Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen (LEA)
Prof. Dr. Ernst W. Mayr
Moritz Fuchs

Winter Semester 2014/15
Problem set 2
October 14, 2014

Automata and Formal Languages

Due October 21, 2014 before class!

Exercise 1 (JFLAP - 10 points)

Go to http://www.jflap.org/ and download JFLAP

(a) Go to the finite automata section and get familiar with its functionality.

(b) In the lecture we saw, that the determination of a DFA can cause an exponential blowup. Prove the following: For each n there is an NFA with $2n$ states such that any DFA recognizing the same language has at least $n(n-1)$ states. (Hint: It suffices to use a singleton alphabet). Check your construction using JFLAP for $n = 6$.

(c) Using a similar construction as above, give an NFA that yields a blowup of $\mathcal{O}(n^9)$.

Exercise 2 (ϵ-NFA to NFA)

During the ϵ-removal (ϵ-NFAtoNFA), no transition is ever again added to the worklist after it has been added to the worklist, processed and removed from the worklist.
Give an example of an ϵ-NFA and a run of the ϵ-removal algorithm where a transition is put into the worklist twice.

Exercise 3 (Synchronizing Automata - 10 points)

A DFA A is called *synchronizing* if there exists a word $w \in \Sigma^*$ and a state $q_b \in Q$ s.t. for every state $q_a \in Q$: $\hat{\delta}(q_a, w) = q_b$.

(a) Give an exponential time algorithm that decides whether a given DFA A is synchronizing or not.

(b) Prove the following statement:

$$\text{DFA } A \text{ is synchronizing}$$
$$\Leftrightarrow$$
$$\text{For every pair of states } p, q \in Q \text{ there exists a word } w \text{ s.t. } \hat{\delta}(p, w) = \hat{\delta}(q, w).$$

(c) Use the result from (b) to construct a polynomial time algorithm that decides whether a given DFA A is synchronizing or not.

Exercise 4 (Numbers - 10 points)

(a) In the lecture we saw an automaton that recognized some decimal numbers. Give an automaton that recognizes **all** decimal numbers of finite length without sign and leading zeros.

(b) Build a finite automaton that recognizes all rational numbers given as pairs of integers p and q.

(c) Given an integer x and a base b, define an automaton that recognizes any integer divisible by x in base b. Use the most significant digit first encoding.