

# FoSAP Lecture Notes

Emma van Emelen

May 9, 2020

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	A Possible Solution (featuring cryptic C code) . . . . .	3
1.2	Finite Automation . . . . .	3
<b>2</b>	<b>Words and Languages</b>	<b>4</b>
2.1	What is a word and what is a language? . . . . .	4
<b>3</b>	<b>The Formal Definition</b>	<b>4</b>

# 1 Introduction

Look at the Following problem:

Input: a string 'w' consisting of 0s and 1s

Examples: 0101, 1001, 00110, 0101010

Question: Are these requirements fullfilled?

→ 11 is not a sub-word

→ w is divisible by 3 in binary

## 1.1 A Possible Solution (featuring cryptic C code)

```
1 int F[] = { 1, 0, 0, 0, 1, 0, 0 };
2 int delta[][2] = {{ 0, 1 }, { 3, 6 }, { 3, 4 }, { 2, 5 }, { 0, 6 }, { 2, 6 }, { 6, 6 }};
3
4 int three_not_11(char *w)
5 {
6     int q = 0;
7     while( *w ) q = delta[ q ][ *w++ - '0' ];
8     return F[q];
9 }
10
```

## 1.2 Finite Automation

The above C code essentially simulates the following finite automaton:

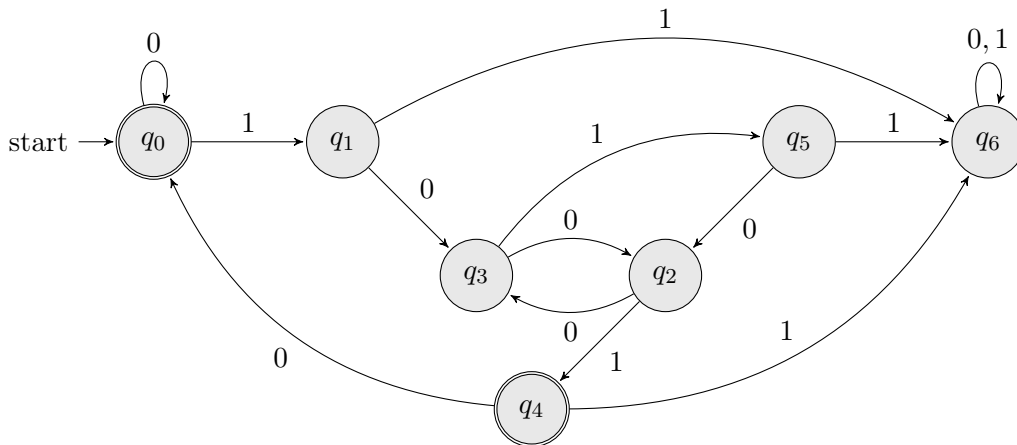


Figure 1: The Finite Automaton

This automaton works in the following way:

$q_0$  - the initial state, as well as an accepting state, we return true, if we end here

$q_4$  - an accepting state, we return true, if we land here

$q_6$  - the default false state, if we have 11 as subword

other - if we end on any other state, we also return false, because it isn't divisible by 3

## 2 Words and Languages

### 2.1 What is a word and what is a language?

An informal answer to the question could be, that a word  $w$  is a concatenation of symbols  $s$  in a specified alphabet  $\Sigma$ . A language  $L$  would then be a set of words defined by a pattern, like  $\{a^{n^2} | n \geq 0\} = \{\epsilon, a, aaaa, aaaaaaaaa, \dots\}$ .

Words have a natural and intuitive mathematical structure. We can concatenate words, split a word, and parse sentences naturally. We can also get a section of the word and remove parts to get a new word.

Flughafen  $\rightarrow$  Flug & Hafen

Baumhaus  $\rightarrow$  Baum & Haus

In the case of natural language, if it uses latin characters, the alphabet and language is easily defined:

$$L = \{s^* | s \in \Sigma\} = \{a, a, they, them, you, ich, er, sie, \dots\} \quad (1)$$

$$\Sigma = \{a, b, c, \dots, x, y, z, A, B, C, \dots, X, Y, Z\} \quad (2)$$

As we can see here, the alphabet  $\Sigma$  is just every uppercase and lowercase latin character and our language  $L$  is no more than a set of words that have one or more character  $s \in \Sigma$ .

## 3 The Formal Definition

- A semigroup  $(H, \circ)$  consists of a set  $H$  and an associative relation  $\circ : H \times H \rightarrow H$
- A *monoid* is a semigroup with a neutral element
- Let  $(M, \circ)$  be a monoid and  $E \subseteq M$   
 $E$  is a generating system of  $(M, \circ)$ , if every  $m \in M$  can be represented as  $m = e_1 \circ \dots \circ e_n$  with  $e_i \in E$ . A neutral element  $e$  is left and right neutral.  $\forall x : e \circ x = x \circ e = x$ .

A great example, using what we already established, would be to define the Alphabet  $\Sigma$  as the generating system of our monoid  $(L, \circ)$  where  $\circ$  is defined as concatenation.