Harvard University
Computer Science 121
Final Exam — December 17, 2012

This is a closed-book examination. You may use any result from lecture, Sipser, or the problem sets, as long as you quote it clearly. The alphabet is $\Sigma = \{a, b\}$ and L denotes a language, except where otherwise stated. You may give high-level descriptions of Turing machines, except where otherwise specified.

You have 180 minutes. The problems total 155 points. Use pen and write your name on all bluebooks you use. Good luck!

PROBLEM 1 (5 points)

Write a regular expression that generates the set of strings over alphabet $\{a, b\}$ in which every b is both preceded by an a and followed by an a. No justification is necessary.

PROBLEM 2 (20 points)

Fill the blank entries of the following table with `YES`, `NO`, or `??` ("currently unknown"). No explanations are needed. You can either copy the table into your bluebook or turn in this sheet with your bluebook.

Language:	regular	context-free	decidable	r.e.	P	NP
$\{\langle N \rangle : N$ is an NFA that accepts $abbababa\}$						
$\{w : w$ contains $abbababa\}$						
$\{\langle M \rangle : M$ is a TM that accepts $abbababa\}$						
$\{\langle \varphi \rangle : \varphi$ is a satisfiable boolean formula$\}$						

PROBLEM 3 (5 points)

Show that the language $L = \{w \in \{a, b\}^* : |w|$ is odd and the middle symbol of w is $a\}$ is context free. (If you show this by construction, you do not need to prove the correctness of your construction.)

PROBLEM 4 (10 points)

Let $L = \{wa^n : w \in \{a, b\}^*, n =$ the number of a's in $w\}$. Is L regular? Prove your answer.

PROBLEM 5 (10+15 points)

Define a *Stay-Still TM (SSTM)* to be like an ordinary TM, but with an additional possibility of staying still in a transition (instead of being required to move left or right in each step).
(A) Show, using implementation-level descriptions, that SSTMs are equivalent in power to TMs.
(B) Prove that the language $L = \{\langle M\rangle : M \text{ is an SSTM that stays still in at least one step when run on } \varepsilon\}$ is undecidable.

PROBLEM 6 (10+10 points)

For an integer N, let let $\langle N\rangle_B$ denote the binary representation of N (over alphabet $\{0, 1\}$) and $\langle N\rangle_U$ denote the unary representation of N (i.e. a string of N ones). For each of the following two functions, state whether it is computable in polynomial time and justify your answer. You may quote without proof basic facts about the complexity of arithmetic operations (e.g. addition and multiplication).
(A) $\text{BinaryToUnary}(\langle N\rangle_B) = \langle N\rangle_U$.
(B) $\text{UnaryToBinary}(\langle N\rangle_U) = \langle N\rangle_B$.

PROBLEM 7 (20 points)

SetCover is the following problem: given a finite universe U, a family of subsets $S_1, \ldots, S_n \subseteq U$, and a number $\ell \in \mathbb{N}$, are there ℓ of the sets S_i whose union equals the entire universe U?

Prove that SetCover is NP-complete. (Hint: recall the NP-complete problem VertexCover: given a graph G and a number k, determine whether there is a set of k vertices that includes an endpoint of every edge in G.)

PROBLEM 8 (10*5=50 points)

Determine whether the following statements are true or false. Justify your answers (a sentence or two per part should suffice).
(A) $\{\varepsilon\}^* = \emptyset$.
(B) The intersection of a regular language and a context-free language is always regular.
(C) If L is a context-free language, then $L \in$ co-NP.
(D) Every Turing machine recognizes some language.
(E) The Church-Turing Thesis has been mathematically proven.
(F) If 3-SAT $\in$ P, then the Travelling Salesman Problem is in P.
(G) If $M = (Q, \Sigma, \delta, q_0, F)$ is a DFA and $M' = (Q, \Sigma, \delta, q_0, \overline{F})$, where $\overline{F} = Q - F$ is the complement of F, then it is necessarily the case that $L(M') = \overline{L(M)}$.
(H) If $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ is a PDA and $M' = (Q, \Sigma, \Gamma, \delta, q_0, \overline{F})$, where $\overline{F} = Q - F$ is the complement of F, then it is necessarily the case that $L(M') = \overline{L(M)}$.
(I) $n^3 + 3n = O(5n^2)$.
(J) $2^n = \omega(5n^2)$.

PROBLEM 9 (CHALLENGE! 2 extra credit points)

Show that there is no "hardest" language: for every language L, there is a language L' that cannot be reduced to L (via a mapping reduction).