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

Answers to Assignment 2: Due Saturday February 28, 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, Shubhashish Kar, `shubhashish.kar@unlv.nevada.edu`

1. True(T) or False(F).

   i **T** If Σ is any alphabet, then Σ^* is a regular language.

   ii **T** Every language is countable.

   iii **F** If Σ is any alphabet, then there are countably many languages over Σ.

   iv **T** If Σ is any alphabet, then there are countably many context-free languages over Σ.

   v **T** The intersection of any regular language with any context-free language is context-free.

   vi **T** Every context-free language over the unary alphabet is regular.

   vii **F** If L is any language, then $L + \{\lambda\} = L$.

   viii **T** If L is any language, then $L\{\lambda\} = L$. (Concatenation)

   ix **T** If L is any language, then $L + \emptyset = L$.

   x **F** If L is any language, then $L\emptyset = L$. $L\emptyset = \emptyset$.

   xi **T** If there is some program which enumerates a language L, then L must be recursively enumerable.

   xii **O** The well-known game *Rush Hour* (look it up on the internet) on a board of unbounded size is $\mathcal{P}$–TIME, that is, there is a $\mathcal{P}$–TIME algorithm which determines whether a solution exists for a particular configuration. That statement is true if $\mathcal{P}$–TIME $\mathcal{P}$–SPACE, false otherwise.

   xiii **F** Every context-free language is generated by some unambiguous context-free grammar. One counter-example is the complement of the language $\{a^n b^n c^n\}$.

2. State the pumping lemma for regular languages.

   I have formatted the lemma to ephasize the nesting of the quantifiers.

   For any regular language L
   There exists a positive number p such that
   For any string $w \in L$ of length at least p
   There exist strings x, y, and z such that the following statements hold:
   1. $xyz = w$.
   2. $|xy| \leq p$.
   3. $|y| > 0$, that is, y is not the empty string.
   4. For any integer $i \geq 0$ $xy^iz \in L$.

3. Use the pumping lemma to prove that $L = \{a^n b^n\}$ is not regular.

   Our proof is by contradiction. Assume that L is regular. Pick an integer $p > 0$ as required by the pumping lemma. Let $w = a^p b^p$, which is a member of L. Note that $|w| = 2p$, hence we can choose string x, y, and z which which satisfy the four conditions in the conclusion of the pump;ing lemma.

   Thus $a^p b^p = xyz$. Since $|xy| \leq p$, the string xy is a substring of the first p symbols of w, hence $y = a^k$ for some k. By the statement that y is not empty, we have $k > 0$.

   Pick $i = 0$. By statement 4, $xy^0 z = xz \in L$ But $xz = a^{p-k} b^p \notin L$, since $p - k \neq p$.

   This is a contradiction. We conclude that L is not regular.

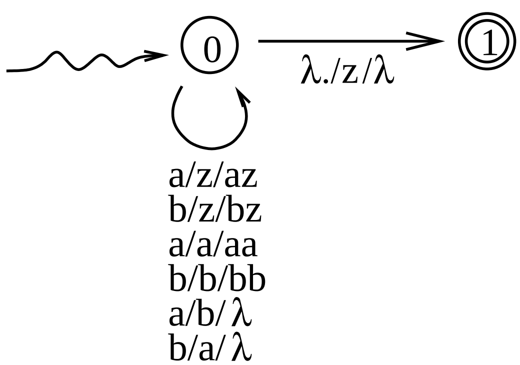
4. Give a context-free grammar for $L = \left\{w \in \{a,b\}^* : \#_a(w) = \#_b(w)\right\}$, that is, all strings over $\{a,b\}$ which have the same number of a's as b's.

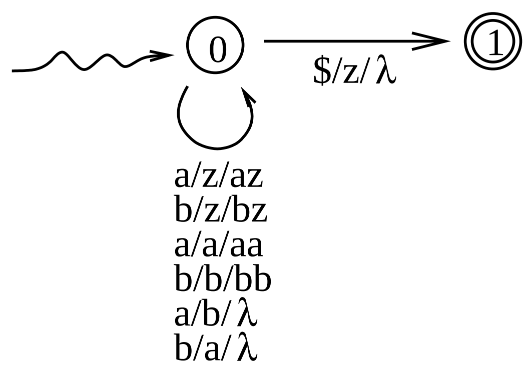
   There is an unambiguous CFG for L, but it's tricky. Here is an ambiguous grammar. The start symbol is S.

   1. $S \to SS$
   2. $S \to aSb$
   3. $S \to bSa$
   4. $S \to \lambda$

5. Sketch a DPDA which accepts $L = \left\{w \in \{a,b\}^* : \#_a(w) = \#_b(w)\right\}$, that is, all strings over $\{a,b\}$ which have the same number of a's as b's.

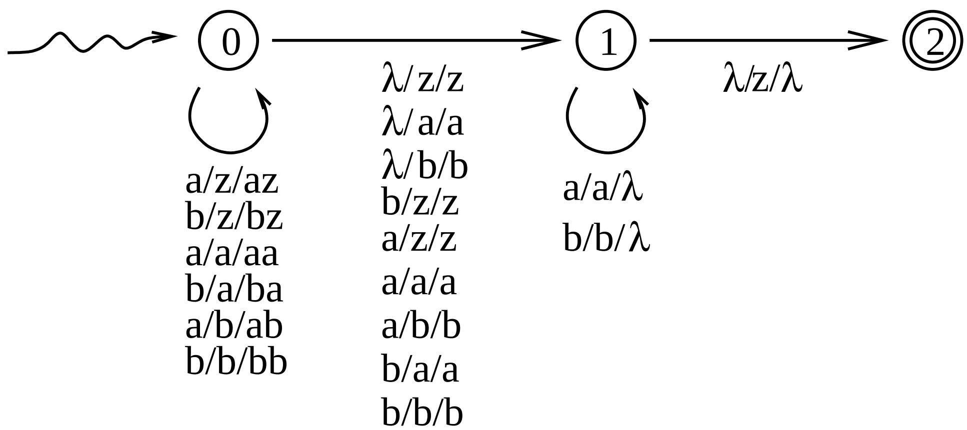
   There are many correct answers. I give two figures, both are push-down automata which accept L, but the first figure shows a non-deterministic PDA, and hence is not an answer to the question given. The the second figure shows a DPDA, as required. This example shows that it is sometimes necessary to consider the end-of-file symbol in the construction of a DPDA.





6. Sketch a PDA which accepts the language of all palindromes over $\{a, b\}$.

   There is no DPDA which accepts the language. The PDA below reads and pushes symbols until it "guesses" it is at the middle of the string, then moves to state 1, then matches stack symbols with the rest of the input.



7. Give two different right-most derivations for the string $x + x * x$ generated by the ambiguous grammar:
   1. $E \to E + E$
   2. $E \to E * E$
   3. $E \to x$

   The start symbol of this grammar is E.

   $E \Rightarrow E{+}E \Rightarrow E{+}E{*}E \Rightarrow E{+}E{*}x \Rightarrow E{+}x{*}x \Rightarrow x{+}x{*}x$

   $E \Rightarrow E * E \Rightarrow E * x \Rightarrow E + E * x \Rightarrow E + x * x \Rightarrow x + x * x$

8. Work problems 1. through 8. on pages 3 through 6 of `lalrhandout1`.

   The answers are in `lalrhandout1ans`.