Each question is worth 10 points. You may work with one partner and sign both of your names to your paper.

1. The Undo, No Redo algorithm has the following description: Transactions First transfer before-images of each page to the audit trail and then put the after-images in the database. Suppose we changed it so that we transfer the after-image to the database before putting the before-image to the audit trail. Will this be correct? Prove your answer.

2. Suppose that a transaction manager has done the following steps in the two phase commit protocol for a transaction T:
   (a) It has asked all servers whether they are willing to commit and they all responded affirmatively.
   (b) It has told all servers to commit, but has not waited for a response.

Does the transaction manager still need to keep a record of transaction T? If not, why not? If so, for how long?

3. The available copies algorithm will abort a transaction if any site that the transaction read from or wrote to while executing has failed between that time and commit time. Would this still be necessary if there were a separate lock manager with an independent failure mode. How would you then modify the available copies commit rules given in the class notes?

4. Suppose we are trying to decide whether to put a non-clustering index on attribute B for relation R to support equality selections on B. The relation has 10 million records. Each page can store 10 records. There are 10,000 different values of B. A sequential scan will fetch 10 pages per read. Explain why you would or would not want to include a non-clustering index.

5. A company with 100,000 employees has an employee relation including name and salary. The accounting department issues queries concerning the statistics of salaries (their total, their average, etc.) several times a day, so you have decided to construct a redundant relation holding that information. That information must be updated whenever employees get raises, are hired, are demoted, or are fired. Assume the redundant relation is worthwhile now. Decide which of the following events would cause you to reconsider its usefulness. Say why.
   a. Your locking mechanism changes from page-level locking to record-level locking.
   b. All secretarial employees are to receive a 10% salary increase.
   c. The four senior vice-presidents are to receive a 20% salary increase.
   d. You decide to add a non-clustering index on salary to employee.

6. Suppose that each of relations R, S, T, V, W has A as its only key. Which of the following queries may output a different number of records if DISTINCT is removed? Prove your answer.

   a. SELECT DISTINCT R.A, S.A
      FROM R, S
      WHERE R.B = S.C

   b. SELECT DISTINCT R.A
      FROM R, S
WHERE R.B = S.C

c. SELECT DISTINCT R.A
FROM R, S
WHERE R.B = S.A

d. SELECT DISTINCT R.A
FROM R, S, T, V, W
WHERE R.B = S.A
AND R.D = T.D
AND R.C = V.A
AND T.A = S.B
AND W.A = S.D

e. SELECT DISTINCT R.A
FROM R, S, T
WHERE R.B = T.A
AND R.B = S.C

7. Suppose that the following four transactions are the only ones that execute during some interval (R stands for read, W for write, and different letter arguments represent different data items).

T1: R(A) W(B) R(C) W(D)
T2: W(A) W(C)
T3: R(D) R(B)
T4: R(A) R(E)

a. What is the finest chopping of T1 assuming that its reads and writes cannot be reordered? Show the chopping.

b. What is the finest chopping of T1 assuming that its reads and writes can be reordered? Show the reordering, then the chopping.

c. Can any other transactions be chopped?

8. Suppose you were designing a concurrency control mechanism for an office system. In this system, users may read the database, modify the relation in real time (by modifying records explicitly) and then write the result back into the database. This would be a single transaction and could take long. For this reason, we want a mechanism that penalizes particularly slow users (e.g. the ones that go out for a coffee break). Decide on a concurrency control mechanism for this environment. Why would two phase locking not be appropriate? How about an optimistic protocol? There is no single correct answer, but your reasons must be backed up by provable properties of the protocols.