false positive rates? Answer this for two cases: (1) when there are typically 4 elements inserted into the datastructure, (2) when there are typically 100 elements inserted into the datastructure. (We recommend though, that you solve the problem with the variables *k, n, m, etc.*, and then apply these values. But solving with only these two values of 4, 100, would be ok as well.)
2. Help, the next Squid Game challenge looks a bit tough (but really isn't). Don't get eliminated! In a system of N processes (N large enough, etc.), THREE quorum sets Q1, Q2, and Q3 are selected in an arbitrary manner, all of the same size M.
    a. What should this <u>minimum</u> value of M be so that at least one process belongs to ALL three of Q1, Q2, and Q3 (that is, the intersection of all three sets is non-null)?
    b. Can you repeat the problem if there are 4 quorum sets (Q1, Q2, Q3, Q4) and the requirement was that all four of them should intersect in at least one process?
    (Hint: start with 2 intersecting sets, and quantitatively reason/think about why it has to be N/2+1).
3. (For this question you can search resources on the Web.) One of the actresses, named Meryl, is consistently a good actress and consistently wins awards. It's no surprise that she is very interested when you tell her about consistency models. She asks you about the differences between linearizability, sequential

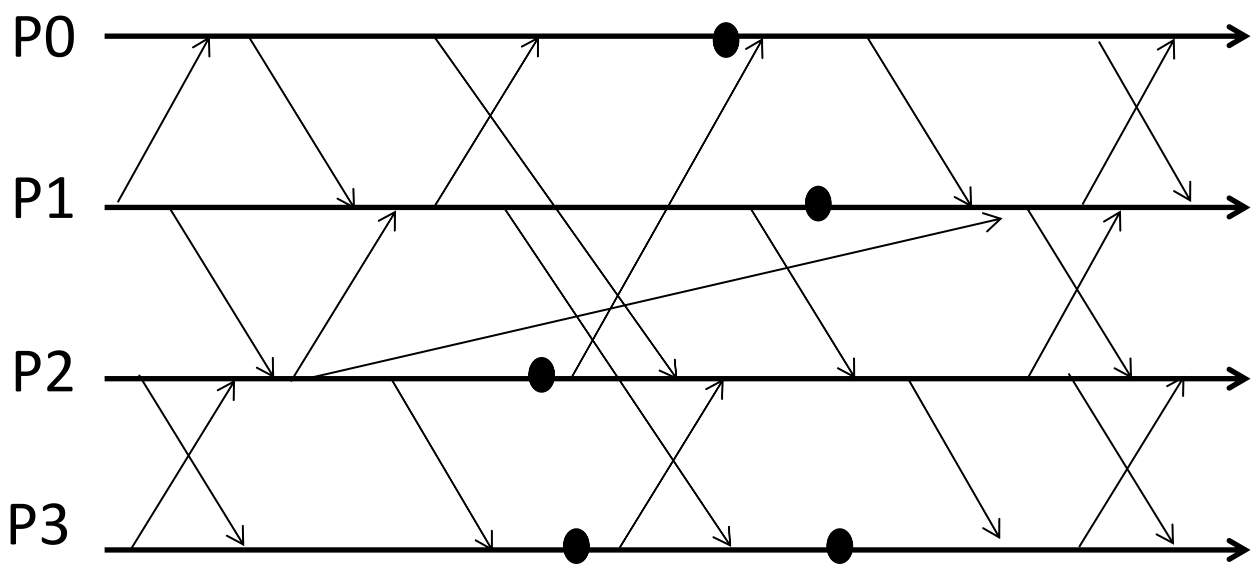
consistency, causal consistency, and eventual consistency (for key-value stores with get/put operations on keys).

a. Can you say briefly, and clearly what the differences are between the above models? (50 words or less)
b. Give an example (using 2-3 clients writing and reading objects), where, for a particular read, using each (not ~~one~~) of these models above gives a completely different return value. While you can search the Web to clarify differences between these models, you cannot borrow an example from the Web. Be concise.

4. To coordinate time between two teams in the real world and the "Upside Down" world, the actors decide to use Cristian's algorithm. One of the characters, oddly named "Thirteen", finds that the round-trip time for one round of synchronization messages is 11.11 ms. She would like to find the error in the run, and so she measures some minimum delays. On the client side, she finds that there is a delay of at least 0.66 ms for a packet to get from an application to the network interface, but she is unable to measure the delay of for the opposite path (network interface to application buffer). On the server side, she measures that the time to get from the application buffer to the network interface is 66 microseconds, but before she can measure the reverse path, Count Vecna arrives and so she has to scramble. What is the error, given the data just presented?
5. Spiderman and his doppelgangers are trying to figure out the communication among the different metaverses. They have the following timeline of messages exchanged, where each dot represents an instruction. Can you mark Lamport timestamps on each event? It is ok to print out this and hand-draw/write the timestamps (then scan or photograph your solution, and insert it into your solution doc).



6. Whoops, the simulanteous and sudden arrival of all of the concurrent villains (Doc Ock, Green Goblin, Electro, Sandman, and the others) means Lamport timestamps are unusable for distinguishing concurrent events from causal events. Can you mark vector timestamps for the same timeline as in the previous question? It is ok to print out this and hand-draw/write the timestamps (then scan or photograph your solution, and insert it into your solution doc).
7. The Squid game hosts found out all the communications among your 4 team members, and their next challenge is --- mark Lamport timestamps in the following figure! Um, try not to be "eliminated"! It is ok to print out this and hand-draw/write the timestamps (then scan or photograph your solution, and insert it into your solution doc).



8. Oops, you 4 survived, but now the next challenge is to mark vector timestamps on the timeline of the previous question. Can you do it correctly and not to be "eliminated"? It is ok to print out this and hand-draw/write the timestamps (then scan or photograph your solution, and insert it into your solution doc).
9. The teams at Paw Patrol and Peppa Pig have come up with a rather childish algorithm to replace Lamport timestamps. In their modified algorithm, called the Cartoon Algorithm, each process keeps a local FIFO counter (just like in the FIFO ordering). Let the FIFO counter for an event *e* be *F(e)*. The FIFO counter F at a process works as follows. F at a given process pi is initialized to 0. Whenever an event e (send, receive, instruction) occurs at the process pi, F is incremented by 1 and the resultant value of F is assigned to the event e as its FIFO timestamp F(e).

Note that each process maintains its own local FIFO counter. In the Lamport algorithm, instead of incrementing timestamps by +1 (on an instruction/send or in the equation for a receive), the new algorithm increments it instead by (*F(e)+10*). Is the Cartoon Algorithm correct, i.e., does it preserve causality? If so, give a formal proof. If not, show a clear counterexample.

10. The Mandalorian has been away for a while, and he left Baby Yoda (or Grogu) in charge of a little Cassandra key-value store cluster. Poor Baby Yoda doesn't know programming. Because the Mandalorian has been away for a while, compaction has become disabled at every node in his Cassandra key-value store (for a while), while writes and reads have kept on coming in the meantime. Can you explain to Baby Yoda how this disabling of compaction affects:
    a. write latency at a single replica node
    b. read latency at a single replica node
    c. correctness of writes at a single replica node
    d. correctness of reads at a single replica node
    e. correctness of writes across the cluster (for a given consistency level)
    f. correctness of reads across the cluster (for a given consistency level)

====== END OF HOMEWORK 2 =====