Assignment 2, Theory of Computation

1. Prove that the following languages over the given alphabet are not regular.
   (a) $\{w \in \Sigma^* \mid |w| \text{ is a power of } 2\}$, $\Sigma = \{a\}$.
   (b) $\{w \mid w \text{ is a string of balanced parenthesis}\}$, $\Sigma = \{(,)\}$.
   (c) $\{0^n 1 0^n \mid n \geq 0\}$, $\Sigma = \{0, 1\}$.

2. Let w be a string over $\Sigma = \{a, b\}$. The function weight : $\{1, \ldots, |w|\} \to \mathbb{N}$ is defined as follows:

$$\text{weight}(i) = \begin{cases} 0 & \text{if } w(i) = a \\ \text{number of times } a \text{ occurs before i} & \text{if } w(i) = b \end{cases}.$$

   Let $\text{weight}(w) = \Sigma_{i=1}^{|w|} \text{weight}(i)$. Design a minimal DFA to accept exactly those strings w such that weight(w) is even. Prove that your DFA is minimal.

3. Recall the following algorithm we discussed in class for DFA minimization. Input is a DFA $A = (Q', \Sigma, \delta, q_0, F)$.
   (a) After deleting all states that are not reachable from the initial state, suppose $Q = \{q_1, \ldots, q_n\}$ are the remaining states.
   (b) Write down a table of all unordered pairs $\{q_i, q_j\}$ of states, initially unmarked.
   (c) For every pair $\{q_i, q_j\}$, mark $\{q_i, q_j\}$ if $q_i \in F$ and $q_j \notin F$ or vice-versa.
   (d) Repeat until no more pairs can be marked: if there exists an unmarked pair $\{q_i, q_j\}$ such that $\{\delta(q_i, a), \delta(q_j, a)\}$ is marked for some $a \in \Sigma$, then mark $\{q_i, q_j\}$.

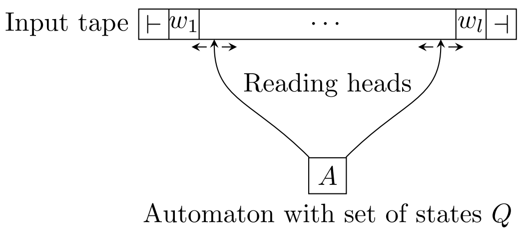
   Assume that we have proved that the algorithm marks a pair $\{q_i, q_j\}$ iff there exists $w \in \Sigma^*$ such that $\hat{\delta}(q_i, w) \in F$ and $\hat{\delta}(q_j, w) \notin F$ or vice-versa.

   (a) Let $\equiv$ be the binary relation on Q such that $q_i \equiv q_j$ if the pair $\{q_i, q_j\}$ is not marked by the above algorithm. Prove that $\equiv$ is an equivalence relation.
   (b) Let us denote by $Q/\equiv$ the set of equivalence classes of Q induced by $\equiv$. Let us denote by $[q]_\equiv$ the equivalence class containing q. Consider the function $\delta' : (Q/\equiv) \times \Sigma \to \mathcal{P}(Q/\equiv)$ defined as $\delta'([q]_\equiv, \sigma) = \{[\delta(q', \sigma)]_\equiv \mid q' \equiv q\}$. Prove that for every $q \in Q$ and $\sigma \in \Sigma$, $\delta'([q]_\equiv, \sigma)$ is a singleton set, thus concluding that $A' = (Q/\equiv, \Sigma, \delta', [q_0]_\equiv, \{[q_f]_\equiv \mid q_f \in F\})$ is a DFA, whose language is equal to $\mathcal{L}(A)$.

(c) For any string $w \in \Sigma^*$, prove that $\hat{\delta}'([q_0]_\equiv, w) = [\hat{\delta}(q_0, w)]_\equiv$.

(d) Let L be the language of the original DFA. Consider the Myhill-Nerode relations $\equiv_L$ and $\equiv_{A'}$. Prove that $\equiv_L$ refines $\equiv_{A'}$, thus concluding that A' is indeed the minimal DFA whose language is L.

4. We have seen two–way automata in class. Consider another extension of automata where there are two reading heads instead of one.



As shown above, there is an input tape on which the input word is written, along with left and right end markers. The first reading head is initially reading the first letter and the second reading head is initially reading the last letter. If the automaton A is in some state q, the first reading head is reading the letter σ_1 and the second reading head is reading the letter σ_2, then the transition function of A will make the automaton change its state to q', make the first reading head move one position to the left or one position to the right, finally make the second reading head move one position to the left or one position to the right. The transition function is of the form $\delta : Q \times \Sigma \times \Sigma \to Q \times \{L, R\} \times \{L, R\}$. The input string is accepted if the automaton ever enters a designated accepted state q_a. The input string is rejected if the automaton even enters a designated rejected state q_r of if the automaton never stops. Are these two–head deterministic automata more powerful than standard DFA? If yes, give an example of a non-regular language accepted by a two–head deterministic automaton. If no, prove that every language accepted by a two-head deterministic automaton is regular.

5. For a language $L \subseteq \Sigma^*$, let $\text{prefix}(L) = \{w \in \Sigma^* \mid w \text{ is a prefix of some string in } L\}$ and $\text{suffix}(L) = \{w \in \Sigma^* \mid w \text{ is a suffix of some string in } L\}$. Are the indices of the equivalence relations $\equiv_{\text{prefix}(L)}$ and $\equiv_{\text{suffix}(L)}$ bounded by some function of the index of $\equiv_L$? If yes, what are the bounds? If no, give demonstrating examples.

6. We know that for every NFA, there exists a language equivalent DFA. Subset construction that gives the equivalent DFA results in an exponential blowup in the number of states in the worst case. Prove that this is unavoidable. Concretely, prove that there exists a family of languages $(L_n)_{n \in \mathbb{N}}$ such that for every n, L_n is the language of some NFA with $O(n)$ states and the minimal DFA for L_n has $\Omega(2^n)$ states.