University of Nevada, Las Vegas Computer Science 456/656 Fall 2019

Assignments 6 and 7: Due November 13, 2019

Name:____

You are permitted to work in groups, get help from others, read books, and use the internet. But the handwriting on this document must be your own. Print out the document, staple, and fill in the answers. You may attach extra sheets. Turn in the pages to the graduate assistant at the beginning of class, November 6.

1. True or False. T = true, F = false, and O = open, meaning that the answer is not known science at this time. In the questions below, $\mathcal{P}$ and $\mathcal{NP}$ denote $\mathcal{P}$-TIME and $\mathcal{NP}$-TIME, respectively.

   (i) _____ Every language generated by an unambiguous context-free grammar is accepted by some DPDA.

   (ii) _____ The language $\{a^n b^n c^n d^n \mid n \geq 0\}$ is recursive.

   (iii) _____ Let L be the language over $\{a, b, c\}$ consisting of all strings which have more a's than b's and more b's than c's. There is some PDA that accepts L.

   (iv) _____ The language $\{a^n b^n c^n \mid n \geq 0\}$ is in the class $\mathcal{P}$-TIME.

   (v) _____ Every undecidable problem is $\mathcal{NP}$-complete.

   (vi) _____ The language $\{a^n b^n \mid n \geq 0\}$ is context-free.

   (vii) _____ The language $\{a^n b^n c^n \mid n \geq 0\}$ is context-free.

   (viii) _____ The language $\left\{a^i b^j c^k \mid j = i + k\right\}$ is context-free.

   (ix) _____ Every problem that can be mathematically defined has an algorithmic solution.

   (x) _____ The intersection of two undecidable languages is always undecidable.

   (xi) _____ Every $\mathcal{NP}$ language is decidable.

   (xii) _____ The clique problem is $\mathcal{NP}$-complete.

   (xiii) _____ The traveling salesman problem is $\mathcal{NP}$-hard.

   (xiv) _____ The union of two $\mathcal{NP}$ languages must be $\mathcal{NP}$.

   (xv) _____ The intersection of two $\mathcal{NP}$-complete languages must be $\mathcal{NP}$-complete.

   (xvi) _____ $\mathcal{NC} = \mathcal{P}$.

   (xvii) _____ $\mathcal{P} = \mathcal{NP}$.

   (xviii) _____ $\mathcal{NP} = \mathcal{P}$-SPACE

   (xix) _____ $\mathcal{P}$-SPACE = EXP-TIME

(xx) ______ EXP-TIME = EXP-SPACE

(xxi) ______ There is a deterministic parser for any context-free grammar.

(xxii) ______ The traveling salesman problem (TSP) is $\mathcal{NP}$-complete.

(xxiii) ______ The knapsack problem is $\mathcal{NP}$-complete.

(xxiv) ______ The language consisting of all satisfiable Boolean expressions is $\mathcal{NP}$-complete.

(xxv) ______ The Boolean Circuit Problem is in $\mathcal{P}$.

(xxvi) ______ The Boolean Circuit Problem is in $\mathcal{NC}$.

(xxvii) ______ The set of strings that your high school algebra teacher would accept as legitimate expressions is a context-free language.

(xxviii) ______ The language consisting of all strings over $\{a, b\}$ which have more a's than b's is context-free.

(xxix) ______ 2-SAT is $\mathcal{P}$–TIME.

(xxx) ______ 3-SAT is $\mathcal{P}$–TIME.

(xxxi) ______ Primality, where the input is written in binary, is $\mathcal{P}$-TIME.

(xxxii) ______ There is a $\mathcal{P}$–TIME reduction of the halting problem to 3-SAT.

(xxxiii) ______ Every context-free language is in $\mathcal{P}$.

(xxxiv) ______ Every context-free language is in $\mathcal{NC}$.

(xxxv) ______ Addition of binary numerals is in $\mathcal{NC}$.

(xxxvi) ______ Every context-sensitive language is in $\mathcal{P}$.

(xxxvii) ______ Every language generated by a general grammar is recursive.

(xxxviii) ______ Every language generated by a general grammar is recursively enumerable.

(xxxix) ______ Every language accepted by a non-deterministic machine is accepted by some deterministic machine.

(xl) ______ The problem of whether two given context-free grammars generate the same language is co–$\mathcal{RE}$.

(xli) ______ The problem of whether a given string is generated by a given context-free grammar is decidable.

(xlii) ______ The language of all fractions (using base 10 numeration) whose values are less than π is decidable. (A *fraction* is a string. "314/100" is in the language, but "22/7" is not.)

(xliii) ______ There exists a polynomial time algorithm which finds the prime factors of any positive integer, where the input is given as a unary ("caveman") numeral.

(xliv) ______ For any two languages L_1 and L_2, if L_1 is undecidable and there is a recursive reduction of L_1 to L_2, then L_2 must be undecidable.

(xlv) ______ For any two languages L_1 and L_2, if L_2 is undecidable and there is a recursive reduction of L_1 to L_2, then L_1 must be undecidable.

(xlvi) ______ If L is any $\mathcal{NP}$ language, there must be a $\mathcal{P}$–TIME reduction of the partition problem to L.

(xlvii) ______ If L is $\mathcal{NP}$ and also co–$\mathcal{NP}$, then L must be $\mathcal{P}$.

(xlviii) ______ Recall that if $\mathcal{L}$ is a class of languages, co–$\mathcal{L}$ is defined to be the class of all languages that are not in $\mathcal{L}$. Let $\mathcal{RE}$ be the class of all recursively enumerable languages. If L is in $\mathcal{RE}$ and also L is in co–$\mathcal{RE}$, then L must be decidable.

(xlix) ______ Every language is enumerable.

(l) ______ If a language L is undecidable, then there can be no machine that enumerates L.

(li) ______ There exists a mathematical proposition that can be neither proved nor disproved.

(lii) ______ There is a non-recursive function which grows faster than any recursive function.

(liii) ______ For every real number x, there exists a machine that runs forever and outputs the string of decimal digits of x.

(liv) ______ **Rush Hour**, the puzzle sold in game stores everywhere, generalized to a board of arbitrary size, is $\mathcal{P}$–SPACE–complete.

(lv) ______ If two regular expressions are equivalent, there is a polynomial time proof that they are equivalent.

(lvi) ______ There is a well-defined function f on positive integers, where:

$f(n) = 0$ if $n = 1$

$f(n) = 1 + f(n/2)$ if n is even

$f(n) = 1 + f(3n + 1)$ if n is odd and greater than 1.

For example, $f(1) = 0$, $f(2) = 1$, $f(3) = 7$, $f(4) = 2$, $f(5) = 5$, $f(6) = 8$, $f(7) = 16$, ...

Hint: look on the internet for "Collatz."

(lvii) ______ The *busy beaver* function is recursive.

(lviii) ______ The Post correspondence problem is $\mathcal{NP}$–COMPLETE.

2. Suppose x and y are positive integers, and their binary numerals $\langle x \rangle$ and $\langle y \rangle$ each have length n. Then $\langle xy \rangle$, the binary numeral of their product, has length at most $2n$. Explain how the problem of computing $\langle xy \rangle$ from $\langle x \rangle$ and $\langle y \rangle$ is in the class $\mathcal{NC}$.

3. Let L be the language generated by the context-free grammar below. What is the minimum pumping length of L? (Note that this grammar does not contain the production $S \to iS$.) Hint: read `http://web.cs.unlv.edu/larmore/Courses/CSC456/pumping.pdf`

   $S \to wS$
   $S \to iSeS$
   $S \to a$

4. Explain to me why $\mathcal{NP}$–TIME $\subseteq$ $\mathcal{P}$–SPACE.

5. Recall that a fraction is a string. If x is any real number, let LESS_x be the set of fractions whose values are less than x, and let MORE_x be the set of fractions whose values are more than x.

   (a) Is it true that, for every real number x, LESS_x is decidable?

   (b) Is it true that, for every real number x, MORE_x is decidable?

   (c) Is there a real number x such that LESS_x is decidable but MORE_x is not decidable?

   (d) Is there a real number x such that LESS_x is recursively enumerable but MORE_x is not recursively enumerable?

   Hint: If L is a language over the unary alphabet $\{1\}$, let $x_L = \sum_{i=0}^{\infty} 2^{-a_i}$, where $a_i = 1$ if if $1^i \in L$, and $a_i = 0$ if $1^i \notin L$. Depending on whether L is decidable, or whether L is recursivly enumerable, is LESS_{x_L} decidable? Recursively enumerable?

6. Let L be the following language.

   (a) If $\mathcal{P} = \mathcal{NP}$, then $L = \{1\}$.

   (b) If $\mathcal{P} \neq \mathcal{NP}$, then $L = \{0\}$.

   Is L decidable? Explain your answer.

7. Find a general grammar which generates $\left\{a^{2^n}\right\}$.