Midterm

Note

This is a closed-book exam. Each problem accounts for 10 points, unless otherwise marked.

Problems

1. Draw the state diagram of a DFA, with as few states as possible, that recognizes the language $\{w \in \{0,1\}^* \mid w \text{ doesn't contain 101 or 010 as a substring}\}$.

2. Let $L = \{w \in \{0,1\}^* \mid w \text{ contains 101 as a substring or ends with 10}\}$.

   (a) (5 points) Draw the state diagram of an NFA, with as few states as possible, that recognizes L. The fewer states your NFA has, the more points you will be credited for this problem.

   (b) (5 points) Convert the preceding NFA systematically into an equivalent DFA (using the procedure discussed in class). Do not attempt to optimize the number of states, though you may omit the unreachable states.

3. Is the language $\{a^n b^{(n \bmod K)} \mid n > 0 \text{ and } K \text{ is a positive integral constant}\}$ regular? Please justify your answer.

4. Given a language $L \subseteq \Sigma^*$, an equivalence relation R_L over Σ^* is defined follows:

$$xR_Ly \text{ iff } \forall z \in \Sigma^*(xz \in L \leftrightarrow yz \in L).$$

   Suppose $L = \{w \in \{0,1\}^* \mid w \text{ contains the substring } \texttt{101}\}$. What are the equivalence classes determined by R_L? Please give an intuitive verbal description for each of the equivalence classes.

5. An *all*-NFA M is a 5-tuple $(Q, \Sigma, \delta, q, F)$ that accepts $x \in \Sigma^*$ if every possible state that M could be after reading input x is a state from F. Note, in contrast, that an ordinary NFA accepts a string if *some* state among these possible states is an accept state. Please give a formal definition of this computation model, as we did in class for an NFA, including a formal definition of how an all-NFA accepts an input word.

6. Give a context-free grammar that generates the following language: $\{w \in \{a,b,c\}^* \mid$ the number of a's in w equals that of b's or c's$\}$ (no restriction is imposed on the order in which the input symbols may appear). Please make the CFG as simple as possible and explain the intuition behind it.

7. Prove that, if C is a context-free language and R a regular language, then $C \cap R$ is context free. (Hint: combine the finite control part of a PDA and that of an NFA.)

8. Prove *by induction* that, if G is a CFG in Chomsky normal form, then for any string $w \in L(G)$ of length $n \geq 1$, exactly $2n - 1$ steps are required for any derivation of w.

9. Prove each of the following statements:

   (a) (2 points) The class of context-free languages is closed under *union*.

   (b) (4 points) The class of context-free languages is not closed under *intersection*.

   (c) (4 points) The class of context-free languages is not closed under *complement*.

10. Prove, using the pumping lemma, that $\{1^{n^2} \mid n \geq 0\}$ is not context free.

Appendix

- A context-free grammar is in **Chomsky normal form** if every rule is of the form

$$\begin{array}{lcl} A & \rightarrow & BC \quad \text{or} \\ A & \rightarrow & a \end{array}$$

  where a is any terminal and A, B, and C are any variables—except that B and C may not be the start variable. In addition,

$$S \rightarrow \varepsilon$$

  is permitted if S is the start variable.

- (Pumping Lemma for Context-Free Languages)

  If A is a context-free language, then there is a number p such that, if s is a string in A and $|s| \geq p$, then s may be divided into five pieces, $s = uvxyz$, satisfying the conditions:

  1. for each $i \geq 0$, $uv^ixy^iz \in A$,
  2. $|vy| > 0$, and
  3. $|vxy| \leq p$.