University of Nevada, Las Vegas
Computer Science 456/656 Fall 2020

Practice for the Final on December 9, 2020

The entire examination is 885 points. The actual final examination will be shorter.

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

   (i) ________ Every subset of a regular language is regular.

   (ii) ________ The intersection of any context-free language with any regular language is context-free.

   (iii) ________ The complement of every recursive language is recursive.

   (iv) ________ The complement of every recursively enumerable language is recursively enumerable.

   (v) ________ Every language which is generated by a general grammar is recursively enumerable.

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

   (vii) ________ There exists some proposition which is true but which has no proof.

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

   (ix) ________ 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.

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

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

   (xii) ________ The question of whether two regular expressions are equivalent is $\mathcal{NP}$-complete.

   (xiii) ________ The halting problem is recursively enumerable.

   (xiv) ________ The complement of every context-free language is context-free.

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

   (xvi) ________ The union of any two context-free languages is context-free.

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

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

(xix) ________ The class of languages accepted by non-deterministic push-down automata is the same as the class of languages accepted by deterministic push-down automata.

(xx) ________ The intersection of any two regular languages is regular.

(xxi) ________ The intersection of any two context-free languages is context-free.

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

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

(xxiv) ________ Every language which is accepted by some non-deterministic machine is accepted by some deterministic machine.

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

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

(xxvii) ________ There cannot exist any computer program that decides whether any two given C++ programs are equivalent.

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

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

(xxx) ________ Every regular language is in the class $\mathcal{NC}$

(xxxi) ________ Every function that can be mathematically defined is recursive.

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

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

(xxxiv) ________ 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.)

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

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

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

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

(xxxix) ________ The boolean satisfiability problem is undecidable.

(xl) ________ If anyone ever proves that $\mathcal{P} = \mathcal{NP}$, then all public key/private key encryption systems will be known to be insecure.

(xli) ________ If a string w is generated by a context-free grammer G, then w has a unique leftmost derivation if and only if it has a unique rightmost derivation.

These problem are only a sample of the many true/false questions I have given during the years I have taught this course. I will post a more complete list later.

2. [10 points] If there is an easy reduction from L_1 to L_2, then ________ is at least as hard as ________.

3. [15 points] Draw the state diagram for a minimal DFA that accepts the language described by the regular expression `a*a(b+ab)*`

4. [15 points] Write a regular expression for the language accepted by the NFA shown in Figure 1.

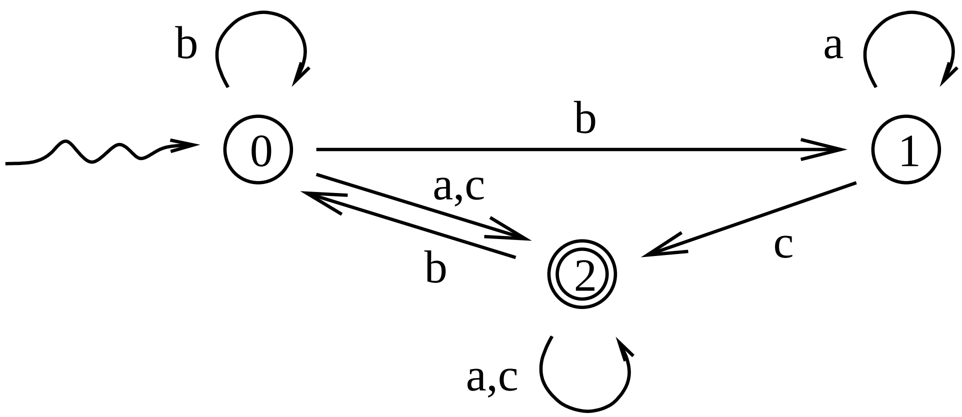

Figure 1: The NFA for Problems 4 and 17.

5. [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.

6. [20 points] Design a PDA that accepts the language of all palindromes over the alphabet $\{a, b\}$.

7. [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 $bswsbwsnn$

8. [5 points] What class of machines accepts the class of context free languages?

9. [5 points] What class of machines accepts the class of regular languages?

10. [5 points] What class of machines accepts the class of recursively enumerable languages?

11. [10 points] What is the Church-Turing Thesis, and why is it important?

12. [10 points] What does it mean to say that a language can be recursively enumerated in *canonical order*? What is the class of languages that can be so enumerated?

13. [5 points] What does it mean to say that machines M_1 and M_2 are *equivalent*?

14. Give definitions: [10 points each]

    (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) Give a definition of a *decidable* language.

15. [15 points] We say a binary string w over is *balanced* if w has the same number of 1's as 0's. Let L be the set of balanced binary strings. Give a context-free grammar for L.

16. [10 points] Give a Chomsky normal form (CNF) grammar for the language of all palindromes of odd length over the alphabet $\{a, b\}$.

17. [20 points] Construct a minimal DFA equivalent equivalent to the NFA shown in Figure 1.

18. [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.

19. [10 points] What does it mean to say that a language L_1 reduces to a language L_2 in polynomial time?

20. [10 points] What does it mean to say that a language L is decidable?

21. Every language we have discussed this semester falls into one of these categories.

    a. $\mathcal{NC}$.

    b. $\mathcal{P}$ but not known to be $\mathcal{NC}$.

    c. $\mathcal{NP}$ but not known to be $\mathcal{P}$ and not known to be $\mathcal{NP}$-complete.

    d. Co-$\mathcal{NP}$ but not known to be $\mathcal{P}$.

    e. Known to be $\mathcal{NP}$-complete.

    f. Known to be $\mathcal{P}$-space, but not known to be $\mathcal{NP}$.

    g. Known to be EXP-time, but not known to be $\mathcal{P}$-space.

    h. Known to be EXP-space, but not known to be EXP-time.

i. Recursive, but not known to be EXP-space.

j. RE (Recursively enumerable), but not recursive.

k. Co-RE, but not recursive.

l. Neither RE nor co-RE.

Identify which of the above categories each language or problem listed below belongs to. 5 points each.

(i) ________ The language consisting of all Pascal programs P such that P halts if given P as its input file.

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

(iii) ________ The 0-1 Traveling Salesman Problem.

(iv) ________ The diagonal language.

(v) ________ L_{sat}, the set of satisfiable boolean expressions.

(vi) ________ The language described by the regular expression a^*b*.

(vii) ________ The Boolean circuit problem.

(viii) ________ Dynamic programming, where the number of subproblems is polynomial.

(ix) ________ The language $\{a^n b^n \,:\, n \geq 0\}$.

(x) ________ Factorization of a binary numeral.

(xi) ________ Dynamic programming, where the number of subproblems is polynomial, and each subproblem can be computed using only the previous subproblem.

(xii) ________ Multiplication of two binary numerals.

(xiii) ________ The 0-1 traveling salesman problem.

(xiv) ________ The restricted subset sum problem where, for each instance, each number is a positive integer that does not exceed the square of the number of items, and all the numbers are written in binary notation.

(xv) ________ The set of position of HEX from which the next player to move can force a win.

(xvi) ________ The set of configurations of RUSH HOUR which are solvable.

(xvii) ________ Factorization of a unary numeral.

(xviii) ________ The halting problem.

(xix) ________ The diagonal language.

(xx) ________ The clique problem.

(xxi) ________ Primality, where the input is written in binary.

(xxii) ________ The language generated by a given context-free grammar.

(xxiii) ________ The language of all monotone increasing sequences of arabic numerals for positive integers. (For example, "1,5,23,41,200,201" is a member of that language.)

(xxiv) ________ The language accepted by a given DFA.

(xxv) ________ The 0/1 factoring problem, *i.e.* 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.)

(xxvi) ________ The unary version of the 0/1 factoring problem, *i.e.* 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 unary numeral for n, followed by a comma, followed by the unary numeral for m.)

(xxvii) ________ The set of all positions from which black can force a win in a game of generalized checkers.

(xxviii) ________ The set of all configurations of the children's game "Boxes" from which the first player can force a win. (I used to play that game as a child, and I never did figure out an optimal strategy. I don't feel bad about that anymore, now that I know the complexity class of that problem.)

(xxix) ________ The set of all configurations of the game "Nim" from which the first player can force a win.

(xxx) ________ The set of all ordered pairs of positive numerals $(\langle n\rangle, \langle m\rangle)$ $m = \Sigma(n)$, where Σ is the busy beaver function.

(xxxi) ________ The traveling salesman problem.

(xxxii) ________ Boolean satisfiability.

(xxxiii) ________ The halting problem.

(xxxiv) ________ Primality of a binary numeral.

(xxxv) ________ Primality of a unary numeral.

(xxxvi) ________ The context-free grammar equivalence problem.

(xxxvii) ________ The independent set problem.

22. [30 points] The grammar below is an alternative unambiguous CF grammar for the Dyck language, and is parsed by the given LALR parser. Write a computation of the parser for the input string *aabb*.

23. [30 points] The grammar below is an alternative unambiguous CF grammar for the Dyck language.

1 $S \to a_2 S_3 b_4 S_5$

2 $S \to \lambda$

	a	b	$	S
0	$s2$		$r2$	1
1			halt	
2	$s2$	$r2$		3
3		$s4$		
4	$s2$	$r2$	$r2$	5
5	$r1$	$r1$	$r1$	

The following ambiguous grammar, with start symbol E, generates certain expressions. The parser illustrated below is intended to parse a string in accordance with C++ rules. However, the action table has one error. Identify that error, and fix it.

1 $E \to x_2$

2 $E \to E -_3 E_4$

3 $E \to -_5 E_6$

4 $E \to (_7 E_8)_9$

	x	$-$	(	)	\$	S
0	$s2$	$s5$	$s7$			1
1		$s3$			halt	
2		$r1$		$r1$	$r1$	
3		$s4$				4
4		$r2$		$r2$	$r2$	
5	$s2$	$s5$	$s7$			6
6		$s3$		$r3$	$r3$	
7	$s2$	$s5$	$s7$			8
8		$s3$		$s9$		
9		$r4$		$r4$	$r4$	

24. For each of the following languages, state whether the language is regular, context-free but not regular, context-sensitive but not context-free, or not context-sensitive.

    [5 points each]

    (a) ________________ The set of all strings over the alphabet $\{a, b\}$ of the form $a^n b^m$.

    (b) ________________ The set of all strings over the alphabet $\{a, b\}$ of the form $a^n b^n$.

    (c) ________________ The set of all strings over the alphabet $\{a, b, c\}$ of the form $a^n b^n c^n$.

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

    (e) ________________ The set of all strings over the alphabet $\{a\}$ of the form $a^{n^2 : n \geq 1}$.

25. [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.

26. [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.

27. [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."

28. 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 3-CNF-SAT to the independent set problem.

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

29. [20 points] Prove that every recursively enumerable language is accepted by some Turing machine.

30. [20 points] Prove that every language accepted by a Turing machine is recursively enumerable.

31. [20 points] Give a context-sensitive grammar for the language $\{a^n b^n c^n \,:\, n \geq 1\}$

32. [20 points] Give a context-sensitive grammar for the language $\{a^n b^n c^n d^n : n \geq 1\}$

33. [20 points] Give a general grammar for the language $\left\{a^{2^n}\right\}$

34. [20 points] Prove that the halting problem is undecidable.