Write up answers to the following sets of numbered questions.
Share your answers in a GoogleDoc with profchadd@gmail.com by 1pm on Nov 21.

Be prepared in class on Nov 21 to discuss the questions below.

0 Write definitions for any terms you run across and are not familiar with.

1 What leads to "data inflation" in a Relation DBMS?

2 What, do you think, is the typical use case of MySQL in the LAMP stack? What does the workload look like? How does this compare to the workload discussed in the Jacobs article?

3 Jacobs talks a good deal about temporal ordering. What does this mean and why is it important. Could you force a MySQL table to be stored in temporal order?

4 Jacobs says: "The prevailing database model today, however, is the relational database, and this model explicitly ignores the ordering of rows in tables." Why is this? What benefit do we get from this?

5 Why is sequential disk access 5 orders of magnitude faster than random disk access but sequential RAM access only 1 order of magnitude faster than random RAM access? Why is this important if, as Jacobs says on page 38, "In fact, our table of "all the people in the world" will fit in the memory of a single, $15K Dell server with 128GB RAM."

6 Is Jacobs looking for a hardware or software solution to the problem of big data?

7 What does Jacobs have to say about normalization with respect to big data?

8 What is Jacobs point about the Zero-one-infinity rule? Is he saying programmers are lazy or stupid or both or something else? Why do most versions of Microsoft Excel limit you to 65,536 rows?

9 Who, which customers, face the problems Jacobs is talking about? Who will face these problems in 10 or 20 years?

10 As a wealthy alumna or alumnus of Pacific University, what research area are you going to donate money to in order to help solve Jacobs's problem?

11 Does Stonebraker offer any solutions to Jacobs?