CSE 531: Complexity Theory
Winter 2007
SAMPLE FINAL EXAM

Instructions: Attempt all questions. The exam is for a maximum of 150 points. It has **six** questions, and you have 1 hour and 50 minutes to answer them. You may use without proof any of the theorems we have proved in class or which are proved in the textbook. This is an open book/notes exam, but reference to any "outside" sources is not allowed.

---

1. For each of the following assertions, state whether they are True, False, or Open according to our current state of knowledge of complexity theory, as described in class. You do *not* have to justify your answer choice. $(3 \times 10 = 30$ points)

   (a) 3SAT $\leq_P$ *TQBF*

   (b) EXPSPACE contains all decidable languages.

   (c) NL $\neq$ PSPACE

   (d) P $=$ NP $\cap$ coNP

   (e) EXP $\subseteq$ P/poly

   (f) BPP $=$ PSPACE

   (g) *HAMPATH* $\in$ coNP

   (h) $2SAT \leq_P CLIQUE$

   (i) $CLIQUE \leq_P 2SAT$

   (j) P contains all context-free languages.

2. (25 points) Let $EQ_{\text{BP}} = \{\langle B_1, B_2\rangle \mid B_1$ and B_2 are branching programs that compute the same Boolean function$\}$. (This is similar to the language we studied in class but with the read-once restriction on the branching programs removed.) Prove that EQ_{BP} is coNP-complete.

3. (25 points) A directed cycle in a directed graph $G = (V, E)$ is a sequence of k distinct nodes $v_1, \ldots, v_k \in V$ for some $k \geq 2$ where there is a directed edge from v_i to v_{i+1} for $1 \leq i < k$ and from v_k to v_1. Define the language

   $DAG = \{\langle G\rangle \mid G$ is a directed acyclic graph, i.e. a directed graph that has **no** directed cycle$\}$.

   Prove that *DAG* is NL-complete.

4. (25 points) In this problem, we let M be a deterministic Turing machine, w a string, i and j *binary* integers, and $\alpha \in Q \cup \Gamma$ where Q is the set of states of M and Γ is its tape alphabet.

   Let $A = \{\langle M, w, i, j, \alpha\rangle \mid \alpha$ is the i'th symbol of the configuration after the j'th step of the computation of M on input $w\}$.

   Prove that A cannot be decided in polynomial time.

5. (25 points) In this problem, we define and study a complexity class called ZPP which will correspond to randomized algorithms that never err, but could sometimes declare a failure.

   Formally, ZPP is defined as follows. A language L belongs to ZPP iff there exists a probabilistic *polynomial time* Turing machine M such that for every input x to M the following properties hold:

   (i) M halts (in time polynomial in $|x|$) and gives one of three possible outputs: Yes, No, Fail.

   (ii) If M outputs Yes, then $x \in L$ and if M outputs No, then $x \notin L$. In other words, M never errs when it doesn't output Fail.

   (iii) The probability that M outputs Fail, over its coin tosses, is at most $1/3$.

   Prove that ZPP $=$ RP $\cap$ coRP, i.e., a language belongs to ZPP if and only if it belongs to both RP and coRP.

6. (20 points) Define the language

$$\text{QNR} = \{(x, a) \mid a \text{ is a quadratic non-residue modulo } x\} .$$

   Give an interative protocol for QNR along the lines of the interactive protocol for graph non-isomorphism. (Note: we know that QNR $\in$ BPP, and thus the verifier can determine membership without the aid of a prover. But you must not appeal to this result.)