

CS 456/656 Study Guide for Examination October 23, 2024

A *binary function* is defined to be a function F on binary strings such that, for each binary string w , $F(w)$ is a binary string. (Of course, the strings could be numerals.)

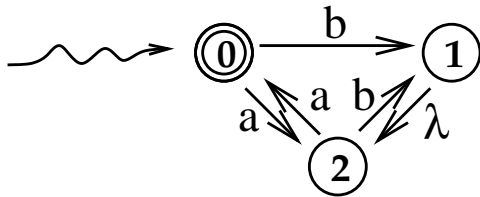
1. True or False. T = true, F = false, and O = open, meaning that the answer is not known science at this time.
 - (i) ----- Every subset of a regular language is regular.
 - (ii) ----- The complement of a CFL is always a CFL.
 - (iii) ----- The class of context-free languages is closed under union.
 - (iv) ----- The class of context-free languages is closed under intersection.
 - (v) ----- The language of binary numerals for multiples of 23 is regular.
 - (vi) ----- The set of binary numerals for prime numbers is in \mathcal{P} -TIME.
 - (vii) ----- Every language is countable.
 - (viii) ----- The set of languages over the binary alphabet is uncountable.
 - (ix) ----- The complement of any \mathcal{NP} language is \mathcal{NP} .
 - (x) ----- The complement of any decidable language is decidable.
 - (xi) ----- The complement of any undecidable language is undecidable.
 - (xii) ----- $\mathcal{P} = \mathcal{NP}$.
 - (xiii) ----- The complement of any \mathcal{P} -TIME language is \mathcal{P} -TIME.
 - (xiv) ----- The complement of any \mathcal{P} -SPACE language is \mathcal{P} -SPACE.
 - (xv) ----- The complement of every recursive language is recursive.
 - (xvi) ----- The complement of every recursively enumerable language is recursively enumerable.
 - (xvii) ----- If a language L is accepted by an NFA M with p states, then L has pumping length p .
 - (xviii) ----- Given any unambiguous context-free grammar G and any string $w \in L(G)$, there is always a unique leftmost derivation of w using G .
 - (xix) ----- For any deterministic finite automaton, there is always a unique minimal non-deterministic finite automaton equivalent to it.
 - (xx) ----- Let π be the ratio of the circumference of a circle to its diameter. Then π is recursive.
 - (xxi) ----- The Kleene closure of any recursive language is recursive.

- (xxii) ----- If $\mathcal{P} = \mathcal{NP}$, then all one-way encoding systems are breakable in polynomial time.
- (xxiii) ----- A language L is in \mathcal{NP} if and only if there is a polynomial time reduction of L to SAT.
- (xxiv) ----- The intersection of any context-free language with any regular language is context-free.
- (xxv) ----- Let L be the set of all strings of the form $\langle G_1 \rangle \langle G_2 \rangle$ where G_1 and G_2 are equivalent context-free grammars. Then L is recursively enumerable.
- (xxvi) ----- If L_1 reduces to L_2 in polynomial time, and if L_2 is \mathcal{NP} and L_1 is \mathcal{NP} -complete, then L_2 is \mathcal{NP} -complete.
- (xxvii) ----- The question of whether two regular expressions are equivalent is \mathcal{NP} -complete. (Do not guess. Look it up.)
- (xxviii) ----- Every language which is accepted by some non-deterministic machine is accepted by some deterministic machine.
- (xxix) ----- The language of all regular expressions over $\{a, b\}$ is a regular language.
- (xxx) ----- The equivalence problem for C++ programs is recursive.
- (xxx1) ----- Every function that can be mathematically defined is recursive.
- (xxx2) ----- A language is L is \mathcal{NP} if and only if there is a polynomial time reduction of L to Boolean satisfiability (SAT).
- (xxx3) ----- If there is a recursive reduction of the halting problem to a language L , then L is undecidable.
- (xxx4) ----- If there is a recursive reduction of a language L to the halting problem, then L is undecidable.
- (xxx5) ----- The set of rational numbers is countable.
- (xxx6) ----- The set of real numbers is countable.
- (xxx7) ----- The set of recursive real numbers is countable.
- (xxx8) ----- There are countably many binary functions.
- (xxx9) ----- There are countably many recursive binary functions.
- (xl) ----- The context-free grammar equivalence problem is co- \mathcal{RE} .
- (xli) ----- Let $L = \{(G_1, G_2)\} : G_1 \text{ and } G_2 \text{ are not equivalent. Then } L \text{ is recursively enumerable.}$
- (xlii) ----- The factoring problem for unary numerals is \mathcal{P} -TIME
- (xl3) ----- If L is a recursively enumerable language, there must be a machine which enumerates L in canonical order.
- (xl4) ----- The set of all positive real numbers is countable.
- (xl5) ----- If L is a context-free language over the unary alphabet, then L must be regular.

(xlv) ----- The union of any two undecidable languages is undecidable.

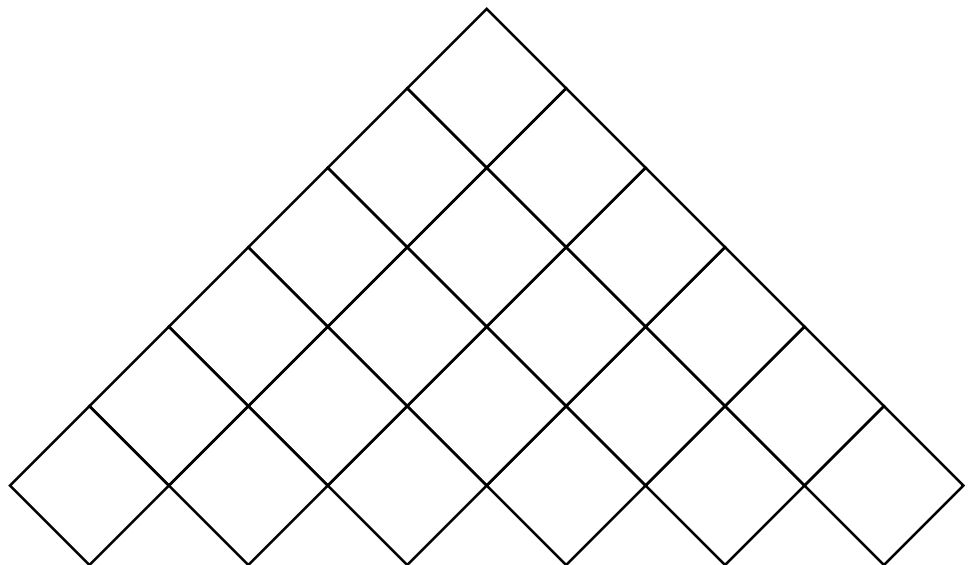
(xlvii) ----- $\text{co-}\mathcal{P}\text{-TIME} = \mathcal{P}\text{-TIME}$

2. Give an unambiguous CFG which generates a language not accepted by any DPDA.
3. Suppose L is a problem such that you can check any suggested solution in polynomial time. Which one of these statements is certainly true?
 - (a) L is \mathcal{P} .
 - (b) L is \mathcal{NP} .
 - (c) L is \mathcal{NP} -complete.
4. Let L be the language of all binary strings where each 0 is followed by 1. Draw a DFA which accepts L .
5. Consider the NFA M pictured below. Construct a minimal DFA equivalent to M .

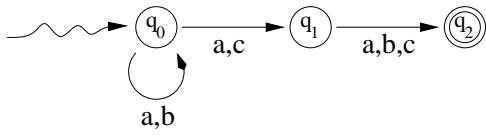


6. Let G_1 be the CF grammar given below. Prove that G_1 is ambiguous by giving two different parse trees for the string $iiwaea$.
 1. $S \rightarrow a$
 2. $S \rightarrow wS$
 3. $S \rightarrow iS$
 4. $S \rightarrow iSeS$
7. The CNF grammar G_2 , given below, is equivalent to the grammar G_1 given in Problem 6. Use the CYK algorithm to prove that $iiwaea$ is generated by G_2 .

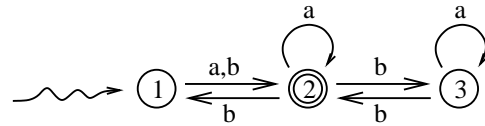
1. $S \rightarrow a$
2. $S \rightarrow WS$
3. $W \rightarrow w$
4. $S \rightarrow IS$
5. $S \rightarrow AB$
6. $A \rightarrow IS$
7. $B \rightarrow ES$
8. $E \rightarrow e$
9. $I \rightarrow i$



8. Give a grammar, with at most 3 variables, for the language accepted by the following NFA.



9. Give a regular expression for the language accepted by the following NFA



10. Let L be the language consisting of all strings over $\{a, b\}$ which have equal numbers of each symbol. Give a CFG for L .
11. Design a DPDA which accepts the language described in Problem 10.
12. Give a context-free language whose complement is not context-free.
13. State the pumping lemma for regular languages.

17. Prove that every decidable language is can be enumerated by some machine in canonical order.

18. Prove that every language which can be enumerated in canonical order by some machine is recursive.

19. Prove that every language which can be enumerated by any machine is accepted by some machine.

20. Give a definition of a recursive real number. (There is more than one correct definition.)

21. Which of these languages (problems) are **known** to be \mathcal{NP} -complete? If a language, or problem, is known to be \mathcal{NP} -complete, fill in the first circle. If it is either known not to be \mathcal{NP} -complete, or if whether it is \mathcal{NP} -complete is not known at this time, fill in the second circle.

- Boolean satisfiability.
- 2-SAT.
- 3-SAT.
- Subset sum problem.
- Traveling salesman problem.
- Dominating set problem.
- C++ program equivalence.
- Partition.
- Regular language membership problem.
- Block sorting.

22. Give a polynomial time reduction of 3-SAT to the independent set problem.

23. Use the pumping lemma to prove that $L = \{a^n b^n : n \geq 0\}$ is not regular.

24. Prove, by induction, that $\sum_{i=0}^n 2^i = 2^{n+1} - 1$. (For example, $1 + 2 + 4 + 8 = 16 - 1$.)