University of Nevada, Las Vegas Computer Science 456/656 Spring 2024

Practice Problems for the Final Examination on April 8, 2024

These problems are taken from Assignment 7 and from final examinations for four earlier semesters.

Try as I might, I could not remove all duplicate questions, so just ignore the duplicates.

Problems Taken from Assignment 7

1. Prove that every decidable language is enumerated in canonical order by some machine.

2. Prove that every language that is enumerated in canonical order by some machine is decided by some other machine..

3. Prove that eny language accepted by any machine can be enumerated by some other machine.

4. Prove that any language which is enumerated by some machine is accepted by some other machine.

5. I have repeatedly stated in class that no language that has parentheses can be regular. For that to be true, there must be parenthetical strings of arbitrary nesting depth. (If you don't know what nesting depth is, look it up.)

   Some programming languages have limitations on nesting depth. For example, I have read that ABAP has maximum nesting depth of 256. (Who would ever want to go that far!)

   The Dyck language is generated by the following context-free grammar. (As usual, to make grading easier, I use a and b for left and right parentheses.)

   1. $S \to aSbS$
   2. $S \to \lambda$

   (a) Use the pumping lemma to prove that the Dyck language is not regular.

   (b) Let D be any positive integer. Let L be the language consisting of all members of the Dyck language whose nesting depth does not exceed D. Prove that L is regular.

6. We known that context-free languages are exactly those which are accepted by push-down automata. We now define a new class of machines, which we call "limited push-down automata." An LPDA is exactly the same as a PDA, but with the restriction that the stack is never allowed to be larger than some given constant. What is the class of languages accepted by limited push-down automata? Think!

7. Prove that every context-sensitive language is decidable. The way to do this is to start with an arbitrary context-sensitive grammar, using the definition I gave in class (that's not the only definition) namely that the right side of any production must be at least as long as the left side, and then design a program which decides whether any given string is generated by that grammar. (If the string has length n, the running time of your program could be very long, maybe an exponentially bounded function of n?)

Problems Taken from the Final Examintion of Fall 2021

1. True/False/Open

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

   (ii) ________ 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.

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

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

   (v) ________ Every language accepted by a non-deterministic machine is accepted by some deterministic machine.

(vi) ________ The language $\{a^n b^n c^n \,:\, n \text{ is prime}\}$ is decidable.

(vii) ________ If a language L is EXP–SPACE complete, L must be decidable.

(viii) ________ $\mathcal{NC} = \mathcal{P}$.

(ix) ________ $\mathcal{P} = \mathcal{NP}$.

(x) ________ EXP-TIME = $\mathcal{P}$-TIME.

(xi) ________ The Boolean Circuit Problem is in $\mathcal{NC}$.

(xii) ________ 3-SAT is $\mathcal{P}$-TIME.

(xiii) ________ Addition of binary numerals is in $\mathcal{NC}$.

(xiv) ________ Every language generated by a general grammar is recursive.

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

(xvi) ________ The problem of whether two given context-free grammars are equivalent is co-$\mathcal{RE}$.

(xvii) ________ The language of all fractions (using base 10 numeration) whose values are less than π is decidable. (Recall that a fraction is a string consisting of two numerals separated by "/".

(xviii) ________ If L is $\mathcal{RE}$ and co-$\mathcal{RE}$, then L must be decidable.

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

(xx) ________ There exists a mathematical proposition that can be neither proved nor disproved.

(xxi) ________ If a Boolean expression is satisfiable, there is a $\mathcal{P}$–TIME proof that it's satisfiable.

(xxii) ________ Every subset of any enumerable set is enumerable.

(xxiii) ________ Every subset of any recursively enumerable set is recursively enumerable.

(xxiv) ________ The binary numeral factorization problem is co-$\mathcal{NP}$.

2. Every language, or problem, falls into exactly one of these categories. For each of the languages, write a letter indicating the correct category.

A Known to be $\mathcal{NC}$.
B Known to be $\mathcal{P}$–TIME, but not known to be $\mathcal{NC}$.
C Known to be $\mathcal{NP}$, but not known to be $\mathcal{P}$–TIME and not known to be $\mathcal{NP}$–complete.
D Known to be $\mathcal{NP}$–complete.
E Known to be $\mathcal{P}$–SPACE but not known to be $\mathcal{NP}$
F Known to be EXP–TIME but not nown to be $\mathcal{P}$–SPACE.
G Known to be EXP–SPACE but not nown to be EXP–TIME.
H Known to be decidable, but not nown to be EXP–SPACE.
K $\mathcal{RE}$ but not decidable.

L co-$\mathcal{RE}$ but not decidable.
M Neither $\mathcal{RE}$ nor co-$\mathcal{RE}$.

(a) ________ All C++ programs which halt with no input.

(b) ________ All base 10 numerals for perfect squares.

(c) ________ All context-free grammars that generate the Dyck language.

(d) ________ All configurations of RUSH HOUR from which it's possible to win.

(e) ________ All satisfiable Boolean expressions.

(f) ________ All binary numerals for composite integers. (Composite means not prime.)

3. Give a definition of each term.

   (a) Accept. (That is, what does it mean for a machine to accept a language.)

   (b) Decide. (That is, what does it mean for a machine to decide a language.)

   (c) Canonical order of a language L.

4. Find a minimal DFA equivalent to the NFA shown in Figure 1.

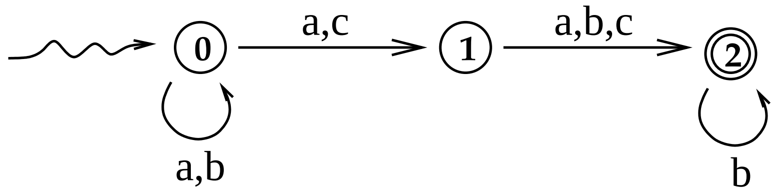


Figure 1: NFA for problems 4 and 5

5. Give a regular grammar with **no more than three variables** for the language accepted by the machine in Figure 1.

6. Give a regular expression for the language accepted by the machine in Figure 2

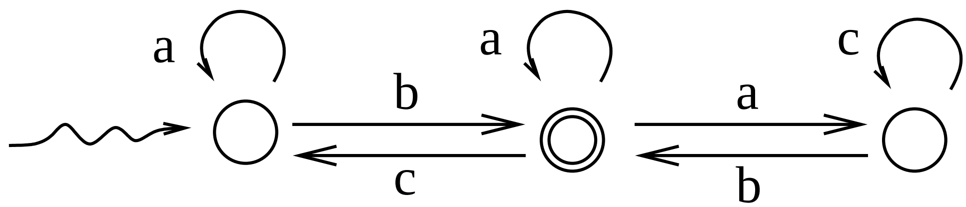


Figure 2: NFA for problem 6.

7. Which class of languages does each of these machine classes accept?

   (a) Deterministic finite automata. ____

   (b) Non-deterministic finite automata. ____

   (c) Push-down automata. ____

   (d) Turing Machines. ____

8. Let $L = \{w \in \{a,b\}^* : \#_a(w) = \#_b(w)\}$, which is generated by the following context-free grammar.

   1. $S \to aSbS$
   2. $S \to bSaS$
   3. $S \to \lambda$

   Draw a PDA which accepts L.

9. The grammar below is an ambiguous CF grammar and is parsed by the LALR parser whose ACTION and GOTO tables are shown here. Write a computation of the parser for the input string *iiwaea*.

   1. $S \to i_2 S_3$
   2. $S \to i_2 S_3 e_4 S_5$
   3. $S \to w_6 S_7$
   4. $S \to a_8$

	a	i	e	w	\$	S
0	$s8$	$s2$		$s6$		1
1					halt	
2	$s8$	$s2$		$s6$		3
3			$s4$		$r1$	
4	$s8$	$s2$		$s6$		5
5			$r2$		$r2$	
6	$s8$	$s2$		$s6$		7
7			$r3$		$r3$	
8			$r4$		$r4$	

Problems Taken from the Final Examintion of Spring 2022

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) There is some PDA that accepts L, where L is 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.

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

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

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

   ______ The intersection of any regular language with any context-free language is context-free.

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

   (vi) ______ If L is a context-free language over an alphabet with just one symbol, then L is regular.

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

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

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

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

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

(xii) ______ There exists a polynomial time algorithm which finds the factors of any positive integer, where the input is given as a binary numeral.

(xiii) ______ The intersection of any two undecidable languages is undecidable.

(xiv) ______ Every $\mathcal{NP}$ language is decidable.

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

(xvi) ______ There exists a $\mathcal{P}$-TIME algorithm which finds a maximum independent set in any graph G.

(xvii) ______ $\mathcal{NC} = \mathcal{P}$.

(xviii) ______ $\mathcal{P} = \mathcal{NP}$.

(xix) ______ $\mathcal{NP} = \mathcal{P}$-SPACE

(xx) ______ EXP-TIME $= \mathcal{P}$-TIME.

(xxi) ______ EXP-SPACE $= \mathcal{P}$-SPACE.

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

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

(xxix) ______ Primality, using binary numerals, is $\mathcal{P}$-TIME.

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

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

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

(xxxiii) ______ Every language generated by an unrestricted grammar is recursive.

(xxxiv) ______ 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.

(xxxv) ______ If P is a mathematical proposition that can be written using a string of length n, and P has a proof, then P must have a proof whose length is $O(2^{2^n})$.

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

(xxxvii) ______ Every bounded function is recursive.

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

(xxxix) ______ 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.

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

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

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

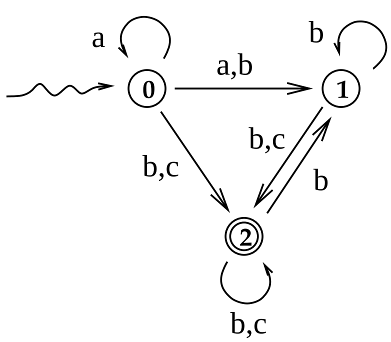
(xliii) ________ Every subset of a regular language is regular.

(xliv) ________ The computer language C++ has Turing power.

(xlv) ________ If L is any $\mathcal{P}$-TIME language, there is an $\mathcal{NC}$ reduction of L to the Boolean circuit problem.

(xlvi) ________ The binary integer factorization problem is co-$\mathcal{NP}$.

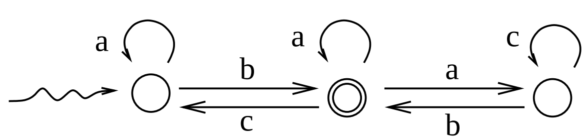
2. Construct a minimal DFA equivalent to the NFA shown below.



3. Find an NFA which accepts the language generated by this grammar.

$$S \to aA|cS|cC$$
$$A \to aA|bS|cB|\lambda$$
$$B \to aA|cB|bC|\lambda$$
$$C \to aB$$

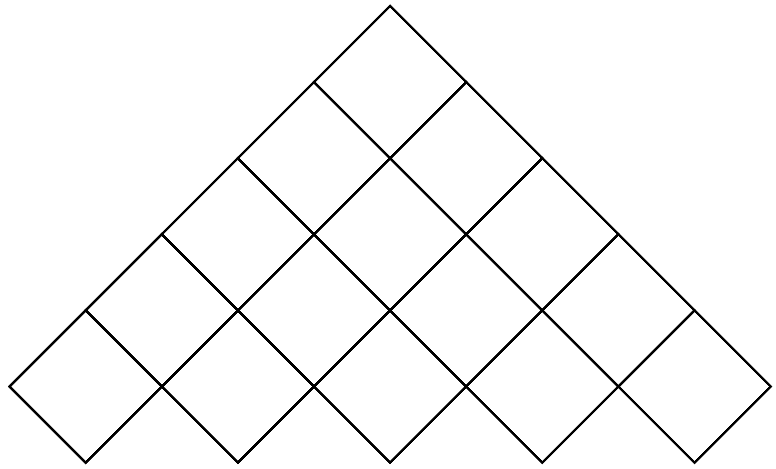
4. Give a regular expression which describes the language accepted by this NFA.



5. Use the CYK algorithm to decide whether $abcab$ is generated by the CNF grammar:

   $S \to AB \,|\, BC \,|CA$
   $A \to a$
   $B \to SA \,|\, SS \,|\, b$
   $C \to c$

   by filling in the matrix.



6. The LALR parser given for this grammar:
   1. $E \to E -_2 E_3$
   2. $E \to E *_4 E_5$
   3. $E \to x_6$

   contains errors, meaning that it might parse a string in a manner that would be considered incorrect by your programming instructor. Find those errors and correct them.

	x	$-$	$*$	$\$$	E
0	$s6$				1
1		$s2$	$s4$	$halt$	
2	$s6$				3
3		$r1$	$r1$	$r1$	
4	$s6$				5
5		$s2$	$s4$	$r2$	
6		$r3$	$r3$	$r3$	

7. Prove that the grammar given in Problem 7 is ambiguous by giving two different leftmost derivations for some string. (If you simply give two different parse trees, you have not answered the question.)

8. State the pumping lemma for regular languages.

9. Give the verifier-certificate definition of the class $\mathcal{NP}$.

10. What is the importance nowadays of $\mathcal{NC}$?

11. What complexity class contains sliding block problems?

12. Give a polynomial time reduction of the subset sum problem to the partition problem.

13. Label each of the following sets as countable or uncountable.

    (i) ________________ The set of integers.

    (ii) ________________ The set of rational numbers.

    (iii) ________________ The set of real numbers.

    (iv) ________________ The set of binary languages.

    (v) ________________ The set of co-$\mathcal{RE}$ binary languages.

    (vi) ________________ The set of undecidable binary languages.

    (vii) ________________ The set of functions from integers to integers.

    (viii) ________________ The set of recursive real numbers.

Problems Taken from the Final Examintion of Fall 2022

1. True/False/Open

    (i) ____ Every subset of a regular language is decidable.

    (ii) ____ The intersection of any two $\mathcal{NP}$ languages is $\mathcal{NP}$. **Think!**

    (iii) ____ Every language accepted by a non-deterministic machine is accepted by some deterministic machine.

(iv) ________ $\mathcal{NC} = \mathcal{P}$.

(v) ________ $\mathcal{P} = \mathcal{NP}$.

(vi) ________ The Boolean Circuit Problem (CVP) is in $\mathcal{NC}$.

(vii) ________ The independent set problem is $\mathcal{P}$-TIME.

(viii) ________ If S is a recursive set of positive integers, then $\sum_{n \in S} 2^{-n}$ must be a recursive real number.

(ix) ________ Multiplication of matrices with binary numeral entries is $\mathcal{NC}$.

(x) ________ Equivalence of regular expressions is decidable.

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

(xii) ________ Equivalence of context-free grammars is co–$\mathcal{RE}$.

(xiii) ________ The language consisting of all fractions whose values are less than the natural logarithm of 5.0 is recursive.

(xiv) ________ If L is in $\mathcal{RE}$ and also co–$\mathcal{RE}$, then L must be decidable.

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

(xvi) ________ Every sliding block problem is $\mathcal{P}$–SPACE.

(xvii) ________ There are uncountably many co–$\mathcal{RE}$ languages.

(xviii) ________ If L is any $\mathcal{P}$–TIME language, there is an $\mathcal{NC}$ reduction of L to CVP, the Boolean circuit problem.

(xix) ________ There is a polynomial time algorithm for checking whether an integer is prime.

(xx) ________ Every finite language is regular.

(xxi) ________ If L is a $\mathcal{P}$–TIME language, there is a Turing Machine which decides L in polynomial time.

(xxii) ________ If anyone ever finds a polynomial time algorithm for any $\mathcal{NP}$–complete language, then $\mathcal{P} = \mathcal{NP}$.

(xxiii) ________ RSA encryption is believed to be secure because it is believed that the factorization problem for integers is very hard.

(xxiv) ________ If S is a recursively enumerable set of positive integers, then $\sum_{n \in S} 2^{-n}$ must be a recursive real number.

2. Every language, or problem, falls into exactly one of these categories. For each of the languages, write a letter indicating the correct category.

A Known to be $\mathcal{NC}$.

B Known to be $\mathcal{P}$–TIME, but not known to be $\mathcal{NC}$.

C Known to be $\mathcal{NP}$, but not known to be $\mathcal{P}$–TIME and not known to be $\mathcal{NP}$–complete.
D Known to be $\mathcal{NP}$–complete.
E Known to be $\mathcal{P}$–SPACE but not known to be $\mathcal{NP}$
F Known to be decidable, but not nown to be $\mathcal{P}$–SPACE.
G $\mathcal{RE}$ but not decidable.
H co-$\mathcal{RE}$ but not decidable.
I Neither $\mathcal{RE}$ nor co-$\mathcal{RE}$.

(a) ________ All C++ programs which do not halt if given themselves as input.

(b) ________ All base 10 numerals for perfect squares.

(c) ________ The Dyck language.

(d) ________ $\{\langle G\rangle : L(G)$ is the Dyck language.$\}$

(e) ________ The Jigsaw problem. (That is, given a finite set of two-dimensional pieces, can they be assembled into a rectangle, with no overlap and no spaces.)

(f) ________ Factorization of binary numerals.

3. Find a DFA equivalent to the NFA shown in Figure 1.

4. Give a regular grammar for the language accepted by the machine in Figure 1.

5. Which class of languages does each of these machine classes accept?

   (a) Deterministic finite automata. ____

   (b) Non-deterministic finite automata. ____

   (c) Push-down automata. ____

   (d) Turing Machines. ____

6. Let $L = \{w \in \{a,b\}^* : \#_a(w) = \#_b(w)\}$, that is, each string of L has equal numbers of each symbol. Draw a PDA which accepts L.

7. The grammar below is an ambiguous CF grammar with start symbol E, and is parsed by the LALR parser whose ACTION and GOTO tables are shown here. The ACTION table is missing actions for

the second column, when the next input symbol is the "minus" sign. Fill it in. Remember the C++ precedence of operators. (Hint: the column has seven different actions: s2, s4, r1, r2, r3, r4, and r5, some more than once, and has no blank spaces.)

1. $E \to E -_2 E_3$
2. $E \to -_4 E_5$
3. $E \to E *_6 E_7$
4. $E \to (_8 E_9)_{10}$
5. $E \to x_{11}$

	x	$-$	$*$	(	)	\$	S
0	$s11$			$s8$			1
1			$s6$			halt	
2	$s11$			$s8$			3
3			$s6$		$r1$	$r1$	
4	$s11$			$s8$			5
5			$r2$		$r2$	$r2$	
6	$s11$			$s8$			7
7			$r3$		$r3$	$r3$	
8	$s11$			$s8$			9
9			$s6$		$s6$		
10			$r4$	$r4$	$r4$	$r4$	
11			$r5$		$r5$	$r5$	

8. Prove that any decidable language can be enumerated in canonical order by some machine.

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

Problems Taken from the Final Examintion of Spring 2023

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) ______ There is some PDA that accepts $\left\{w \in \{a,b,c\}^* : \#_a(w) > \#_b(w) > \#_c(w)\right\}$, that is, 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.

   (ii) ______ The intersection of any two context-free languages is context-free.

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

   (iv) ______ The set of palindromes over $\{a,b\}$ is accepted by some DPDA.

   (v) ______ The language $\{a^n b^n c^n \mid n \geq 0\}$ is in the class $\mathcal{NC}$.

   (vi) ______ There exists a polynomial time algorithm which finds the factors of any positive integer, where the input is given as a binary numeral.

   (vii) ______ Every problem that can be mathematically defined has an algorithmic solution.

(viii) ______ The complement of any undecidable language is undecidable.

(ix) ______ $\mathcal{NC} = \mathcal{P}$.

(x) ______ $\mathcal{P} = \mathcal{NP}$.

(xi) ______ The Boolean Circuit Problem (also known as the CVP problem) is in $\mathcal{P}$.

(xii) ______ The Boolean Circuit Problem (CVP) is $\mathcal{NC}$.

(xiii) ______ co-$\mathcal{P} = \mathcal{P}$.

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

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

(xvi) ______ The set of binary numerals for prime numbers is $\mathcal{P}$-TIME.

(xvii) ______ Every language generated by an unrestricted (general) grammar is recursive.

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

(xix) ______ If L is in $\mathcal{RE}$ and also co-$\mathcal{RE}$, then L must be recursive (decidable).

(xx) ______ There is a mathematical proposition that is true but cannot be proved true.

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

(xxii) ______ There is a Turing machine which, when turned on, runs forever, writing the decimal expansion of π (the well-known ratio of the circumference of a circle to its diameter).

(xxiii) ______ The binary integer factorization problem is co-$\mathcal{NP}$.

(xxiv) ______ If L is $\mathcal{NP}$, there is a polynomial time reduction of L to the subset sum problem.

(xxv) ______ The intersection of any two $\mathcal{NP}$ languages is $\mathcal{NP}$.

(xxvi) ______ The intersection of any two co-$\mathcal{NP}$ languages is co-$\mathcal{NP}$.

(xxvii) ______ The intersection of any two co-$\mathcal{RE}$ languages is co-$\mathcal{RE}$.

(xxviii) ______ Multiplication of matrices with binary numeral entries is $\mathcal{NC}$.

(xxix) ______ Every recursively enumerable language is generated by an unrestricted (general) grammar.

(xxx) ______ Equivalence of context-free grammars is co–$\mathcal{RE}$.

(xxxi) ______ The language of all true mathematical statements is recursively enumerable.

(xxxii) ______ The language of all **provably** true mathematical statements is recursively enumerable.

(xxxiii) ______ There are uncountably many undecidable languages over the binary alphabet.

(xxxiv) ______ If there exists a polynomial time algorithm for any $\mathcal{NP}$–complete problem, then $\mathcal{P} = \mathcal{NP}$.

(xxxv) ________ RSA encryption is accepted as secure by most experts, because they believe that the factorization problem for binary numerals is very hard.

(xxxvi) ________ The language of all $\langle G_1 \rangle \langle G_2 \rangle$ such that G_1 and G_2 are CF grammars which are **not** equivalent is co-$\mathcal{RE}$.

(xxxvii) ________ A real number x is recursive if and only if the set of fractions whose values are greater than x is recursive (decidable).

(xxxviii) ________ For any real number x, there exists a machine that runs forever and outputs the string of decimal digits of x.

(xxxix) ________ If a Boolean expression is satisfiable, there is a $\mathcal{P}$–TIME proof that it's satisfiable.

(xl) ________ If there is a solution to a given instance of any sliding block problem, there must be a solution of polynomial length.

(xli) ________ If the Boolean circut problem (CVP) is $\mathcal{NC}$, then $\mathcal{P} = \mathcal{NC}$.

2. (a) Give a context-sensitive grammar for $\{a^n b^n c^n \; : \; n > 0\}$

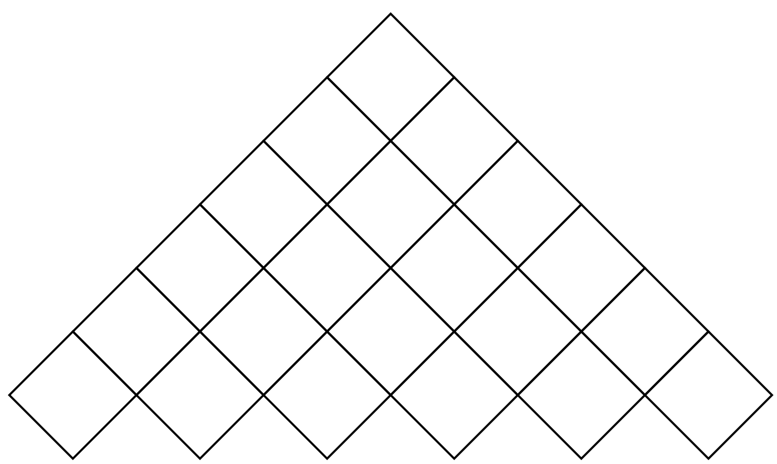
   (b) Using that grammar, give a derivation of the string $aaabbbccc$.

3. Illustrate an NFA which accepts the language generated by this grammar.

$$
\begin{aligned}
S &\to aA|cS|cC \\
A &\to aA|bS|cB|\lambda \\
B &\to aA|cB|bC|\lambda \\
C &\to aB
\end{aligned}
$$

4. Use the CYK algorithm to decide whether $x - x - -x$ is generated by the CNF grammar below, by filling in the matrix.

$$
\begin{aligned}
E &\to ME \\
A &\to EM \\
E &\to AE \\
M &\to - \\
E &\to x
\end{aligned}
$$



5. What is the importance nowadays of $\mathcal{NC}$?

6. Label each of the following sets as countable or uncountable.

   (a) ________________________________ The set of integers.

   (b) ________________________________ The set of rational numbers.

(c) ________________________________ The set of real numbers.

(d) ________________________________ The set of binary languages.

(e) ________________________________ The set of co-RE binary languages.

(f) ________________________________ The set of undecidable binary languages.

(g) ________________________________ The set of functions from integers to integers.

(h) ________________________________ The set of recursive real numbers.

(i) ________________________________ The set of $\mathcal{P}$–SPACE languages over the binary alphabet.

(j) The set of functions from the integers to the binary alphabet $\{0, 1\}$.

7. Let $L = \{w \in \{a, b\}^* : \#_a(w) = \#_b(w)\}$, that is, each string of L has equal numbers of each symbol. Draw a DPDA which accepts L. (Recall that the input to that DPDA must be of the form $w\$$, where $w \in L$ and \$ is the end-of-file symbol.)

8. Consider the CF grammar below. The ACTION and GOTO tables are given below, except that six actions are mising, indicated by question marks. Fill in the missing actions (below the question marks). The actions of your table must be consistent with the precedence of operators in C++.

1. $E \to E -_2 E_3$
2. $E \to -_4 E_5$
3. $E \to E *_6 E_7$
4. $E \to x_8$

	x	$-$	$*$	\$	E
0	$s8$	$s4$			1
1		$s2$	$s6$	HALT	
2	$s8$	$s4$			3
3		?	?	$r1$	
4	$s8$	$s4$			5
5		?	?	$r2$	
6	$s8$	$s4$			7
7		?	?	$r3$	
8	$s8$	$r4$	$r4$	$r4$	

9. Each of the languages listed below falls into one of these categories. Indicate which for each language.

A Known to be $\mathcal{NC}$.

B Known to be $\mathcal{P}$–TIME, but not known to be $\mathcal{NC}$.

C Known to be $\mathcal{NP}$, but not known to be $\mathcal{P}$–TIME and not known to be $\mathcal{NP}$–complete.

D Known to be $\mathcal{NP}$–complete.

E Known to be $\mathcal{P}$–SPACE but not known to be $\mathcal{NP}$

F Known to be decidable, but not known to be $\mathcal{P}$–SPACE.

G $\mathcal{RE}$ but not decidable.

H co-$\mathcal{RE}$ but not decidable.

K Neither $\mathcal{RE}$ nor co-$\mathcal{RE}$.

(a) ________ SAT.

(b) ________ 3-SAT.

(c) ________ 2-SAT.

(d) ________ The Independent Set problem.

(e) ________ The Subset Sum Problem.

(f) ________ The set of all (descriptions of) grammars which generate the Dyck language. That is, all $\langle G \rangle$ such that $L(G)$ is the Dyck language.

(g) ________ All positions of RUSH HOUR from which it is possible to win.

10. Prove that the halting problem is undecidable.