Git: https://github.com/916-Preda-Andrei/Lftc_lab/tree/Lab4

In this lab, I created a new class FiniteAutomata which gets a file in its constructor to get its initialState, finalStates, states, alphabet and transitions. For this, I parse the lines from fa.in to see which parameter it contains. After each parameter, there is an equal sign and then the value of that parameter.

The method getParameters() does this by analyzing the left side of the file line. The function getTransitions(List<String> transitionsList) performs the parsing of the list of all transitions as this is a more complex parameter. The accepts(List<String> dfa) checks if the DFA is accepted by the FA.

The Edge class was created for defining 2 out of 3 elements of a transition, containing the state and an element of the alphabet.

BNF with the form of fa.in:

-file is the BNF form of fa.in

<non_zero_digit> ::= 1|2|...|9

<digit> ::= 0|1|2|...|9

<number> ::= <digit>|<non_zero_digit><number>

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

<character> ::= <letter>|<digit>

<statesList> ::= <letter>|<statesList>,<letter>

<statesLine> ::= states=<statesList>

<initialStateLine> ::= initialState=<letter>

<finalStatesLine> ::= finalStates=<statesList>

<numberList> ::= <number>|<number>,<numberList>

<alphabetLine> ::= alphabet= <numberList>

<transition> ::= (<letter>,<number>,<letter>)

<transitionsList> ::= <transition>| <transition>;<transitionsList>

<transitionsLine> ::= transitions=<transitionsList>

<file> ::= <statesLine> ’\n’ <initialStateLine> ’\n’ <finalStatesLines> ’\n’ <alphabetLine> ’\n’ <transitionsLine>