CS 240 — Lecture 17

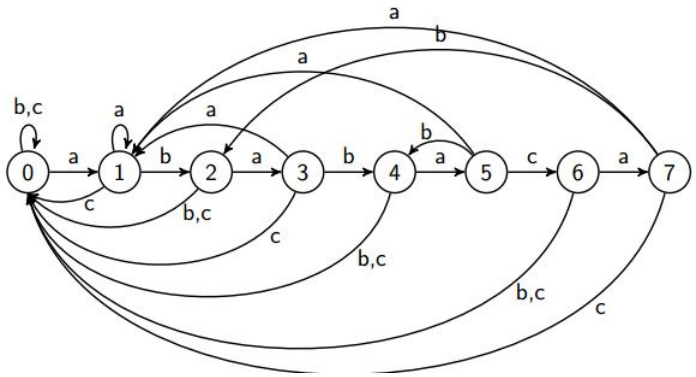
Bartosz Antczak *Instructor: Eric Schost* *March 9, 2017*

17.1 String Matching Algorithms

17.1.1 Finite Automata

(If you took CS 241, this subsection is really a review)
We read our string one character at a time. Starting in an initial state, based on which character we read, we transition into another state. Once we hit a particular state we are done.

Example 17.1.1. *The pattern $P =$ababaca. Once we hit state #7, we are done*



The only difference between the DFAs in CS 240 and the DFAs in CS 241 is that the DFAs in this course require a transition for *every* possible character in our language.
We say there there are m states.

State Transition

To formally define a transition, we define the transition function δ, which takes input of a state q and a character c in Σ:

$$\delta(q, c) = \ell(P[0..q-1]c)$$

where

- $P[0..q-1]c$ is the concatenation of $P[0..q-1]$ and c
- for a string s, $\ell(s) \in \{0, \cdots, m\}$ is the length of the longest prefix of P that is also a suffix of s

Example 17.1.2. *Consider the pattern $P = ababaca$ with $P[0..q] = aba$*

$$\delta(3, a) = \ell(abaa) = 1$$

$$\delta(3, b) = \ell(abab) = 4$$

$$\delta(3, c) = \ell(abac) = 0$$