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

Answers to Assignment 6: Due Saturday November 14, 2026

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. Prove that if L is $\mathcal{RE}$ and $w \in L$, there is a proof that $w \in L$.

   There is a program P which accepts L. The computation of P with input w haLTS, and that computation is a proof that $w \in L$.

2. Prove that the context-free grammar equivalence problem is co-$\mathcal{RE}$. Hint: Look at Problem 1 above.

   Let L be the set of all ordered pairs (G_1, G_2) such that G_1 and G_2 are context-free grammars which are **not** equivalent. We prove that L is $\mathcal{RE}$, which implies that the context free grammar equivalence problem is co-$\mathcal{RE}$. We can assume that the languages generated by G_1 and G_2 are over the alphabet Σ. Let $w_1, \ldots$ be an enumeration of Σ^*. Recall that every context-free language is decidable.

   Theorem: L is $\mathcal{RE}$.

   Proof. Let P be the following program.
   Read G_1, G_2.
   For all $i = 1$ to ∞
   If($w_i \in L(G_1)$ and $w_i \notin L(G_2)$) Halt and Accept
   Else if($w_i \in L(G_2)$ and $w_i \notin L(G_1)$) Halt and Accept

   P accepts L, thus, by Theorem 4 on the handout **CanonEnum,pdf**, L is $\mathcal{RE}$. □

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

   The proofs of this theorem and its converse are found in the handout CanonEnum.pdf.

4. Prove that every language accepted by a machine is recursively enumerable.

   The proofs of this theorem and its converse are found in the handout CanonEnum.pdf.

5. Prove that the halting problem is undecidable. By definition, the halting problem consists of all ordered pairs of the form (P, w), where w is a string and P is a program which halts with input w.

   Theorem: the halting problem is undecidable.

   Proof. By contradiction. Assume that halting problem is decidable. There exists a program H which decides the halting problem. That is, if P is a program and w a string, $H(P, w) = 1$ if P halts with input w and $H(P, w) = 0$ otherwise.

   Let Q be the following program, whose input is any program P.

Read P
If $(H(P,P))$
    Run forever.
Else
Halt

Now execute Q with input Q. If $H(Q,Q) = 1$ then Q runs forever with input Q, contradiction. But if $H(Q,Q) = 0$ then Q halts with input Q, also contradiction.

We conclude that H cannot exist, meaning that the halting problem is undecidable. □

6. Prove that every regular language is $\mathcal{NC}$. Hint: Look at the proof, using the tournament method, that finding the maximum item in a row of items is $\mathcal{NC}$

   The proof is in the henadout **reglrNC.pdf**.

7. Read the handout lalrhandout1.pdf. Staple extra sheets for the problems below..

   (a) Work problem 2 on page 3 of the handout lalrhandout1.pdf.

   (b) Work problem 3 on page 4 of the handout lalrhandout1.pdf.

   (c) Work problem 4 on page 4 of the handout lalrhandout1.pdf.

   (d) Work problem 5 on page 5 of the handout lalrhandout1.pdf.

   (e) Work problem 6 on page 5 of the handout lalrhandout1.pdf.

   (f) Work problem 7 on page 5 of the handout lalrhandout1.pdf.

   (g) Work problem 8 on page 6 of the handout lalrhandout1.pdf.

   (h) Work problem 9 on page 6 of the handout lalrhandout1.pdf.

   The answers are in the handout lalrhandout1ans.pdf.

8. Read the handout lalrhandout2.pdf. Staple extra sheets for the problems below.

   (a) Work problem 1 on page 1 of the handout lalrhandout2.pdf.

   (b) Work problem 2 on page 2 of the handout lalrhandout2.pdf.

   (c) Work problem 3 on page 2 of the handout lalrhandout2.pdf.

   (d) Work problem 4 on page 2 of the handout lalrhandout2.pdf.

   (e) Work problem 5 on page 3 of the handout lalrhandout2.pdf.

   (f) Work problem 6 on page 4 of the handout lalrhandout2.pdf.

   The answers are in the handout lalrhandout2ans.pdf.