

## Lab 4 – Documentation

Iovan Alina Ioana

Github link: <https://github.com/alinaiovan/LFTC/tree/Lab4>

Github link (Bonus point Use FA to detect tokens <identifier> and <integer constant> in the scanner program) : <https://github.com/alinaiovan/LFTC/tree/Lab3>

FiniteAutomata represents the class that has the fields Q, E, q0, F, S:

- Q – the states
- E – the alphabet
- Q0 – initial state
- F – final states
- S – the transitions

The transitions are saved in a dictionary with this form: (p, 0) -> [r] meaning p goes to r with value 0.

We check if a FA is a DFA by taking each key in the dictionary and checking if the destination list has a length > 1.

To check if a sequence is accepted by the FA we iterate through the sequence and we check if the point can be reached by following the FA transitions.

letter ::= a|b|..|z|A|B..|Z

digit ::= 0|1|..|9

character ::= letter | digit

state ::= letter

states ::= "Q = " state { " " state }

alphabet ::= "E = " character { " " character }

initalState ::= "q0 = " character

finalStates ::= "F = " character { " " character }

transition ::= "(" character "," character ")" -> " character

transitions ::= "S = \n" {transition "\n"}

FA-file ::= states "\n" alphabet "\n" initalState "\n" finalStates "\n" transitions