University of Nevada, Las Vegas Computer Science 456/656 Fall 2026

Assignment 5: Due Saturday November 7, 2026, midnight

Name:_____

You are permitted to work in groups, get help from others, read books, and use the internet. Turn in the assignment as instruced by the Graduate Assistant, Sabrina Wellace, `wallace4@unlv.nevada.edu`

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

   (a) _____ The set of languages over the binary alphabet is countable.

   (b) _____ $\mathcal{P} = \mathcal{NC}$.

   (c) _____ The complement of any $\mathcal{NC}$ language is $\mathcal{NC}$.

   (d) _____ Every context-free language is $\mathcal{NC}$.

   (e) _____ CVP, the circuit value probelem, is $\mathcal{NC}$.

   (f) _____ LALR parsing can be used for any context-free grammar.

   (g) 5_____ LALR parsing cannot be used for any ambiguous grammar.

   (h) _____ The context-free grammar equivalence problem is recursively enumerable.

   (i) _____ If L is any recursively enumerable language and $w \in L$, then there is a proof that $w \in L$.

2. Give a definition of the class $\mathcal{P}$-complete, and then name a language which is $\mathcal{P}$-complete.

3. If the set is countable, enter **C**, if uncountable, enter **U**.

   (a) _____ the set of all rational numbers.

   (b) _____ the set of all irrational numbers.

   (c) _____ The set of all finite sets of integers.

   (d) _____ The set of all sets of integers.

   (e) _____ The set of all finite sets of real numbers.

   (f) _____ The set of all sets of real numbers.

   (g) _____ The set of all languages over the binary alphabet.

   (h) _____ The set of all undecidable languages over the binary alphabet.

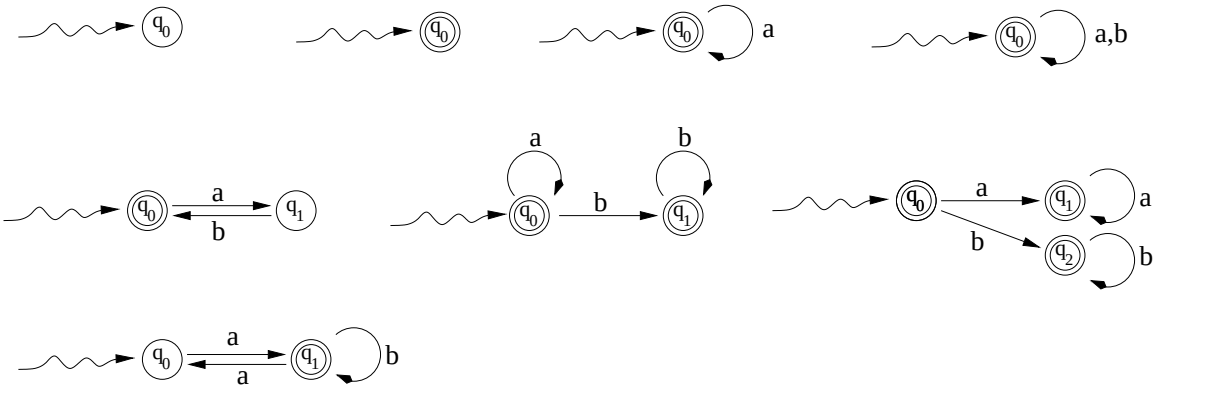
   (i) _____ the set of all recursive real numbers.

   (j) _____ The set of all recursively enumerable languages over the binary alphabet.

   (k) _____ The set of all regular expressions over the binary alphabet.

4. Let L be the language of all strings over the alphabet $\{a, b\}$ which cootain the substring aba. Write a regular expression for L.

5. For each of these regular expressions: $a(b+aa)^*$, $(ab)^*$, a^*b^*, $\emptyset$, λ, $(a+b)^*$, a^*, a^*+b^*, find a corresponding DFA in the figures below.



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

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

8. Prove that every recursively enumerable language is accepted by some machine.

9. The halting problem is defined as follows. Given a machine M and a string w, does M halt if its input is w? We will, later in this course, prove that the halting problem is undecidable. **However:**

   Here is a "proof" that the halting problem is decidable! Explain why this "proof" is invalid.

   Every machine is equivalent to some C++ program, and each such program has finite length. Let $\mathcal{M}$ be the set of all ordered pairs (M, w) such that M is a machine and w is a string. We define the length of such a pair (M, w) to be the length of the C++ program equivalent to M, plus the length of w. There can be only finitely many members of $\mathcal{H}$ of length n. Let $\mathcal{H}$ be the set of pairs (M, w) such that M halts with input w, a subset of $\mathcal{M}$, hence finite.

   For any n, let $T(n)$ be the maximum number of steps it takes M to halt with input w for any $(M, w) \in \mathcal{H}$.

   Given $(M, w) \in \mathcal{M}$, we can determine whether M halts with input w, the following algorithm gives us the answer.

   Let n be the length of (M, w). Run M with input w for $T(n) + 1$ steps or until it halts, whichever comes first. If it halts, the answer is yes, but if it runs $T(n) + 1$ steps, the answer is no. Thus, the halting problem is decidable.

10. Read the handouts lalrhandout1.pdf and lalrhanout2.pdf.

    Consider the following annotated CF grammar G, where E is the start symbol, and the action and goto tables of an LALR parser for G.

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

(a) There are two entries in the action table which together guarantee that multiplication has precedence over addition. There is one entry which guarantees that addition is left associative, and one which guarantees that multiplication is left associative. Identify each of those four entries, by row and column.

	x	$+$	$*$	(	)	\$	
0	$s9$			$s6$			1
1		$s2$	$s4$	$s6$		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		$r4$	$r4$		$r4$	$r4$	

(b) Walk through a computation of the parser with the input string $x * x + x * ((x + x * x) + x)$. I have left the next page blank for this computation.