1. Fault Collapsing: For the circuit shown below with the node names shown, perform the following. (a) Give the list of collapsed faults using local equivalence from input to output pass. (b) Give the list of collapsed faults using dominance from output to input pass. (c) If we were to use the Checkpoint Theorem without any collapsing, what faults need to tested? Give the list of these faults.

![Circuit Diagram](image)

2. Pin Faults vs. Internal Faults: Following XOR circuit can be tested for Pin Faults on A, B using just two vectors: 00 and 11. First produce a list of collapsed faults using equivalence relation. Then list all faults that the vector 00 detects from this list. Repeat this for vector 11. List all faults that were missed by these two test vectors.

![Circuit Diagram](image)

3. From the Handout slide 26, the bridge detection probability with two test vectors (for G/0 and G/1) is 75% assuming 50% Probability of 1’s and 0’s on line H, assuming the bridge fault to be “H dominates G”. Re-compute the bridge detection probability assuming the Signal Probability on line H is 20%. Generalize this result for Signal Probability being p and the number of times the node G tested is n. (Signal Probability is defined as the probability that the logic value is 1).

4. A logic simulator encodes 0, 1 and X by two bits as v1v2: 00 for 0, 11 for 1 and 01 for X. Express Boolean operations AND, OR, NOT and XOR in terms of two bits v1 and v2. From these, derive C-language operations for bit-by-bit parallel word operations for AND, OR, NOT and XOR. See for example Table II in PROOFS paper for a different code. The input variables A and B are coded in four 32-bit words A1, A2 and B1, B2 and the output C is coded in words C1, C2.