1. **Explain the actions taken in the two phase commit protocol after restart if the coordinator or subordinate nodes fail.**
   - Restart after a failure at a node.

   - If we have a commit or abort log rec for Xact T, but not an end rec, must redo/undo T.
     - If this node is the coordinator for T, keep sending commit/abort msgs to subs until acks received.

   - If we have a prepare log rec for Xact T, but not commit/abort, this node is a subordinate for T.
     - Repeatedly contact the coordinator to find status of T, then write commit/abort log rec; redo/undo T; and write end log rec.

   - If we don't have even a prepare log rec for T, unilaterally abort and undo T.
     - This site may be coordinator! If so, subs may send msgs.

Reference: Lecture Notes

2. **Multi-version concurrency control**
   (a) Let T1 read and T2 read and update data item D. If transaction T3 updates only D, provide a schedule where the multi-version concurrency control method will not restart T3.
   - T1 reads D, T2 reads D, T2 updates D, T3 updates D.

   (b) Again, assume that T1 reads and T2 and T3 read and update only data item D. Provide a schedule where the multi-version concurrency control method will restart T2.
   - T1 reads D, T3 reads D, T3 updates D, T2 reads D, T2 updates D.

3. **Quorum reading**

Let the number of copies of a data item in a cluster, N, be 10 and the number of nodes that participate in a successful write, W, be 5.

(a) What is minimum number of nodes that should participate in a successful read, R?

- 6**.**

(b) Now, consider a sloppy quorum with the same values for N and W. What is the minimum number of nodes that should participate in a successful read, R?

- 1

4. **Eventual consistency and vector clocks (2 points)**
   Each row in the following table shows vector clocks of different copies of the same data on a system with three nodes SX, SY , and SZ. Explain if the copies in each row have a conflict.

- Vector clocks track causality between versions therefore the first row do not have any conflicts, since the version of Sx and Sy data items are of same version but the second row data items have a conflict because of newer version of Sy data.

5. PageRank

- Page without successors has nowhere to send its importance and those are the pages with PageRank equal to 0 which are also called dead ends. In the Figure.1 there is no such page which is not forwarding its importance to other pages therefore they all are greater than zero. Their relative PageRank values in the graph are:
    - Va = ½ Vf
    - Vb = ½ Va
    - Vc = ½ Vb
    - Vd = ½ Vc
    - Ve = ½ Vd
    - Vf = ½ Ve
    - Vm = ½ Ve
    - Vg = ½ Vm + ½ Vl
    - Vh = ½ Vg
    - Vi = ½ Vh
    - Vj = ½ Vi
    - Vk = ½ Vj
    - Vl = ½ Vk

6. MangoDB

1. **Write a query that returns the cuisine type of the restaurant named The Dead Rabbit**
    - db.restaurants.find({“name”: “The Dead Rabbit”})
2. **Write a query to create an index on the name attribute of the restaurants.**
    - db.restaurants.createIndex({“name”: “text”})
3. **Write a query that uses the index created in part b to return all the restaurants containing the term Rabbit in their name.**
    - db.restaurants.find({ $text: {$search: “Rabbit”}})
4. **Write an aggregate query to show total count of restaurants in each borough.**
    - db.restaurants.aggregate([{$group: {“_id”: “$borough”, “count”: {$sum: 1}}}]);