CS 221: Computational Complexity	**Prof. Salil Vadhan**
Problem Set 2	
Assigned: Fri. Feb. 21, 2014	Due: Fri. Mar. 7, 2014 (5 PM sharp)

- You must *type* your solutions. LaTeX, Microsoft Word, and plain ascii are all acceptable. Submit your solutions *via email* to `cs221-hw@seas.harvard.edu`. If you use LaTeX, please submit both the compiled file (`.pdf`) and the source (`.tex`). Please name your files `PS2-yourlastname.*`.
- Strive for clarity and conciseness in your solutions, emphasizing the main ideas over low-level details. Do not despair if you cannot solve all the problems! Difficult problems are included to stimulate your thinking and for your enjoyment, not to overwork you. *'ed problems are extra credit.

Problem 1. (Boolean Matrix Multiplication) Given two $n \times n$ matrices A, B with Boolean entries, their *Boolean product* $A \cdot B$ is the $n \times n$ matrix C such that

$$C_{ij} = \bigvee_{k=1}^{n} (a_{ik} \wedge b_{kj})$$

1. Give a logspace algorithm to compute $A \cdot B$ given A and B.
2. Given an $n \times n$ matrix A and $k \in \mathbb{N}$, describe an $O((\log n)(\log k))$-space algorithm to compute A^k, the k'th Boolean power of A. (Hint: first consider k that is a power of 2)
3. Give another proof of Savitch's Theorem using Item 2.

Problem 2. (NL-completeness) Prove that 2SAT is **NL**-complete. (Hint: To prove that it is in **NL**, show that the satisfiability of ϕ can be determined from the answers to polynomially many PATH questions involving the directed graph G_ϕ that includes edges $(\neg x, y)$ and $(\neg y, x)$ for every clause $(x \vee y)$ in ϕ.)

Problem 3. (Complete Problems for PH) Show that if the **PH** has a complete problem, then the polynomial hierarchy collapses (i.e. $\mathbf{PH} = \mathbf{\Sigma}_k^p$ for some k.)

Problem 4. (More Time-Space Tradeoffs for Satisfiability) The time-space tradeoffs done in class optimize the space lower bound ($n^{1-\epsilon}$) while giving a relatively weak time lower bound ($n^{1+o(1)}$). On this problem, you'll do the opposite, giving a time lower bound of $n^{1.41}$ while giving a weaker space lower bound ($n^{o(1)}$).

Do not worry about constructibility of the time and space bounds on this problem.

1. Show that for every $T(n) \geq n^2$, $\mathbf{TISP}(T, T^{o(1)}) \subseteq \Sigma_2\mathbf{TIME}(T^{1/2+o(1)})$.
2. Use the above to prove that SAT $\notin \mathbf{TISP}(n^c, n^{o(1)})$ for any $c < \sqrt{2}$. (Hint: Use a NONdeterministic-time Hierarchy Theorem.)

Problem 5. (regular expression problems) Consider regular expressions R with concatenation, union, Kleene star, and exponentiation. Recall that in class we showed the language $\text{ALL}_{\text{REX}\uparrow} = \{R : L(R) = \Sigma^*\}$ is **EXPSPACE**-complete. Here we classify the complexity of variants of this problem.

1. Show that if we do not allow exponentiation, the problem becomes **PSPACE**-complete.
2. Show that the equivalence problem $\{(R_1, R_2) : L(R_1) = L(R_2)\}$ where R_1 and R_2 are regular expressions with exponentiation but no Kleene stars is **coNEXP**-complete.