CS/ECE 374 A ✧ Fall 2023

Homework 4

Due Tuesday, September 19, 2023 at 9pm Central Time

This is the last homework before Midterm 1.

---

1. Recall the following string functions from Homework 1:

$$stutter(w) := \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ aa \bullet stutter(x) & \text{if } w = ax \end{cases} \qquad \mathsf{grow}(w) := \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ \mathtt{0} \cdot \mathsf{grow}(x) & \text{if } w = \mathtt{1}x \\ \mathtt{10} \bullet \mathsf{grow}(x) & \text{if } w = \mathtt{0}x \end{cases}$$

For example, $stutter(\mathtt{1001}) = \mathtt{11000011}$, and $\mathsf{grow}(\mathtt{1001}) = \mathtt{0} \bullet \mathtt{10} \bullet \mathtt{10} \bullet \mathtt{0} = \mathtt{010100}$.

Let L be an arbitrary regular language over the alphabet $\Sigma = \{\mathtt{0}, \mathtt{1}\}$. Prove that the following languages are also regular.

(a) $\textsc{Stutter}(L) = \{stutter(w) \mid w \in L\}$

(b) $\textsc{Unstutter}(L) = \{w \mid stutter(w) \in L\}$

(c) $\textsc{Grow}(L) = \{grow(w) \mid w \in L\}$

(d) $\textsc{Ungrow}(L) = \{w \mid grow(w) \in L\}$

2. Give context-free grammars for the following languages, and clearly explain how they work and the set of strings generated by each nonterminal. Grammars with unclear or missing explanations may receive little or no credit. On the other hand, we do *not* want formal proofs of correctness.

(a) $\left\{\mathtt{0}^a\mathtt{1}\mathtt{0}^b\mathtt{1}\mathtt{0}^c \;\middle|\; b = 2a + 2c\right\}$.

(b) $\left\{\mathtt{0}^a\mathtt{1}\mathtt{0}^b\mathtt{1}\mathtt{0}^c \;\middle|\; 2b = a + c\right\}$.

(c) The set of all palindromes in Σ^* whose lengths are divisible by 7.

*(d) ***Practice only. Do not submit solutions.***

Strings in which the substrings $\mathtt{00}$ and $\mathtt{11}$ appear the same number of times. For example, $\mathtt{1100011} \in L$ because both substrings appear twice, but $\mathtt{01000011} \notin L$.

Yes, you've seen most of these languages before.

*3. ***Practice only. Do not submit solutions.***

Let L_1 and L_2 be arbitrary regular languages over the alphabet $\Sigma = \{0, 1\}$. Prove that the following languages are also regular.

(a) $\textsc{Faro}(L_1, L_2) := \{\mathit{faro}(x,z) \mid x \in L_1 \text{ and } z \in L_2 \text{ with } |x| = |z|\}$, where

$$\mathit{faro}(x,z) := \begin{cases} z & \text{if } x = \varepsilon \\ a \cdot \mathit{faro}(z, y) & \text{if } x = ay \end{cases}$$

For example, $\mathit{faro}(0011, 0101) = 00011011$ and $\textsc{Faro}(0^*, 1^*) = (01)^*$.

(b) $\textsc{Shuffles}(L_1, L_2) := \bigcup_{w \in L_1, y \in L_2} \mathit{shuffles}(w, y)$, where $\mathit{shuffles}(w, y)$ is the set of all strings obtained by shuffling w and y, or equivalently, all strings in which w and y are complementary subsequences. Formally:

$$\mathit{shuffles}(w, y) = \begin{cases} \{y\} & \text{if } w = \varepsilon \\ \{w\} & \text{if } y = \varepsilon \\ \{a\} \bullet \mathit{shuffles}(x, y) \cup \{b\} \bullet \mathit{shuffles}(w, z) & \text{if } w = ax \text{ and } y = bz \end{cases}$$

For example, $\mathit{shuffles}(001, 1) = \{0011, 0101, 1001\}$ and $\mathit{shuffles}(00, 11) = \{0011, 0101, 0110, 1001, 1010, 1100\}$. Finally, $\textsc{Shuffles}(0^*, 1^*) = (0+1)^*$.

Both of these names are taken from methods of mixing a deck of playing cards. A *shuffle* divides the deck into two smaller stacks, and then interleaves those two stacks arbitrarily. A *Faro shuffle* or *perfect shuffle* divides the pack of cards exactly in half, and then interleaves them perfectly; the final deck alternates between cards from one half and cards from the other half. Faro shuffles are the basis of several card tricks.

Solved problems

4. (a) Fix an arbitrary regular language L. Prove that the language $\mathit{half}(L) := \{w \mid ww \in L\}$ is also regular.

Solution: Let $M = (\Sigma, Q, s, A, \delta)$ be an arbitrary DFA that accepts L. We define a new NFA $M' = (\Sigma, Q', s', A', \delta')$ with ε-transitions that accepts $\mathit{half}(L)$, as follows:

$$Q' = (Q \times Q \times Q) \cup \{s'\}$$
$$s' \text{ is an explicit state in } Q'$$
$$A' = \{(h, h, q) \mid h \in Q \text{ and } q \in A\}$$

$$\delta'(s', \varepsilon) = \{(s, h, h) \mid h \in Q\}$$
$$\delta'(s', a) = \varnothing$$

$$\delta'((p, h, q), \varepsilon) = \varnothing$$
$$\delta'((p, h, q), a) = \big\{\big(\delta(p, a), h, \delta(q, a)\big)\big\}$$

M' reads its input string w and simulates M reading the input string ww. Specifically, M' simultaneously simulates two copies of M, one reading the left half of ww starting at the usual start state s, and the other reading the right half of ww starting at some intermediate state h.

- The new start state s' non-deterministically guesses the "halfway" state $h = \delta^*(s, w)$ without reading any input; this is the only non-determinism in M'.
- State (p, h, q) means the following:
  - The left copy of M (which started at state s) is now in state p.
  - The initial guess for the halfway state is h.
  - The right copy of M (which started at state h) is now in state q.
- M' accepts if and only if the left copy of M ends at state h (so the initial non-deterministic guess $h = \delta^*(s, w)$ was correct) and the right copy of M ends in an accepting state.

■

Solution (smartass): A complete solution is given in the lecture notes. ■

Rubric: 5 points: standard langage transformation rubric (scaled). Yes, the smartass solution would be worth full credit.

(b) Describe a regular language L such that the language $double(L) := \{ww \mid w \in L\}$ is *not* regular. Prove your answer is correct.

Solution: Consider the regular language $L = 0^*1$.

Expanding the regular expression lets us rewrite $L = \{0^n 1 \mid n \geq 0\}$. It follows that $double(L) = \{0^n 1 0^n 1 \mid n \geq 0\}$. I claim that this language is not regular.

Let x and y be arbitrary distinct strings in L.

Then $x = 0^i 1$ and $y = 0^j 1$ for some integers $i \neq j$.

Then x is a distinguishing suffix of these two strings, because

- $xx \in double(L)$ by definition, but
- $yx = 0^i 1 0^j 1 \notin double(L)$ because $i \neq j$.

We conclude that L is a fooling set for $double(L)$.

Because L is infinite, $double(L)$ cannot be regular. ■

Solution: Consider the regular language $L = \Sigma^* = (0 + 1)^*$.

I claim that the language $double(\Sigma^*) = \{ww \mid w \in \Sigma^*\}$ is not regular.

Let F be the infinite language 01^*0.

Let x and y be arbitrary distinct strings in F.

Then $x = 01^i 0$ and $y = 01^j 0$ for some integers $i \neq j$.

The string $z = 1^i$ is a distinguishing suffix of these two strings, because

- $xz = 01^i 01^i = ww$ where $w = 01^i$, so $xz \in double(\Sigma^*)$, but
- $yx = 01^j 01^i \notin double(\Sigma^*)$ because $i \neq j$.

We conclude that F is a fooling set for $double(\Sigma^*)$.

Because F is infinite, $double(\Sigma^*)$ cannot be regular. ■

Rubric: 5 points:

- 2 points for describing a regular language L such that $double(L)$ is not regular.
- 3 point for the fooling set proof (standard fooling set rubric, scaled and rounded)

These are not the only correct solutions. These are not the only fooling sets for these languages.

5. Give context-free grammars for the following languages over the alphabet $\Sigma = \{0, 1\}$. Clearly explain how they work and the set of strings generated by each nonterminal. Grammars with unclear or missing explanations may receive little or no credit; on the other hand, we do *not* want formal proofs of correctness.

   (a) In any string, a ***run*** is a maximal non-empty substring of identical symbols. For example, the string $0111000011001 = 0^1 1^3 0^4 1^2 0^2 1^1$ consists of six runs.

   Let L_a be the set of all strings in Σ^* that contain two runs of 0s of equal length. For example, L_a contains the strings 01101111 and 01001011100010 (because each of those strings contains more than one run of 0s of length 1) but L_a does not contain the strings 000110011011 and 00000000111.

Solution:

$S \to ACB$	strings with two blocks of 0s of same length
$A \to \varepsilon \mid X1$	empty or ends with 1
$B \to \varepsilon \mid 1X$	empty or starts with 1
$C \to 0C0 \mid 0D0$	$0^n y 0^n$, where y starts and ends with 1
$D \to 1 \mid 1X1$	starts and ends with 1
$X \to \varepsilon \mid 1X \mid 0X$	all strings: $(0+1)^*$

Every string in L has the form $x0^n y 0^n z$, where x is either empty or ends with 1, y starts and ends with 1, and z is either empty or begins with 1. Nonterminal A generates the prefix x; non-terminal B generates the suffix z; nonterminal C generates the matching runs of 0s, and nonterminal D generates the interior string y.

The same decomposition can be expressed more compactly as follows:

$S \to B \mid B1A \mid A1B \mid A1B1A$	strings with two blocks of 0s of same length
$A \to 1A \mid 0A \mid \varepsilon$	all strings: $(0+1)^*$
$B \to 0B0 \mid 010 \mid 01A10$	$0^n y 0^n$, where y starts and ends with 1

■

Rubric: 5 points = 3 for clearly correct grammar + 2 for clear explanation. These are not the only correct solutions.

(b) $L_b = \{w \in \Sigma^* \mid w \text{ is } \textit{not} \text{ a palindrome}\}$.

Solution:

$$
\begin{aligned}
S &\to 0S0 \mid 0S1 \mid 1S0 \mid 1S1 \mid A && \text{non-palindromes}\\
A &\to 0B1 \mid 1B0 && \text{start and end with different symbols}\\
B &\to 0B \mid 1B \mid \varepsilon && \text{all strings}
\end{aligned}
$$

Every non-palindrome w can be decomposed as either $w = x0y1z$ or $w = x1y0z$, for some substrings x, y, z such that $|x| = |z|$. Non-terminal S generates the prefix x and matching-length suffix z; non-terminal A generates the distinct symbols, and non-terminal B generates the interior substring y. ■

Solution:

$$
\begin{aligned}
S &\to 0S0 \mid 1S1 \mid A && \text{non-palindromes}\\
A &\to 0B1 \mid 1B0 && \text{start and end with different symbols}\\
B &\to 0B \mid 1B \mid \varepsilon && \text{all strings}
\end{aligned}
$$

Every non-palindrome w must have a prefix x and a substring y such that either $w = x0y1x^R$ or $w = x1y0x^R$. Specifically, x is the longest common prefix of w and w^R. In the first case, the grammar generates w as follows:

$$S \rightsquigarrow^* x\,A\,x^R \rightsquigarrow x\,0B1\,x^R \rightsquigarrow^* x0y1x^R = w$$

The derivation for $w = x1y0x^R$ is similar. ■

Rubric: 5 points = 3 for clearly correct grammar + 2 for clear explanation. These are not the only correct solutions.