University of Amsterdam
System and Network Engineering Lab
Dr Clemens Grelck

Compilerbouw
2017/2018
February 14, 2018

Assignment Series 2

Lexicographic Analysis / Abstract Syntax Tree

Consider the following regular expression:

$$a(c|ab)*$$

Assignment 6: Thompson's Construction

Construct a non-deterministic finite automaton (NFA) to recognise the above regular expression using Thompson's Construction.

Assignment 7: Subset Construction

Convert the resulting non-deterministic finite automaton (NFA) of the previous assignment into a deterministic finite automaton (DFA) using Subset Construction.

Assignment 8: Hopcroft's Algorithm

Minimise the resulting deterministic finite automaton (DFA) of the previous assignment using Hopcroft's Algorithm.

Assignment 9: Direct-coded Scanner

Construct a direct-coded scanner based the resulting minimal deterministic finite automaton (DFA) of the previous assignment.

Note:

For Assignments 6–9 it is required to follow the algorithms and not your intuition, despite the simplicity of the regular expression considered. Describe intermediate steps and partial results to illustrate your systematic approach.

Assignment 10: Abstract Syntax Tree Definition

Design an appropriate intermediate representation (abstract syntax tree) for your CiviC compiler by extending the file `ast.xml` of the compiler construction framework. Focus on the language core first and only then consider the extensions.

If you need further attribute types, provide their implementation alongside. Submit the html representation of the abstract syntax tree, as automatically generated by the compiler construction framework (check out the doc directory).

Assignment due date:
Assignments 6–9: February 25, 2018
Assignment 10: February 21, 2018 !!