
HKN CS/ECE 374

Midterm 1 Review

Aditya Chitnis, Abhinav Das, Mahir Morshed

Overview

Strings galore!

- String induction!
- Regular expressions!
- DFAs!
- NFAs!
- CFGs!
- Fooling sets!
- ...and so much more...

String Induction

What is your base case?

How will you formulate your inductive hypothesis? What is your induction variable n ?

How will you use the induction hypothesis to prove the statement true for all n ?

Practice, practice, practice!

1. The *reversal* w^R of a string w is defined recursively as follows:

$$w^R := \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ x^R \bullet a & \text{if } w = ax \text{ for some } a \in \Sigma \text{ and } x \in \Sigma^* \end{cases}$$

1.A. Prove that $(w \bullet x)^R = x^R \bullet w^R$ for all strings w and x . *⟨⟨lab, F14⟩⟩*

Deterministic Finite Automatas (DFA's)

Formal Definition Includes:

- Σ The **finite** alphabet which is used to construct strings in the language
 - Q The set of states in the DFA [visually represented by circles]
 - $s [\in Q]$ The **single** start state [visually represented by source-less arrow to state]
 - $A [\subseteq Q]$ The set of accepting states in
 - δ The transitions between states depending on your next input [visually represented by arrows between states]
 - Accepts a language L *iff* it accepts all strings in L and rejects all strings not in L
-

Non-Deterministic Finite Automatas (NFA's)

- Formal definition same as DFA, only now with ϵ -transitions as well
 - Accepts a language L *iff* it accepts all strings in L and rejects all strings not in L

 - Recall: DFA \Rightarrow NFA \Rightarrow Regex \Rightarrow DFA
 - Nice way to prove regular languages

 - Regex \rightarrow NFA: Thompson's Algorithm
-

—

**Provide a DFA and a regex for the
following language:**

The set of all strings in $\{a,b\}^*$ that
do not contain the substring

aaaaa

Fall 2017 #3

For any language L , let $\text{MID}(L) = \{y \mid xyz \in L \text{ for some } x, y, z \in \Sigma^*\}$ be the language containing all substrings of all strings in L . For example, if $L = \{000, 100, 110, 111\}$, then

$\text{MID}(L) = \{\varepsilon, 0, 00, 000, 1, 10, 100, 11, 110, 111\}$.

Prove that for any regular language L , the language $\text{MID}(L)$ is also regular (suggestion: first describe the necessary construction, and then prove the correctness of the construction).

Closure Properties

- Given regular languages A and B , the following are all regular:
 - $A \cup B$
 - AB
 - A^*
 - $A-B$
 - $A \cap B$ (why?)
 - A^R (why?)
 - Homomorphisms of A (why?)
-

DFAs from NFAs

NFAs do not have a single current state but a set of current states

A DFA accepts the same language as an NFA **if every possible set of states in an NFA is itself a state in the DFA**

An NFA with Q states has an equivalent DFA with at most 2^Q states

A DFA can accept any language that an NFA can!

—

**Draw an NFA and a DFA that
accepts $L = \{w \mid w \text{ ends with}$
 $00 \text{ or } 11\}$**

Regular vs Non-Regular Languages

DFA/NFAs describe regular languages

DFAs/NFAs use fixed memory, even for arbitrarily large inputs

How can we “guess” whether or not a language is regular?
How can we prove it?

$L = \{ w \in \{0,1\}^* \mid w = 0^m 1^n \text{ for } m > n \}$. Is this regular?

Fooling Sets

A fooling set F is a set of strings that are all **distinguishable with respect to a given language**.

Suppose you take a pair of strings x, y from F . **There is some w such that $xw \in L$ and $yw \notin L$.**

This means no pair $x, y \in F$ reaches the same state in a DFA. The DFA must have at least $|F|$ states.

What if F is infinite? This means it is impossible to build a DFA that accepts L . L is not regular.

—

**Prove whether or not this is
regular: $L = \{ww^R \mid w \in \{a,b\}^*\}$**

Bitstrings are another name for strings over the binary alphabet $\{0, 1\}$. Given a bitstring w let $\text{flip}(w)$ be the string obtained by “flipping” each bit of the string, that is changing a 0 to 1 and a 1 to a 0. For example $\text{flip}(010110) = 101001$. Given a language $L \subset \{0, 1\}^*$ we define $\text{flip}(L) = \{\text{flip}(w) \mid w \in L\}$. As an example, if $L = \{0, 0110\}$ then $\text{flip}(L) = \{1, 1001\}$. Given a language $L \in \{0, 1\}^*$ we define $\text{flipsuffix}(L)$ as follows.

$$\text{flipsuffix}(L) = \{u \text{flip}(v) \mid uv \in L\}.$$

As an example, if $L = \{0, 0110\}$ then $\text{flipsuffix}(L) = \{0, \underline{1}, 0110, 011\underline{1}, 01\underline{01}, 000\underline{1}, \underline{1001}\}$ where the underlined segments indicate the flipped suffixes.

- Given a DFA $M = (Q, \{0, 1\}, \delta, s, A)$ for a regular language L , describe a DFA or NFA that accepts the language $\text{flip}(L)$.
- Given a DFA $M = (Q, \{0, 1\}, \delta, s, A)$ for a regular language L , describe a DFA or NFA that accepts the language $\text{flipsuffix}(L)$. Note that $\text{flipsuffix}(L)$ is not necessarily same as $\text{PREFIX}(L) \cdot \text{flip}(\text{SUFFIX}(L))$. The previous part is to help you think about this second part. If you are confident about the solution to this part you can skip the previous part and get full credit.

—

T/F: The language
 $\{0^i 1^j 0^k 1^l \mid i, j, k, l \geq 0\}$
is regular.

Context-Free Grammars

- Terminology note: CFGs *generate* languages
- Formally consist of the following:

Σ The set of terminal symbols [usually all lowercase]

Γ The set of non-terminal symbols [usually all capitals]

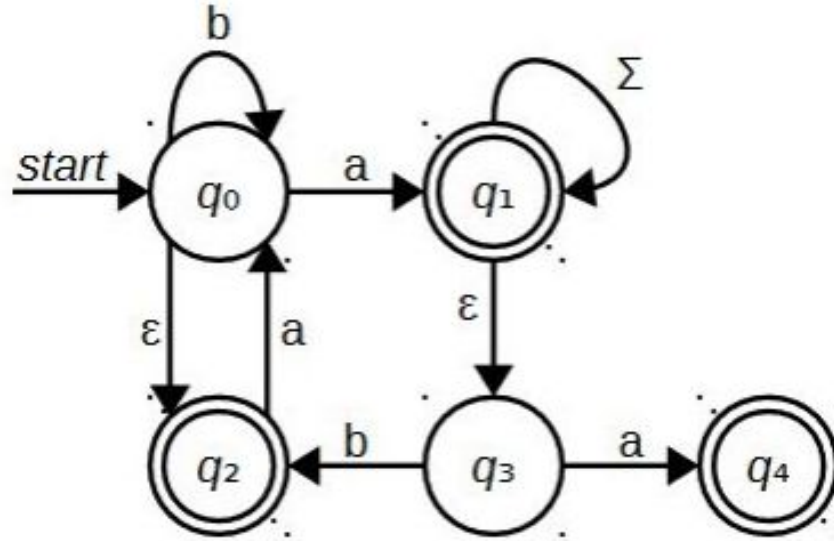
R The set of production rules [usually of form $\langle \text{non-terminal} \rangle \rightarrow \{ \langle \text{symbol} \rangle \}$]

$S \in R$ The **single** start state

- What are these closed under?
-

Give a context-free
grammar for
 $L = \{1^m a 1^n b 1^{m+n} \mid$
 $m, n \geq 0\}$

Convert this NFA into a DFA and regex



—

**Give a context-free
grammar for
 $L = \{a^n b^m c^{2n+1} \mid m, n \geq 0\}$**

Fall 2017 #4

4.B. Provide a regular expression for the following language: The set of all strings in $\{a, b\}^*$ that contain both ab and aa as substrings.

—

**Give a context-free
grammar for
 $L = \{x \subseteq \{0,1\}^* \mid x \text{ of}$
odd length and with
middle symbol a}**