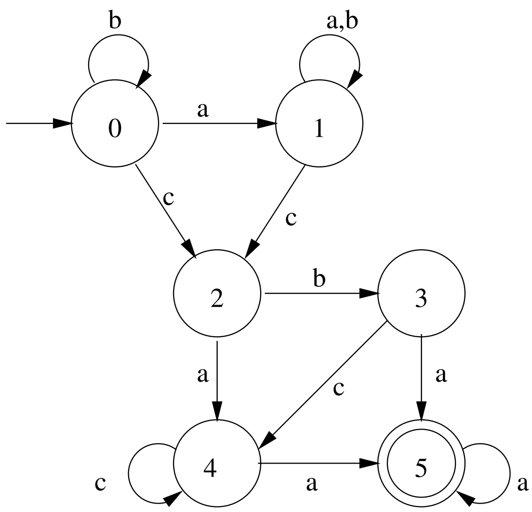
Master of Science in Computer Science - University of Camerino
Compilers A. Y. 2018/2019
Written Test of 23rd July 2019 (Appello V)
Teacher: Luca Tesei

NOTE: Regular expressions should be written using the usual rules of precedence: the $*$ operator has precedence on concatenation, which has precedence on the | operator. The notation $(r)^+$ can be used with the usual meaning.

EXERCISE 1 (10 points)

Consider the following automaton:



1. Express the language accepted by the automaton using a regular expression
2. Is the given automaton minimum? If not, give a minimal equivalent automaton.

EXERCISE 2 (10 points)

Consider the following grammar:

$$\begin{array}{lcl} S & \rightarrow & aA \mid B \mid \epsilon \\ A & \rightarrow & aA \mid C \\ C & \rightarrow & aCb \mid \epsilon \\ B & \rightarrow & CD \\ D & \rightarrow & bD \mid b \end{array}$$

1. Write formally the language generated by the grammar as a set of strings.
2. Prove that the grammar is not SLR(1).

EXERCISE 3 (12 points)

Consider a language of expressions defined recursively as follows:

(i) a, b, and c are expressions;

(ii) if e is an expression then $a(e)$, $b(e)$ and $c(e)$ are expressions.

Your tasks are:

1. Give an LL(1) grammar for the language and provide the parsing table for the top-down parser.
2. Define a Syntax Directed Translation Scheme based on the given grammar. The SDT has to compute, for the starting symbol, three attributes: $n_{\mathbf{a}}$, $n_{\mathbf{b}}$ and $n_{\mathbf{c}}$. The values of the attributes must be the number of a's, b's and c's that occur before an open bracket in the expression. For instance, for the expression $a(a(b(a(c(b(c))))))$ it must result $n_{\mathbf{a}} = 3$, $n_{\mathbf{b}} = 2$ and $n_{\mathbf{c}} = 1$.