RW778: Implementation and Application of Automata, 2006
Week 1 Lecture 1

L. van Zijl

Department of Computer Science
University of Stellenbosch

2006

Admin and Intro

- Text book: None. Articles to be handed out as needed.
- Lecture times: Monday 10:00–11:00, Tuesday 10:00–11:30
- Lecture room: Committee room, CS dept
- Prerequisites: RW314 (Formal language theory), programming expertise

Evaluation

- Examination: Date and time, format (3hr closed, 3hr open, 24hr take home), weight (between 40% and 60%).
- Practicals: Weekly – Latex report and/or implementation (source code and executable, Linux only). Weight between 40% and 60%.
- Practicals are compulsory – if you fail to hand in even one of the practicals, you receive an ‘incomplete’ for the course.

Course Overview

- Week 1: Introduction. Implementation of automata.
- Week 3: Suffix automata and linear dictionary searches.
- Week 5: CA and free-form graphics modelling.
- Week 7: CA and robot path finding.
- Week 9: CA and random number generation.
- Week 11: $\oplus$-NFAs and ciphering.
- Week 13: Statecharts and real-time modelling.
- Week 14: Exam.

Why IAA?
Introduction

- Renewed interest (more than NLP) – CIAA (1999 onwards)
- More efficient implementations
- Always keep in mind deterministic/nondeterministic; complete/incomplete.

Implementation of Automata

- Transition matrix
- Adjacency lists
- Transition list
- Failure function
- Table compression (self study, for examination purposes)

Implementation of Automata
Transition matrix

- $Q \times \Sigma$ array.
- Space: $O(\text{card}(Q) \times \text{card}(\Sigma))$
- Delay: $O(1)$
- Small alphabet, complete.

Implementation of Automata
Adjacency lists

- $\forall p \in Q$, store list of $(a, \delta(p, a))$
- Space: $O(\text{card}(Q) + \text{card}(E))$
- Delay: $O(\log d)$, where d is max outdegree of any $p \in Q$
- Store lists in arrays

Implementation of Automata
Transition lists

- $\forall e \in \delta$, store edges $e = (p, a, q)$
- Space: $O(\text{card}(E))$
- Best implementation: hashing table defined on (p, a)

Implementation of Automata
Failure function

- Assume complete automaton
- $\gamma : Q \times A \to Q$, $f : Q \to Q$
- (γ, f) represents δ if

$$\delta(p,a) = \begin{cases} \gamma(p,a) & \text{if } \gamma(p,a) \text{ is defined,} \\ \delta(f(p),a), & \text{if } \gamma(p,a) \text{ is undefined and } f(p) \text{ is defi} \\ i, & \text{otherwise} \end{cases}$$

Homework: Write a program to implement a deterministic finite automaton, based on the three methods above (excluding failure function and table compression). Use the Grail input format. Your program must be able to state whether a given word is accepted by the DFA, or not. Remember the report! Highlight issues such as space requirements, ease of programming, and timing issues.