CSC 456/656 Fall 2025 Answers to Second Examination October 23, 2025

The entire exam is 310 points.

Name:____

No books, notes, scratch paper, or calculators. Use pen or pencil, any color. Use the rest of this page and the backs of the pages for scratch paper. If you need more scratch paper, it will be provided.

The *jigsaw puzzle problem* is, given a set of various polygons, and given a rectangular table, is it possibe to assemble those polygons to exactly cover the table?

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.

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

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

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

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

   (vi) **T** The problem of whether a given string is generated by a given context-free grammar is decidable. (By the CYK algorithm.)

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

   (viii) **T** The language $\{a^n b^n c^n d^n \mid n \geq 0\}$ is $\mathcal{P}$–TIME.

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

   (x) **T** Every undecidable problem is $\mathcal{NP}$-hard.

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

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

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

   (xiv) **T** The traveling salesman problem (TSP) is known to be $\mathcal{NP}$-complete.

   (xv) **T** 2-SAT is $\mathcal{P}$-TIME.

(xvi) **T** Primality is $\mathcal{P}$-TIME.

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

(xviii) **T** The language of all fractions (using base 10 numeration) whose values are less than π is decidable.

(xix) **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.

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

(xxi) **F** Every bounded function is recursive.

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

(xxiii) **T** A language is $\mathcal{RE}$ if and only if it is generated by a grammar.

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

(xxv) **F** If a language L is undecidable, then there can be no machine that enumerates L.

(xxvi) **T** There exists a mathematical proposition which is true, but can be neither proved nor disproved.

(xxvii) **T** There is a function which grows faster than any recursive function.

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

(xxix) **T** The binary integer factorization problem is co-$\mathcal{NP}$.

(xxx) **O** There is a polynomial time reduction of the subset sum problem to the binary numeral factorization problem.

(xxxi) **T** The jigsaw puzzle problem is known to be $\mathcal{NP}$ complete.

(xxxii) **F** The jigsaw puzzle problem is known to be $\mathcal{P}$–SPACE complete.

(xxxiii) **T** For any infinite countable sets A and B, there is a 1-1 correspondence between A and B.

(xxxiv) **T** A language L is recursively enumerable if and only if there is a machine which accepts L.

(xxxv) **F** The halting problem is decidable.

2. [20 points] Consider the following annotated context-free grammar.

1. $E \to E -_2 E_3$
2. $E \to -_4 E_5$
3. $E \to (_6 E_7)_8$
4. $E \to x_9$

where x is a placeholder for an arbitrary variable. Fill in all missing entries of the ACTION and GOTO tables for this grammar.

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

3. [10 points] Give a context-free grammar for the language $\{a^n b^m c^n \ : \ n \geq 0, m \geq 0\}$

$S \to aSc$
$S \to B$
$B \to bB$
$B \to \lambda$

4. [10 points] Give two context-free languages whose intersection is not context-free.

There are many examples. Here is one.

$\{x^n y^n z^k \ : \ n, k \geq 0\}$
$\{x^k y^n z^n \ : \ n, k \geq 0\}$

Consider the following context-free grammar G and LALR parser for G.

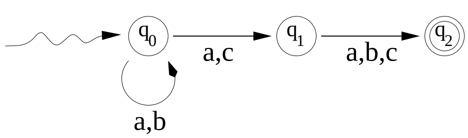
1. $E \to E +_2 E_3$
2. $E \to E \wedge_4 E_5$
3. $E \to (_6 E_7)_8$
4. $E \to x_9$

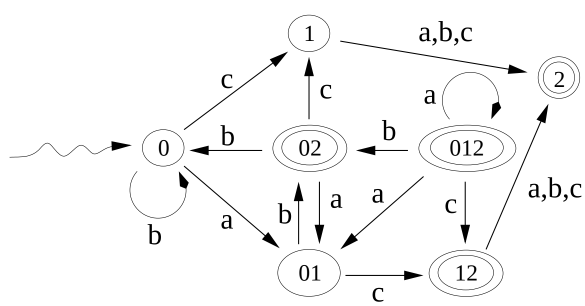
	x	$+$	$\wedge$	(	)	\$	E
0	$s9$			$s6$			1
1		$s2$	$s4$			HALT	
2	$s9$			$s6$			3
3		$r1$	$s4$		$r1$	$r1$	
4	$s9$			$s6$			5
5		$r2$	$s4$		$r2$	$r2$	
6	$s9$			$s6$			7
7		$s2$	$s4$		$s8$		
8		$r3$	$r3$		$r3$	$r3$	
9		$r4$	$r4$		$r4$	$r4$	

Walk through the parser with the input string $x \wedge x \wedge (x + x)$.

STACK	INPUT	OUTPUT	ACTION
$\$_0$	$x \wedge x \wedge (x+x)\$$		
$\$_0 x_9$	$\wedge x \wedge (x+x)\$$		$s9$
$\$_0 E_1$	$\wedge x \wedge (x+x)\$$	4	$r4$
$\$_0 E_1 \wedge_4$	$x \wedge (x+x)\$$	4	$s4$
$\$_0 E_1 \wedge_4 x_9$	$\wedge(x+x)\$$	4	$s9$
$\$_0 E_1 \wedge_4 E_5$	$\wedge(x+x)\$$	44	$r4$
$\$_0 E_1 \wedge_4 E_5 \wedge_4$	$(x+x)\$$	44	$s4$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6$	$x+x)\$$	44	$s6$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 x_9$	$+x)\$$	44	$s9$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 E_7$	$+x)\$$	444	$r4$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 E_7 +_2$	$x)\$$	444	$s2$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 E_7 +_2 x_9$	$)\$$	444	$s9$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 E_7 +_2 E_3$	$)\$$	4444	$r4$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 E_7$	$)\$$	44441	$r1$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 (_6 E_7)_8$	\$	44441	$s8$
$\$_0 E_1 \wedge_4 E_5 \wedge_4 E_5$	\$	444413	$r3$
$\$_0 E_1 \wedge_4 E_5$	\$	4444132	$r2$
$\$_0 E_1$	\$	44441322	$r2$
$\$_0 E_1$	\$	44441322	HALT

5. [20 points] Draw a DFA equivalent to the following three state NFA.



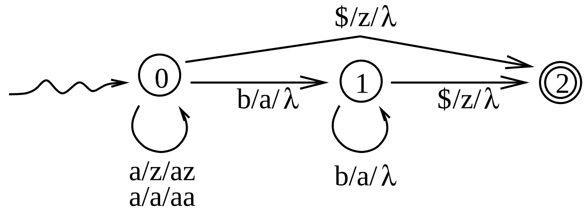


6. [20 points]State the pumping lemma for regular languages.

For any regular language L,
there exists a positive integer p, such that
for any $w \in L$ of length at least p,
there exist strings x, y, z such that:

1. $w = xyz$
2. $|xy| \le p$
3. $|y| > 0$
4. For any integer $i \ge 0$, $xy^iz \in L$.

7. [20 points] Draw a DPDA which accepts the language $\{a^n b^n : n \geq 0\}$



8. [20 points] Give a polynomial time reduction from the subset sum problem to the partition problem.

Let $I = (x_1, x_2, \ldots x_n, K)$ be and instance of the subset sum problem. Let $S = \sum_{i=1}^{n}$. If $K > S$, I trivially has no solution, so we can assume $K \leq S$. Let $R(I) = (x_1, x_2, \ldots x_n, K+1, S-K+1)$, an instance of the partition problem. I has a solution if and only if $R(I)$ has a solution.

9. [20 points] Prove that the halting problem is undecidable.

By contradiction. Assume there exists a machine H such that $H(\langle M \rangle, w) = 1$ if and only if the machine M halts with input w. Let Q be a machine that implements the following program.

Read a machine description $\langle M \rangle$.
If $H(\langle M \rangle, \langle M \rangle) = 1$ run forever.
Else halt.

Run Q with input $\langle Q \rangle$. If it halts, $H(\langle Q \rangle, \langle Q \rangle) = 1$, which means that Q does not halt with input $\langle Q \rangle$, contradiction. Otherwise, Q does halt with input $\langle Q \rangle$, also a contradiction. Thus, H cannot exist, meaning that the halting problem is undecidable.