CSC 456/656 Fall 2021 Answers to Examination November 17, 2021

The entire test is 270 points.

1. True or False. T = true, F = false, and O = open, meaning that the answer is not known science at this time.

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

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

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

   (iv) **F** Every language generated by an unambiguous context-free grammar is accepted by some DPDA.

   (v) **T** The language $\{a^n b^n c^n d^n \mid n \geq 0\}$ is $\mathcal{NC}$.

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

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

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

   (ix) **T** Multipliction of binary numerals is in $\mathcal{NC}$.

   (x) **F** The problem of whether two given context-free grammars generate the same language is decidable.

   (xi) **T** 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.

   (xii) **F** For any two languages L_1 and L_2, if L_2 is undecidable and there is a polynomial time reduction of L_1 to L_2, then L_1 must be undecidable.

   (xiii) **T** For any two $\mathcal{NP}$ languages L_1 and L_2, if L_1 is $\mathcal{NP}$-complete and there is a polynomial time reduction of L_1 to L_2, then L_2 must be $\mathcal{NP}$-complete.

   (xiv) **O or F** For any two $\mathcal{NP}$ languages L_1 and L_2, if L_2 is $\mathcal{NP}$-complete and there is a recursive reduction of L_1 to L_2, then L_1 must be $\mathcal{NP}$-complete.

   (xv) **T** If L_1 is $\mathcal{NP}$ and L_2 is $\mathcal{NP}$-complete, there is a polynomial time reduction of L_1 to L_2.

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

   (xvii) **T** If L is in $\mathcal{RE}$ and also co-$\mathcal{RE}$, then L must be decidable.

   (xviii) **T** The computer language C++ has Turing power.

   (xix) **T** If an abstract Pascal machine can perform a computation in polynomial time, there must be some Turing machine that can perform the same computation in polynomial time.

2. (5 points each) For each language or problem listed below, fill in the blank with a letter from A to G, where each letter has the following meaning:
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: Undecidable.

(i) **A** All Boolean expressions using parentheses, and, or, and not.

(ii) **A** Addition of binary numerals.

(iii) **B** Dynamic Programming.

(iv) **D** The Independent Set problem.

(v) **A** The language $\{a^n b^n c^n : n > 0\}$.

(vi) **C** Factoring integers expressed as decimal numerals.

(vii) **D** The subset sum problem.

(viii) **E** Rush Hour (the sliding block puzzle).

(ix) **F** All positions in generalized checkers where Black can force a win.

3. (5 points each) For each language or problem listed below, fill in the blank with a letter from H to L, where each letter has the following meaning:
H: Decidable.
I: $\mathcal{RE}$ but not decidable.
K: co-$\mathcal{RE}$ but not decidable.
L: Neither $\mathcal{RE}$ nor co-$\mathcal{RE}$.

(i) **I** The halting problem.

(ii) **K** The diagonal language.

(iii) **I** The complement of the diagonal language.

(iv) **H** 3-SAT

(v) **K** Equivalence of context-free grammars.

(vi) **H** Rush Hour (the sliding block puzzle).

4. [20 points] Give a $\mathcal{P}$-TIME reduction of the subset sum problem to the partition problem.

Let $(K, x_1, x_2, \ldots x_n)$ be an instance of the subset problem. That instance reduces to $(K+1, S-K+1, x_1, x_2, \ldots x_n)$, an instance of the partition problem, where $S = x_1 + \cdots + x_n$.

5. [20 points]

What follows is the LALR praser for the following context-free grammar, where E is the start symbol and the alphabet of terminals is $\{a, +, -, (,)\}$.

1. $E \to E -_2 E_3$
2. $E \to E *_4 E_5$
3. $E \to (_6 E_7)_8$
4. $E \to a_9$

	a	$-$	$*$	(	)	\$	E
0	$s9$			$s6$			1
1		$s2$	$s4$			*halt*	
2	$s9$			$s6$			3
3		$r1$	$s4$		$r1$	$r1$	
4	$s9$			$s6$			5
5		$r2$	$r2$		$r2$	$r2$	
6	$s9$			$s6$			7
7		$s2$	$s4$		$s8$		
8		$r3$	$r3$		$r3$	$r3$	
9		$r4$	$r4$		$r4$	$r4$	

Show the steps of the parser with the input file $(a-a)*a$.

$\$_0$	$(a-a)*a\$$		
$\$_0($	$a-a)*a\$$	$s6$	
$\$_0(_6a_9$	$-a)*a\$$	$s9$	
$\$_0(_6E_7$	$-a)*a\$$	$r4$	4
$\$_0(_6E_7-_2$	$a)*a\$$	$s2$	4
$\$_0(_6E_7 -_2 a_9$	$)*a\$$	$s9$	4
$\$_0(_6E_7 -_2 E_3$	$)*a\$$	$r4$	44
$\$_0(_6E_7$	$)*a\$$	$r1$	441
$\$_0(_6E_7)_8$	$*a\$$	$s8$	441
$\$_0E_1$	$*a\$$	$r3$	4413
$\$_0E_1*_4$	$a\$$	$s4$	4413
$\$_0E_1 *_4 a_9$	$\$$	$s9$	4413
$\$_0E_1 *_4 E_5$	$\$$	$r4$	44134
$\$_0E_1$	$\$$	$r2$	441342
HALT			441342

The output of the LALR parser is 441342, the reverse rightmost derivation of the input string.

6. [20 points] State the pumping lemma for context-free languages.

If L is a context-free language, there exists an integer p such that, for any $w \in L$, if $|w| \geq p$, there exist strings u, v, s, y, z such that:
1. $w = uvxyz$,
2. $|vxy| \leq p$,
3. v and y are not both the empty string,
4. For any integer $i \geq 0$, $uv^ixy^iz \in L$.

7. [20 points] Suppose there is a machine that enumerates a language L in canonical order. Prove that L is decidable.

Case 1: L is finite. Every finite language is decidable.

Case 2: L is infinite. Let $w_1, w_2, \ldots$ be an enumeration of L in canonical order. The following program decides L.

```
Read w.
For i from 1 to ∞
  if(w_i = w)
    HALT accept
  else if(w_i > w) (in the canonical order)
    HALT reject
```

An alternative is to run the loop for i from 1 to 2^{n+1}, where $n = |w|$ This is correct if L is a language of binary strings, but not if the alphabet is larger. I gave full credit.

8. [20 points]

Let L be the language generated by the Chomsky Normal Form (CNF) grammar given below.

$S \to IS$
$S \to XY$
$S \to WS$
$S \to a$
$X \to IS$
$Y \to ES$
$W \to w$
$I \to i$
$E \to e$

Use the CYK algorithm to prove that the string *iwiaea* is a member of L. Use the figure below for your work.