

Due March 31

COURSE WEBSITE: www.cs.umd.edu/gasarch/858/858.html
(there is a tilde before the “g” in “gasarch”)

1. (20 points) Write your name clearly. Write which HW this is on top of your paper clearly. Staple the HW so its not loose pages. Where and when will the final be?
2. (40 points) In this problem we present two attempts at zero-knowledge protocols for 3-COL. For each one argue informally why it is NOT a zero-knowledge protocol for 3-COL. Assume the graphs vertices are $\{1, \dots, n\}$. Assume an honest Verifier.
 - (a) Prover sends over commit-bits to commit to a 3-coloring of the graph. Verifier picks TWO edges at random and demands that the colors of both endpoints be revealed. Prover reveals the endpoints. If they are colored differently then Verifier says YES else he says NO. REPEAT n times. If the Verifier always says YES then he ACCEPTS, else he REJECTS.
 - (b) Prover sends over commit-bits to commit to a 3-coloring of the graph. Verifier picks an edge where both endpoints are vertices which are $\leq \sqrt{n}$ and demands that the colors of both endpoints be revealed. Prover reveals such. If they are colored differently then the Verifier says YES, else he says NO. REPEAT the above n times. If the Verifier always says YES then he ACCEPTS else he REJECTS.
3. (40 points) For the following protocol for HAMILTONIAN CYCLE state clearly if it IS or IS NOT a Zero-knowledge protocol, and justify your answer informally.
 - (a) Prover takes the entire graph, permutes the vertex labels randomly, forms the adjacency matrix of the graph, and sends bit-commits for the n^2 entries in the adjacency matrix.
 - (b) Verifier flips a coin.
 - i. If its HEADS then he demands that ALL bit commits be revealed and an isomorphism of the revealed graph to G be shown. The Prover complies. The Verifier checks that it really is an isomorphism. If it is then he says YES, else he says NO.
 - ii. If its TAILS then the Verifier demands that a Ham Cycle be shown- so the Prover then reveals a permutation of $1, \dots, n$ which we will call a_1, \dots, a_n and also reveals the edges (a_1, a_2) and (a_2, a_3) etc. The Prover complies. The Verifier checks that it really is a Ham Cycle. If it is then he says YES, else he says NO.
 - (c) REPEAT the above n times. If the Verifiers always says YES then he ACCEPTS, else he REJECTS.