Data Structures and Algorithms

Prof. Ganesh Ramakrishnan,
Prof. Ajit Diwan,
Prof. D.B. Phatak

Department of Computer Science and Engineering
IIT Bombay

Session: Finite Automaton Algorithm

Introduction: Finite Automata[1]

- A simple machine for string matching
- Scans text T for all occurrences of pattern P

[1] Chapter 32, CLRS, Third Edition

Finite Automata[2]

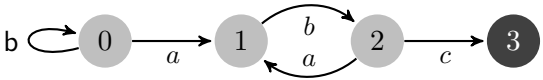
A finite automaton M is a 5-tuple $(Q, q_0, A, \Sigma, \delta)$, where

- Q is a finite set of states
- $q_0 \in Q$ is the start state
- $A \subseteq Q$ is the set of accepting states
- Σ is a finite input alphabet
- δ is the transition function of M from Q x Σ into Q

[2] Chapter 32, CLRS, Third Edition

Illustration

Text $T = ababab$ Pattern $P = abc$

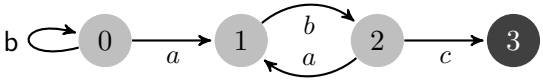


Algorithm for constructing finite automaton

Illustration on $P = ababc$

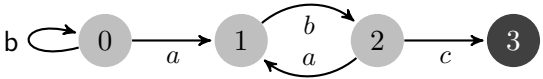
Illustration

Text $T = ababababc$ Pattern $P = abc$



Illustration

Text $T = ababababc$ Pattern $P = abc$

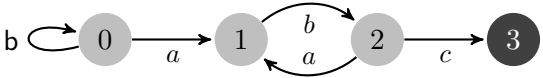


Pattern: abc

State	a	b	c
0	1	0	0
1	1	2	1
2	1	2	3
3	3	3	3

Illustration

Text $T = ababababc$ Pattern $P = abc$



Pattern: abc

State	a	b	c
0	1	0	0
1	1	2	1
2	1	2	3
3	3	3	3

i	-	1	2	3	4	5	6	7
$T[i]$	-	a	b	a	b	a	b	c
δ	0	1	2	1	2	1	2	3

Finite Automaton Algorithm

Algorithm FiniteAutomatonStringMatchingAlgorithm(T,δ, m)
Input Text T of length n and Pattern P of length m
Define s as the shift index to T
$q = 0$
for $i \in (1...n)$ **do**
  $q = \delta(q, T[i])$
  if $q = m$ **then**
    print "Pattern occurs at shift" $i - m$
  end if
end for

Figure: Finite Automaton Algorithm

Thank you