U.C. Berkeley — CS172: Automata, Computability and Complexity Practice Final
Professor Luca Trevisan 5/3/2007

Practice Final

1. (Sipser 1.45) Let $A/B = \{w \mid wx \in A \text{ for some } x \in B\}$. Show that if A is regular and B is any language, then A/B is regular.

2. Let M be a 1-tape Turing machine with q states, and let w be a string of length n. Prove that if on input w the machine M does not move its head left in the first $n + q + 1$ steps, then it *never* moves its head left on this input.

3. A boolean formula is said to be in Monotone 2-CNF if it is the conjunction of clauses, each of which has exactly 2 literals and all the literals in the formula are positive (i.e. no negations). Note that such a formula can be easily satisfied by setting all variables to `true`.

   Consider the following version of the satisfiability problem for Monotone 2-CNF formulas:

$$k - MON - 2SAT = \{\langle \phi, k\rangle \mid \phi \text{ is in Monotone 2-CNF and can be satisfied}\\ \text{by setting at most } k \text{ variables to true}\}$$

   Prove that k-MON-2SAT is **NP**-complete.

4. Define

$$\textsc{Cycle-Length} \;=\; \{\langle G, c\rangle \mid 3 \le c \le |V(G)|, G \text{ is a directed graph and}\\ \text{the length of the shortest cycle in } G \text{ is } c.\}$$

   Prove that Cycle-Length is **NL**-complete.

5. Consider the language

$$EQ_{NFA} = \{\langle N, N'\rangle \mid N, N' \text{ are NFAs with the same alphabet and } L(N) = L(N')\}$$

   Show that $EQ_{NFA} \in \textbf{PSPACE}$.
   (*Hint:* Can you convert this to an appropriate reachability problem?)

6. We define the class Universal Simulator Perfect Zero-Knowledge (USPZK) as the class of zero knowledge protocols for which there is a single universal simulator U, which given the input to the protocol and the code of the any verifier, simulates the verifier's view of the interaction. Sipser gives the following interactive protocol for Graph Non-Ispmorphism, which is is actually in Honest Verifier Perfect Zero Knowledge:

   Input: Two graphs G_1 and G_2.
   Verifier: Picks a random $i \in \{1, 2\}$ and a random permutation π. Sends $H = \pi(G_i)$.
   Prover: Sends i i.e. identifies if H is a permutated copy of G_1 or G_2.

   Prove that if the above protocol is in USPZK i.e. there exists a single universal simulator for all verifiers (not just honest ones), then there is a randomized polynomial time algorithm for Graph Isomorphism.