CSC 456/656 Spring 2022 Answers to First Examination February 23, 2022

The entire test is 260 points.

In the questions of this test, $\mathcal{P}$ and $\mathcal{NP}$ denote $\mathcal{P}$-TIME and $\mathcal{NP}$-TIME, respectively.
If L is a language over an alphabet Σ, we define the *complement* of L to be the set of all strings over Σ which are not in L. If $\mathcal{C}$ is a class of languages, we define co-$\mathcal{C}$ to be the class of all complements of members of $\mathcal{C}$.

1. True or False. 5 points each. T = true, F = false, and O = open, meaning that the answer is not known science at this time.
   Each of these problems can be worked using what has been instroduced in class, although some of the problems require thinking, not just memorizing.

   (i) **F** Every subset of a regular language is regular.

   (ii) **O** $\mathcal{NP} = \text{co-}\mathcal{NP}$

   (iii) **F** Context-free grammar equivalence is decidable.

   (iv) **T** The class of regular languages is closed under intersection.

   (v) **T** The class of regular languages is closed under Kleene closure.

   (vi) **T** The class of context-free languages is closed under union.

   (vii) **F** The class of context-free languages is closed under intersection.

   (viii) **T** Every context-free language is in $\mathcal{P}$.

   (ix) **T** We define a set of integers to be *regular* if the set of unary numerals for those integers is a regular language. If S is the set of terms of an arithmetic sequence, (such as $\{21, 25, 29, \ldots\}$) then S is regular.

   (x) **F** The set of prime numbers is a regular set of integers.

   (xi) **T** If a language L is accepted by some PDA, then L must be generated by some context-free grammar.

   (xii) **F** Every PDA is equivalent to some DPDA.

   (xiii) **F** If L has an unambiguous CF grammar, then there must be a DPDA which accepts L.

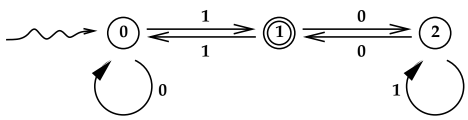
   (xiv) **O** $\mathcal{P} = \mathcal{NP}$.

   (xv) **F** The language of all palindromes over the binary alphabet is inherently ambiguous.

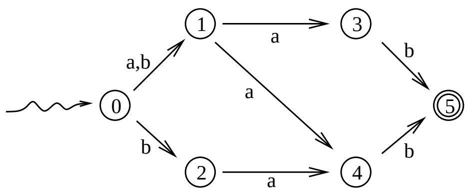
2. [20 points] Name a problem which is known to be $\mathcal{NP}$ and is also known to be co-$\mathcal{NP}$, but is not known to be $\mathcal{P}$.

   The factoring problem for integers, where each integer is represented as its binary numeral.

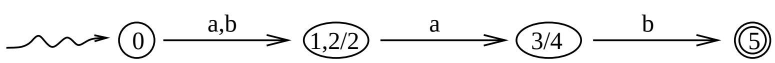
3. [20 points] Let L be the language of all binary strings encoding numbers which are equivalent to 1 modulo 3, where leading zeros are allowed. Draw a DFA which accepts L. (You need exactly three states.)



4. [20 points] Consider the NFA M pictured below.



Construct a minimal DFA equivalent to M.



5. [20 points] Let G be the CF grammar given below. Show that G is ambiguous by drawing two different rightmost derivations for the string $iiwaea$.

1. $S \to a$
2. $S \to wS$
3. $S \to iS$
4. $S \to iSeS$

$$S \Rightarrow iS \Rightarrow iiSeS \Rightarrow iiSea \Rightarrow iiwSea \Rightarrow iiwaea$$

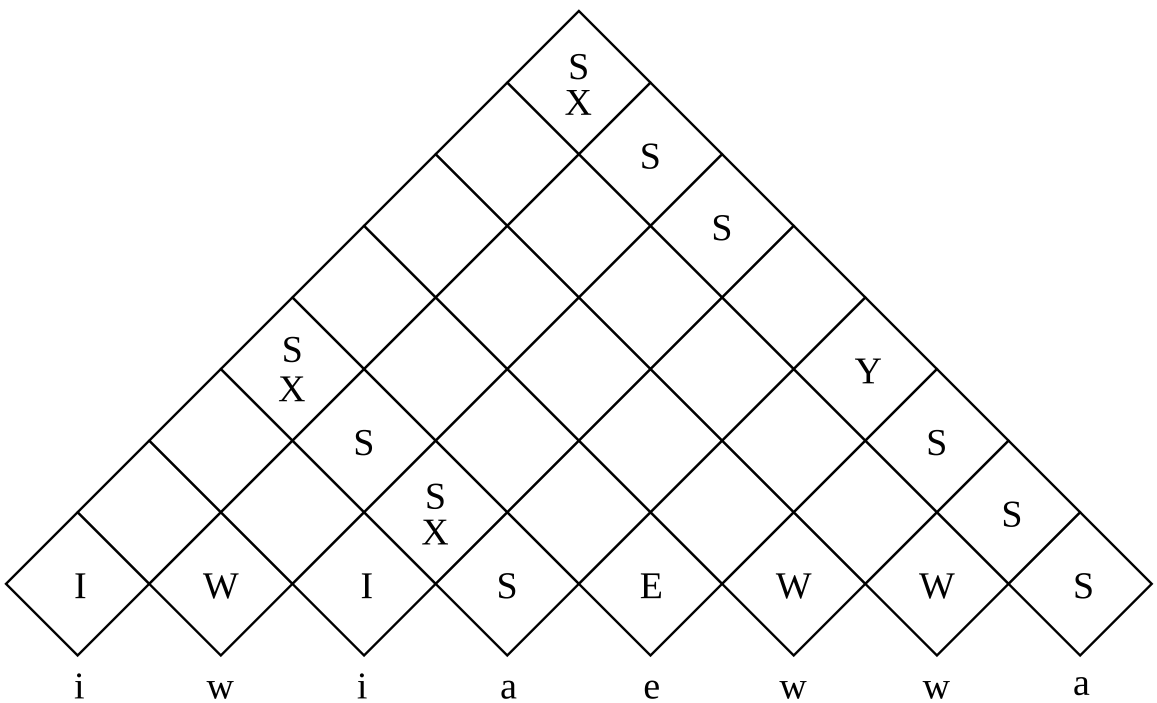
$$S \Rightarrow iSeS \Rightarrow iSea \Rightarrow iiSea \Rightarrow iiwSea \Rightarrow iiwaea$$

This grammar gives a tiny model of a programming language grammar, where

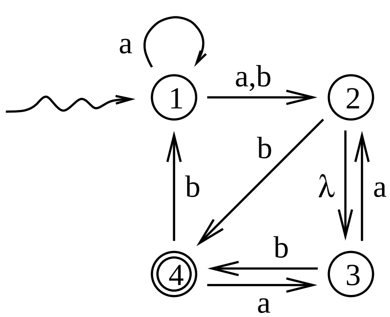
1. S means statement
2. a means assignment statement
3. w means while
4. i means if
5. e means else

6. [20 points] Here is a Chomsky normal form grammar for the language given in problem 5. Use the CYK algorithm to decide whether the string *iwiaewwa* is generated by that grammar, by filling in the diagram below.

1. $S \to a$
2. $S \to WS$
3. $S \to IS$
4. $S \to XY$
5. $X \to IS$
6. $Y \to ES$
7. $I \to i$
8. $E \to e$
9. $W \to w$



7. [20 points] Give a regular grammar for the language accepted by the follwing NFA.

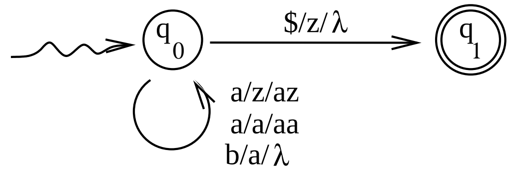


We use the variable S to represent state 1, A for state 2, B for state 3, and C for state 4.

$S \to aS \mid aA \mid bA \mid aC \mid bC$
$A \to bC \mid aA$
$B \to aA \mid aB\, bC$
$C \to bS \mid \lambda$

8. [20 points] Let L be the language over $\{a, b\}$ generated by the following context-free grammar:
$S \to aSbS$
$S \to \lambda$

Design a DPDA which accepts L. (Note that L is just the Dyck language, where a and b are used instead of left and right parentheses.)



9. [25 points]Which of these languages (problems) are **known** to be $\mathcal{NP}$-complete? Mark each blank Yes/No.
(a) SAT **Yes**
(b) Integer factorization **No**
(c) The CYK membership problem **No**
(d) The set of positions of the game RUSH HOUR from which it is possible to win. **No**
(e) 2-SAT **No**

10. [20 points] State the pumping lemma for regular languages *correctly*. Pay close attention to the order in which you write the quantifiers. If you have all the correct words in the wrong order, you still might get no credit.

For any regular language L there exists an integer p such that for any $w \in L$ of length at least p there exist strings x, y, and z such that the following four conditions hold:
1. $w = xyz$,
2. the length of xy is less than or equal to p,
3. y is not the empty string,
4. for any integer $i \geq 0$, $xy^iz \in L$.