

FINAL EXAM (12/6/96)  
COP 6617 Distributed System Design

1. (40 pts)

- Suppose the dynamic voting algorithm is applied to a network of five sites, the network was partitioned into  $\{A, B, C\}$  and  $\{D, E\}$  after six updates. Two updates have been performed at  $\{A, B, C\}$  before another partition occurs:

$$\{A : (8, 3), D : (6, 5)\} \quad \{B : (8, 3), C : (8, 3), E : (6, 5)\}$$

Clearly,  $\{B, C, E\}$  forms a majority of the latest version (version 8), therefore further updates are possible. Site  $E$  can catch up and gets a copy of version 8. Can  $D$  catch up at  $\{A, D\}$ ? and Why? If your answer is negative, provide a list of potential problems if  $D$  is allowed to catch up.

- John proposes a voting method called *voting-with-witness* that replaces some of the replicas by *witnesses*. Witnesses are copies that contain only the version number but no data. The witnesses are assigned votes and will cast them when they receive voting requests. Although the witnesses do not maintain data, they can testify about the validity of the value provided by some other replica. How should a witness react when it receives a read quorum request? and how about a write quorum request? Discuss pros and cons of this method.

2. (40 pts) Wu and Fernandez (1992) gave the following safe and unsafe node definition: A nonfaulty node is unsafe if and only if either of the following conditions is true: (a) There are two faulty neighbors, or (b) there are at least three unsafe or faulty neighbors. Consider a 4-cube with faulty nodes 0100, 0011, 0101, 1110, and 1111.

- Find out the safety status (safe or unsafe) of every node. Use the figure provided.
- Prove that if the source node is safe, then there is a healthy Hamming distance path from the source to any destination node in any given cube.

(Hint: Prove by induction based the distance between the source and the destination. Use the fact that there are  $k$  node-disjoint paths between two nodes that are  $k$  distance apart.)

3. (20 pts) Determine the deadlock status for each of the following wait-for graphs.

- AND condition is used.
- OR condition is used.

**EXTRA POINTS:**

4. (10 pts) One potential problem of voting is that the number of nodes required in a quorum for performing an operation increases linearly with the number of replicas. One way to reduce quorum is by organizing a hierarchical voting structure. Basically, this structure forms a multi-level tree and a quorum is associated with each level. The physical copies of the object are stored only at the leaves of the tree. The same quorum requirement for read ( $r$ ) and write ( $w$ ) is applied to each level. read (write) at level  $i$  depends on the corresponding groups at level  $i - 1$  (we assume that the leaves have the lowest levels and the root of the tree has the highest level). That is, a read (write) at level  $i$  is possible if all its children form a read (write) quorum at level  $i - 1$ .

Consider a 3-level tree in which each intermediate node has exactly five children. A total of 25 leaves are in the system. Assume that read quorum is 2 and write quorum is 4 for each level. Show that condition  $r + w > v$  is not necessary in the hierarchical voting. What are additional overheads in implementing the hierarchical voting?