# Assignment 2

Theory of Computation

Due: 2:30pm on 25th of October, 2024

---

**Instructions**

- Discussion is allowed, and encouraged. However, answers *must* be written by yourself.
- All sources (including discussions) that are used to reach the solution must be mentioned.

---

## Questions

1. Recall the formal definition of a Turing Machine. It is a 7-tuple $(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$ where
   - $Q$ is the set of states.
   - $\Sigma$ is the input alphabet and should not contain the blank symbol.
   - $\Gamma$ is the tape alphabet, is a superset of $\Sigma$ and contains the blank symbol.
   - $\delta : Q \times \Gamma \to Q \times \Gamma \times$ L,R is the transition function.
   - $q_0$ is the start state.
   - $q_{\text{accept}}$ is the accept state.
   - $q_{\text{reject}}$ is the reject state.

   Answer the following questions about the above definition, giving reasons for your answers.

   (a) Can Turing Machines write the blank symbol on the tape? **[2]**

   (b) Can the tape alphabet be the same as the input alphabet? **[2]**

   (c) Can the Turing Machine's head be in the same position at the beginning of two successive steps? **[2]**

   (d) Can a Turing Machine contain a single state? **[2]**

2. (a) Let $A$ be a context-free language and $B$ be a regular language. Show that $A \cap B$ is context-free by constructing a PDA for it. You may assume that a PDA is given for $A$ and a DFA is given for $B$. Argue correctness of your construction. **[4]**

(b) Use (a) to show that

$$\{w \ : \ w \in \{a,b,c\}^* \text{ and } w \text{ contains equal number of a's, b's and c's}\}$$

is not context-free. **[4]**

3. Consider the following languages over $\{0,1\}$.

$$L = \{\langle M \rangle \ : \ M \text{ is a DFA that accepts some string containing an equal number of 0s and 1s.}\}$$

$$L' = \{w \ : \ w \text{ has equal number of 0's and 1s}\} \subseteq \{0,1\}^*$$

$$L'' = \{\langle G \rangle \ : \ G \text{ is a CFG that generates the empty language}\}$$

Using the fact that $L'$ is context-free and $L''$ is decidable, prove that $L$ is decidable. **[4]**

4. Recall the definition for the index of a language.

**Definition.** *Let L be a language over $\Sigma$ and $S \subseteq \Sigma^*$. We say that x and y are* distinguishable *by L, for $x, y \in S$, if $\exists z \in \Sigma^*$ such that either $xz \in L$ or $yz \in L$, but not both. If this is not the case, then x and y are said to be indistinguishable by L.*

*S is said to be pairwise distinguishable by L if for every $x, y \in S$, x and y are pairwise distinguishable. The index of L is the size of the largest S that is pairwise distinguishable by L.* ◊

Also recall your *guess* from Assignment 1.

**Theorem** (Myhill-Nerode Theorem)**.** *The index of a language is equal to the size of the smallest DFA accepting it.*

Now consider the language $L = \{a^i b^j c^k \ : \ i, j, k \geq 0; \text{ if } i = 1 \text{ then } j = k\}$.

(a) Show that $S = \{ab^j \ : \ j \geq 0\}$ is pairwise distinguishable by $L$. **[2]**

(b) Fill in the blank: Using Myhill Nerode Theorem, (a) shows that $L$ is ________. **[1]**

(c) Show that $L$ satisfies all the conditions of the Pumping Lemma. **[5]**

(d) Why does (b) and (c) not lead to a contradiction? **[2]**

5. In this question we will show that single tape Turing Machines that can not write on the portion of the tape containing the input string can only recognise regular languages.

Let $M$ be a single tape Turing Machine that can not write on the portion of the tape containing the input string. Note that the head of $M$, when given an input $x$, will start in the input portion of the tape and then keep moving between that and the non-input portion.

Let $q_x$ denote the state $M$ is in when its head first moves out of the input portion. If this never happens, then $f_x$ is set to $q_{\text{accept}}$ if $M$ accepts $x$, and to $q_{\text{reject}}$ otherwise.

We also define a function $f_x : Q' \to Q$ (for an appropriate $Q' \subseteq Q$) as follows. For any $q \in Q'$, $f_x(q) = q'$ denotes that if $M$ was in state $q$ while moving from a non-input portion to an input portion, then the next time it moves into the non-input portion again, $M$ would be in state $q'$. If that does not happen, $f_x(q)$ is set to $q_{\text{accept}}$ if $M$ accepts $x$ and to $q_{\text{reject}}$ otherwise.

(a) What should $Q'$ be so that the definition of $f_x$ makes sense? **[1]**

(b) Why is $f_x$ well-defined? **[2]**

(c) Show that for two strings $x$ and $y$, if $q_x = q_y$ and for every $q \in Q'$, $f_x(q) = f_y(q)$, then $x$ and $y$ are indistinguishable by $L(M)$. **[4]**

(d) For a fixed $x$, how many choices are there for $q_x$? How many for $f_x$? **[1+2]**

(e) Why does (c) and (d) imply that $L(M)$ is regular? **[2]**

6. Let $E$ be an enumerator that enumerates descriptions of Turing Machines, $\{\langle M_1 \rangle, \langle M_2 \rangle, \ldots\}$, where every $M_i$ is a decider. Prove that there exists a decidable language that is not decided by any decider $M_i$ enumerated by $E$. **[8]**