Name: _____ TF: _____

Harvard University
Computer Science 121
Professor Salil Vadhan

Final Examination
Tuesday, January 20, 2009

You should answer the Problem 3 directly on this sheet. Please write your name on the exam and return it with the blue book. Do the rest of the problems in the blue book.

The points total 100. On the last page, there are a couple of fun extra credit problems that should only be attempted after you have completed the rest of the exam. You may use any result proven in lecture, Sipser, or the homework, provided that you state it clearly.

PROBLEM 1 (3 points)

Write a regular expression for the language $L = \{a^{3i}b^{5j} : i, j \in \mathbb{N}\}$.

PROBLEM 2 (4+4+4 points)

For each of the following conditions, either state that no language satisfies it, or give an example of a language that satisfies it. No justifications are necessary in either case.
(A) L is decidable but not recognized by any PDA.
(B) L is Turing-recognizable but not decidable.
(C) L is in P but not in NP.

PROBLEM 3 (21 points)

Fill the blank entries of the following table with `YES`, `NO`, or `?? (CURRENTLY UNKNOWN)`. No explanations needed. M stands for a Turing machine, D is a DFA, and $\Sigma = \{a, b\}$.

Language:	regular	context-free	decidable	recognizable	P	NP
$\{a,b\}^* - \{a^n b^n : n \in \mathbb{N}\}$						
$\{\langle M\rangle : L(M)$ is context-free$\}$				X		
$\{a^n : n$ is a prime number$\}$						
$\{\langle D, w\rangle : D$ is a DFA and $w \in L(D)\}$	X	X				

(TURN OVER!)

PROBLEM 4 (4+4+4+4+4 points)

Answer TRUE or FALSE. Justify your answers in a sentence or two.
(A) $n/2 = O(\log_2(2^5 n^4))$
(B) If L is regular, then $L' = \{x \in L : |x| > 2009\}$ is also regular.
(C) Every language is countable.
(D) The number of regular languages over alphabet $\{a, b\}$ is countably infinite.
(E) If a Turing-recognizable language contains exactly one string of length n for every n, then it is decidable.

PROBLEM 5 (10+5 points)

(A) Give an algorithm to decide whether a given NFA N recognizes the language Σ^*. (Hint: first consider how to test whether an NFA recognizes $\emptyset$.) Briefly explain why your algorithm works.
(B) Does your algorithm run in polynomial time? Briefly justify your answer.

PROBLEM 6 (10+5 points)

(A) Give a context-free grammar over the alphabet $\{x, \neg, \vee, \wedge, (,)\}$ for the set of all Boolean formulas in Conjunctive Normal Form, using x, xx, xxx, $xxxx$, etc. for the names of the variables.
(B) Do you think it likely that there is a context-free grammar for the set of all *satisfiable* Boolean formulas? Explain why or why not.

PROBLEM 7 (4+10 points)

(A) Define NP-complete.
(B) Consider the following two languages, whose instances are of the form $\langle G, k\rangle$ where G is an undirected graph and k is a natural number:

- CLIQUE consists of all $\langle G, k\rangle$ for which there is a set S of k vertices in such that every pair of vertices in S are connected by an edge in G.
- INDEPENDENT SET consists of all $\langle G, k\rangle$ for which G has a set S of k vertices such that no pair of vertices in S is connected by an edge in G.

In lecture, we proved that CLIQUE is NP-complete. Show that INDEPENDENT SET is also NP-complete.

Extra Credit Problems

PROBLEM 8 (3 points)

On your first day on your new job at SmartCompilers.com your boss assigns you the task of adding the following new feature to the company's development suite: Analyze the source code of a function, and verify that no array will ever be accessed out of bounds. Output TRUE if the code is safe in this respect, and FALSE if for some inputs, the function will access an array out of bounds.

For example, the following C program should fail the verification. (Explanatory comments are included after the %'s for those not familiar with C.)

```
void sloppycode(int k) {  % sloppycode takes an integer k as input and has no output
      int x[10];                          % initialize x to be an array of length 10
      for (int i = 0; i < k; i++)                              % loop from i = 0 to k − 1
            { x[i] = i; }                          % set the i'th entry of x equal to i
}
```

Program *sloppycode* should fail, because on input $k = 15$, the array x will be accessed at position 11 (i.e., $x[11] = 11$;) which is outside its declared range of 10.

Demonstrate that your boss' request is unreasonable by showing that the above verification problem is undecidable.

In your answer, you may assume that the behavior of the function is fully determined by its source code an inputs (i.e., it does not make any outside calls to libraries or other functions). Also assume for simplicity that the only variables a function uses are (possibly unbounded) integers, and integer arrays. Additionally, assume that you have a way to simulate C programs by Turing machines and vice-versa.

PROBLEM 9 (2 points)

Answer at least one of the following:
(A) What came first? Turing Machines or the Halting Problem?
(B) Why did the DFA cross the road?
(C) Why did the TM cross the road?
(D) What did the DFA say to the TM when they had both crossed the road?