University of Nevada, Las Vegas
Computer Science 456/656 Fall 2003
Review for Final Exam

This version `Fri Dec 5 09:38:56 PST 2003` **I may add just a few more questions if I realize that I omitted a topic.**

Disclaimer: This "practice final" is much longer than the actual exam will be, since it is mostly a concatenation of problems from several past exams. To save paper, I did not leave enough room to work the problems, as I will on the actual examination.

1. True or False.

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

   (b) _____ The intersection of any context-free language with any context-free language is context-free.

   (c) _____ The complement of every recursive language is recursive.

   (d) _____ The complement of every recursively enumerable language is recursively enumerable.

   (e) _____ Every language which is generated by an unrestricted grammar is recursively enumerable.

   (f) _____ The question of whether two context-free grammars generate the same language is undecidable.

   (g) _____ There exists some proposition which is true but which has no proof.

   (h) _____ The set of all binary numerals for prime numbers is in the class $\mathcal{P}$.

   (i) _____ If L_1 reduces to L_2 in polynomial time, and if L_2 is $\mathcal{NP}$, and if L_1 is $\mathcal{NP}$-complete, then L_2 must be $\mathcal{NP}$-complete.

   (j) _____ Given any context-free grammar G and any string $w \in L(G)$, there is always a unique leftmost derivation of w using G.

   (k) _____ For any finite automaton, there is always a unique minimal non-deterministic finite automaton equivalent to it.

   (l) _____ Using multi-processors and other advanced technology, it is possible to design a machine which decides the halting problem.

   (m) _____ The question of whether two context-free grammars are equivalent is decidable.

   (n) _____ The question of whether two regular expressions are equivalent is decidable.

2. Every context-free language is accepted by some _____.

3. If there is an easy reduction from L_1 to L_2, then _____ is at least as hard as _____.

4. For each question, give one example of a language, either by using the standard name of the language, or describing it in very few words. No proofs are required.

   (a) Give an example of an infinite language that is regular.

(b) Give an example of a context-free language that is not regular.

(c) Give an example of a language in the class $\mathcal{P}$ that is not context-free.

(d) Give an example of a recursive language that is $\mathcal{NP}$-complete.

(e) Give an example of a recursive language that is not in the class $\mathcal{NP}$.

(f) Give an example of a recursively enumerable language that is not recursive.

(g) Give an example of a language that is not recursively enumerable.

5. Let L be the language of all binary numerals for positive integers equivalent to 2 modulo 3. Thus, for example, the binary numerals for 2, 5, 8, 11, 13, ... are in L. We allow a binary numeral to have leading zeros; thus (for example) $001110 \in L$, since it is a binary numeral for 14. Draw a minimal DFA which accepts L. ERROR IN THIS PROBLEM WAS CORRECTED.

6. Define what it means to say that a certain language L is *accepted in polynomial time* by a certain machine M.

7. Consider the context-free grammar G_1 with start symbol S and productions as follows:

   $S \to SS$

   $S \to aSb$

   $S \to \epsilon$

   (a) Show that G_1 is ambiguous.

   (b) Find an unambiguous context-free grammar G_2 that is equivalent to G_1.

8. Consider the context-free grammar with start symbol S and productions as follows:

   $S \to s$

   $S \to bLn$

   $S \to wS$

   $L \to \epsilon$

   $L \to SL$

   Write a leftmost derivation of the string *bawabwann*

9. (a) What class of machines accepts the class of context free languages?

   (b) What class of machines accepts the class of regular languages?

(c) What class of machines accepts the class of recursively enumerable languages?

(d) What is the Church-Turing Thesis, and why is it important?

(e) Define what it means to say that a language L is accepted in *polynomial time* by a certain machine M.

(f) Give a definition of the language class $\mathcal{NP}$-TIME.

(g) Give a definition of *$\mathcal{NP}$-complete language.*

(h) What does it mean to say that a language L is *decidable*?

(i) What is an unrestricted (same as general) grammar?

(j) What does it mean to say that a language can be generated in *canonical order*? What is the class of languages that can be so generated?

(k) What does it mean to say that machines M_1 and M_2 are *equivalent*?

(l) What does it mean to say that a context free grammar is *ambiguous*?

(m) Anita has developed a new programming language with lots of useful features. But her professor asks her to write a program (in this new language) that emulates a universal Turing machine. Why is that important?

10. Let $\Sigma = \{0, 1\}$, the binary alphabet. We say a string w over Σ is *mostly positive* if w has more 1's than 0's. Let L be the set of mostly positive strings over Σ.

    Give a context-free grammar for L.

11. Explain how, if a language is accepted by some non-deterministic Turing Machine, it is also accepted by some deterministic Machine.

12. Let $\Sigma = \{0, 1\}$, the binary alphabet. Let L be the set of all strings w over Σ of the form $1^n 0^n 1^n$, where $n \geq 0$. Use the pumping lemma to prove that L is not a context-free language.

13. True or False:

    (a) _____ The intersection of any context-free language with any regular language is context-free.

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

    (c) _____ The complement of every recursively enumerable language is recursively enumerable.

    (d) _____ Every context-free language over an alphabet of size 1 is regular.

    (e) _____ Let $L = \{\langle M \rangle \mid M \text{ halts with no input}\}$. Then L is recursively enumerable.

    (f) _____ The complement of every context-free language is context-free.

    (g) _____ No language which has an ambiguous context-free grammar can be accepted by a DPDA.

    (h) _____ The question of whether two context-free grammars generate the same language is decidable.

    (i) _____ There exists some proposition which is true but which has no proof.

(j) _____ The union of any two context-free languages is context-free.

(k) _____ The question of whether a given Turing Machine halts with empty input is decidable.

(l) _____ The class of languages accepted by non-deterministic finite automata is the same as the class of languages accepted by deterministic finite automata.

(m) _____ The intersection of any two regular languages is regular.

(n) _____ The intersection of any two context-free languages is context-free.

(o) _____ If L_1 reduces to L_2 in polynomial time, and if L_2 is $\mathcal{NP}$, then L_1 must be $\mathcal{NP}$.

(p) _____ For any finite automaton, there is a unique minimal deterministic finite automaton equivalent to it.

(q) _____ The complement of every recursive language is recursive.

14. For each language given, write "R" if the language is recursive, write "RE not R" if the language is recursively enumerable but not recursive, and write "not RE" if the language is not recursively enumerable.

    (a) _____ The language consisting of all Pascal programs p such that p halts if given p as its input file.

    (b) _____ The language of all encodings of Turing Machines which fail to halt for at least one possible input string.

    (c) _____ The 0-1 Traveling Salesman Problem.

    (d) _____ The diagonal language.

    (e) _____ The universal language.

    (f) _____ L_{sat}, the set of satisfiable boolean expressions.

15. Draw a minimal DFA which accepts the language L over the binary alphabet $\Sigma = \{0, 1\}$ consisting of all strings in which every '001' is followed by '1'.

16. Consider the context-free grammar G, with start symbol S and productions as follows:

    $S \to s$

    $S \to bLn$

    $S \to iS$

    $S \to iSeS$

    $L \to \epsilon$

    $L \to LS$

    Prove that G is ambiguous by giving two different leftmost derivations for some string.

17. (a) Give a definition of what it means for a language to be *accepted in polynomial time* by a certain machine M.

    (b) Give a definition of the language class $\mathcal{P}$-TIME.

    (c) Give a definition of the language class $\mathcal{NP}$-TIME.

    (d) Give one example of an $\mathcal{NP}$-complete language.

    (e) What does it mean to say that a language L_1 reduces to a language L_2 in polynomial time?

    (f) What does it mean to say that a language L is decidable?

18. Prove that, if a language is accepted by some non-deterministic Turing Machine, it is also accepted by some deterministic Machine.

19. Let $\Sigma = \{0, 1\}$, the binary alphabet. Let $L \subseteq \Sigma^\star$ be recursively enumerable, but not recursive, and let M be a Turing machine that accepts L. If $w \in L$, let $T_M(w)$ be the number of steps of M in the valid computation with input w. For any string $w \in \Sigma^\star$, define

$$f(w) \quad = \quad \begin{cases} T_M(w) \text{ if } w \in L \\ 0 \text{ if } w \notin L \end{cases}$$

    Prove that f is not recursive.

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

    (b) _____ The complement of every context-free language is context-free.

    (c) _____ The question of whether two context-free grammars generate the same language is decidable.

    (d) _____ There exists some proposition which is true but which has no proof.

    (e) _____ The union of any two context-free languages is context-free.

    (f) _____ The intersection of any two context-free languages is context-free.

    (g) _____ If L_1 reduces to L_2 in polynomial time, and if L_2 is $\mathcal{NP}$, then L_1 must be $\mathcal{NP}$.

    (h) _____ The complement of every recursive language is recursive.

21. For each language given, write "REG" if the language is regular, write "CF not REG" if the language is context-free but not regular, write "R not CF" if the language is recursive but not context-free, write "RE not R" if the language is recursively enumerable but not recursive, and write "not RE" if the language is not recursively enumerable.

    (a) _____ The set of all strings over the alphabet $\{a, b, c\}$ where are **not** of the form $a^n b^n c^n$.

    (b) _____ The language of all encodings of Turing Machines which halt for at least one possible input string.

    (c) _____ The 0-1 Traveling Salesman Problem.

    (d) _____ The diagonal language.

(e) _____ The set of all binary numerals for integers of the form $3i + 5j$, where i, j could be any natural numbers.

22. Draw a minimal DFA which accepts the language L over the binary alphabet $\Sigma = \{a, b, c\}$ consisting of all strings which contain either *aba* or *caa* as a substring.

23. Consider the context-free grammar with start symbol S and productions as follows:

    $S \to s$

    $S \to bLn$

    $S \to iSeS$

    $L \to \epsilon$

    $L \to SL$

    Write a leftmost derivation of the string *ibssneises*

24. What does it mean to say that a language L is decidable?

25. What does it mean to say that machines M_1 and M_2 are *equivalent*?

26. Give a definition of the language class $\mathcal{P}$-TIME.

27. Give a definition of the language class $\mathcal{NP}$-TIME.

28. Consider the language L of all strings which would be acceptable as algebraic expressions involving variables and constants, where:

    - Every variable name is either x, y, or z.
    - Every constant is a natural number between 0 and 9
    - The only operators are addition, subtraction, and multiplication.
    - The symbol '$-$' is used only for subtraction. There is no negation.
    - There is no multiplication symbol. Multiplication is indicated by concatenating strings.
    - In multiplication of a constant by anything else, the constant must come first, and there can be at most one constant factor in any term.
    - Parentheses can be used.

    Here are some strings in the language $x(y + 2z)$, $x - 1 - z$, $4(zx - 2y)(x + z(x - 1))$.

    Give a grammar for L which is consistent with the usual semantics (as you learned in school) of such expressions.

29. Let $\Sigma = \{1\}$, the unary alphabet. Let $L = \{1^{2^n}\}$, the powers of 2 in unary. Use the pumping lemma to prove that L is not a context-free language.

30. True or False:

    (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) _____ The question of whether a the language generated by a given context-free grammar is decidable is decidable. (Warning: this is a trick question.)

(g) _____ Let L be the language consisting of just one string, and that string is "1" if $\mathcal{P} = \mathcal{NP}$ and "0" otherwise. Is L decidable? (For this problem, you may answer either T or F or "unknown." But this is a trick question.)

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

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

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

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

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

(m) _____ 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.)

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

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

31. Draw a minimal DFA which accepts the language of all strings over $\{a, b, c\}$ which do not contain the substring *abaa*.

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

33. 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.")

34. 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.")

35. 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.

36. 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*.

37. 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 $\text{SUFF}(L)$.)

38. 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.

39. Indicate, using words and diagrams, how to prove that any 2-tape Turing machine can be emulated by a 1-tape Turing machine with a multiple track tape.

40. Draw a transition-diagram (called "state diagram" in our textbook) for a Turing Machine that accepts $\{a^n b^n | n \geq 0\}$. (Do **not** draw a transition diagram for a PDA.)