Computer Science 456/656 UNLV Fall 2001
Final Examination, December 11, 2001

Name:_____

No books, notes, or scratch paper. Use pen or pencil, any color. Use the rest of this page and the backs of the pages for scratch paper. If you need more scratch paper, it will be provided.

The entire test is 200 points.

1. A True or False. [5 points each]

   (a) _____ Every subset of a regular language is regular.

   (b) _____ Let $F(0) = 1$, and let $F(n) = 2^{F(n-1)}$ for $n > 0$. Then F is Turing-computable.

   (c) _____ Every language which is accepted by some non-deterministic machine is accepted by some deterministic machine.

   (d) _____ There cannot exist any computer program that can decide whether any two C++ programs are equivalent.

   (e) _____ An undecidable language is necessarily $\mathcal{NP}$-complete.

   (f) _____ Every Chomsky normal form grammar is equivalent to some unambiguous grammar.

   (g) _____ Every context-free language is in the class $\mathcal{P}$-TIME.

   (h) _____ Every function that can be mathematically defined is Turing computable.

   (i) _____ The language of all binary strings which are the binary numerals for multiples of 23 is regular.

   (j) _____ The language of all binary strings which are the binary numerals for prime numbers is context-free.

   (k) _____ Every bounded function from integers to integers is Turing-computable. (We say that f is *bounded* if there is some B such that $|f(n)| \leq B$ for all n.)

   (l) _____ The language of all palindromes over $\{0, 1\}$ is inherently ambiguous.

   (m) _____ Every context-free language over the unary alphabet $\{1\}$ is regular.

2. [15 points] Draw a minimal DFA which accepts the language of all strings over $\{a, b, c\}$ which do not contain the substring $abaa$.

3. [20 points] Give a Chomsky normal form grammar for the language of all palindromes of odd length over the alphabet $\{0, 1\}$.

4. [10 points] Give one example of a language which is undecidable. (Do not attempt to prove that it is undecidable. You may refer to the language by its standard name, such as, "The boolean satisfiability problem.")

5. [10 points] Give one example of a language which is $\mathcal{NP}$-complete. (Do not attempt to prove that it is $\mathcal{NP}$-complete. You may refer to the language by its standard name, such as, "The boolean satisfiability problem.")

6. [20 points] Give an unambiguous context-free grammar for the language of all regular expressions over the alphabet $\{a, b\}$. Your grammar should be consistent with the semantics of regular expressions.

7. [20 points] What does it mean to say that a language L is $\mathcal{NP}$-*complete*? In your explanation, you may use the following terms without definition or explanation: *machine*, *accept*, *compute*, *deterministic*, *non-deterministic*, *polynomial time*, *reduction*.

8. [20 points] Let L be any regular language, and let $\text{SUFF}(L)$ be the set of all suffixes of L. Prove that $\text{SUFF}(L)$ is regular.

   (Hint: There is a DFA that accepts L. Construct an NFA that accepts $\text{SUFF}(L)$.)

9. [20 points] Let L be the set of all strings over $\{a, b\}$ which have equal numbers of a's and b's. Prove, by contradiction, that L is not regular, as follows. First, you know that $\{a^n b^n\}$ is not regular (don't prove it; you can just take it as given). Second, you know that $\{a^i b^j\}$ is regular. (How do you know this?) Third, there is a theorem about intersections of regular language. Finally, obtain a contradiction.