University of Nevada, Las Vegas
Computer Science 456/656 Spring 2018

Practice for the Final on May 9, 2018

The entire examination is 775 points. The real final will be much shorter.

Name:_____

No books, notes, scratch paper, or calculators. 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.

1. [5 points each] True or False. If the question is currently open, write "O" or "Open."

   (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 general 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 deterministic 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 regular expressions are equivalent is decidable.

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

(o) _____ The halting problem is recursively enumerable.

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

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

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

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

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

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

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

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

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

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

2. [5 points each] True or False. If the question is currently open, write "O" or "Open."

(a) _____ The language of all regular expressions over the binary alphabet is a regular language.

(b) _____ Let π be the ratio of the circumference of a circle to its diameter. (That's the usual meaning of π you learned in second grade.) The problem of whether the n^{th} digit of π, for a given n, is equal to a given digit is decidable.

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

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

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

(f) _____ Every function that can be mathematically defined is recursive.

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

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

(i) _____ 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.)

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

(k) _____ Every context-free grammar can be parsed by some deterministic top-down parser.

(l) _____ Every context-free grammar can be parsed by some non-deterministic top-down parser.

(m) _____ Commercially available parsers cannot use the LALR technique, since most modern programming languages are not context-free.

(n) _____ The boolean satisfiability problem is undecidable.

(o) _____ There is a parallel processor machine which can solve the boolean circuit problem in polylogarithmic time.

(p) _____ We can prove that $\mathcal{P} = \mathcal{NP}$ if RSA coding is insecure.

3. [10 points] If there is an easy reduction from L_1 to L_2, then _____ is at least as hard as _____.

4. [5 points each] 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) _____ L_{sat}, the set of satisfiable boolean expressions.

5. [5 points each] 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 not $\mathcal{NP}$.

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

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

6. [15 points] Draw the state diagram for a minimal DFA that accepts the language described by the regular expression `(a+b)*ab`

7. [15 points] Write a regular expression for the language accepted by the NFA shown in Figure 1.

8. [20 points] 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, 14, 17 ... 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.

9. [10 points] 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 *bswsbwsnn*

10. [5 points] What class of machines accepts the class of context free languages?

11. [5 points] What class of machines accepts the class of regular languages?

12. [5 points] What class of machines accepts the class of recursively enumerable languages?

13. [10 points] What is the Church-Turing Thesis, and why is it important?

14. [10 points] 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?

15. [5 points] What does it mean to say that machines M_1 and M_2 are *equivalent*?

16. [10 points each] Give definitions for each of the following. Assume that you are writing the definitions for a top mathematics graduate student who has never taken a course in automata theory, but who did sit in a lecture once where "alphabet," "string," and "language" were defined. The student also once read an article in which Turing machines were defined.

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

    (b) Give the definition of a *polynomial time reduction* of a language L_1 to another language L_2.

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

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

17. [15 points] 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. **Very hard.**

18. [10 points] Give a Chomsky Normal Form grammar for the Dyck language.

19. [10 points] 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'.

20. [20 points] Construct a minimal DFA equivalent equivalent to the NFA shown in Figure 1.

Figure 1: The NFA for Problems 7 and 20.

21. [10 points] 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.

22. [10 points] What does it mean to say that a language L_1 reduces to a language L_2 in polynomial time?

23. [10 points] What does it mean to say that a language L is decidable?

24. [5 points each] Classify each of the following problems or languages as just one of the following three:

    Known to be polynomial, known to be NP but not known to be NP-complete, or known to be NP-complete.

    (a) _____ Boolean satisfiability.

    (b) _____ The 0-1 traveling salesman problem.

    (c) _____ The restricted knapsack problem where, for each instance, the size of each item is a positive integer that does not exceed the square of the number of items, and all the numbers are written in binary notation.

    (d) _____ The clique problem.

    (e) _____ The language generated by a given context-free grammar.

    (f) _____ The set of all pairs of integers (n, m) such that n has a proper divisor which is at least m. (The input for an instance of this problem is the string consisting of the binary numeral for n, followed by a comma, followed by the binary numeral for m.)

25. [30 points] Consider the grammar given below. Design an LALR parser for that grammar.

    For your convenience, stack states are given on the right hand side of the production.

    (a) $S \to \epsilon$

    (b) $S \to a_2 S_3 b_4 S_5$

26. [5 points each] 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 0-1 Traveling Salesman Problem.

(c) _____ The diagonal language.

27. [15 points] 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.

28. [30 points] 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 an unambiguous context-free grammar for L which is consistent with the usual semantics (as you learned in school) of such expressions.

29. [5 points each] Which of the problems below are known to be $\mathcal{NP}$-complete?

(a) _____ The traveling salesman problem.

(b) _____ Boolean satisfiability.

(c) _____ The halting problem.

(d) _____ Primality.

(e) _____ The context-free grammar equivalence problem.

(f) _____ The independent set problem.

30. [10 points] State the pumping lemma for regular languages accurately. If you have all the right words but in the wrong order, that means you truly do not understand the lemma, and you might get no partial credit at all.

31. [20 points] In class, we demonstrated that a language is in the class $\mathcal{NP}$ if and only if it has a polynomial time *verifier*.

What is a polynomial time verifier of a language? Your explanation should include the word "certificate," or as it is sometimes known, "witness."

32. [30 points] **Warning: this problem requires you to go beyond the contents of the lectures.** Prove that the class of regular languages is a subclass of $\mathcal{NC}$ (Nick's class).

33. [30 points] **Warning: this problem is very hard.** Every sliding block puzzle

    `http://www.maa.org/editorial/mathgames/mathgames_12_13_04.html`

    can be encoded as a string of bits. All sliding block puzzles that you read about are solvable, or else they wouldn't be proposed, but you could obviously make up a sliding block puzzle that has no solution. Is the set of encodings of solvable sliding block puzzles $\mathcal{NP}$?

34. These are reduction problems. I could give one of them on the test. The proof should be very informal.

    (a) Find a $\mathcal{P}$-time reduction of the knapsack problem to the partition problem.

    (b) Find a $\mathcal{P}$-time reduction of 3-CNF-SAT to the independent set problem.

35. [20 points] Prove that every recursively enumerable language is accepted by some Turing machine.

36. [20 points] Prove that every language accepted by a Turing machine is recursively enumerable.