GitHub Link: <https://github.com/Dalia-Sharra29/FLCD-Homework/tree/master/Lab4>

<https://github.com/Dalia-Sharra29/FLCD-Homework/tree/master/Lab2-3withFA/SymbolTableImplementation>

**Finite Automata**

Requirements:

Write a program that:

1. Reads the elements of a FA (from file)
2. Displays the elements of a finite automata, using a menu: the set of states, the alphabet, all the transitions, the initial state and the set of final states.
3. For a DFA, verify if a sequence is accepted by the FA.

A picture containing text

Description automatically generatedFinite Automata File Format

1. letter ::= 'A' | 'B' | 'C' | ... | 'Z' | 'a' | 'b' | ... | 'z'
2. digit ::= 0|1|2|…|9
3. alphabetCharacter ::= letter | digit
4. state ::= letter
5. transition ::= state ’,’ letter ’,’ state
6. states ::= {state}+
7. alphabet ::= {alphabetCharacter}+
8. initialState ::= state
9. finalStates ::= {state}+
10. file ::= “SET OF STATES\n” states “ALPHABET\n” alphabet “TRANSITIONS\n” transitions “FINAL STATES\n” finalStates “INITIAL STATE\n” initialState

Diagram

Description automatically generated

How it works

1. Parses a given file respecting the rules defined above
2. Mapps it to a FiniteAutomata object which has properties : setOfStates, alphabet, transitionsList, finalStates, initialState
3. Uses a command menu for showing different parts of the automata already initialized

Text

Description automatically generated

1. Can check if the parsed automata is a Deterministic by checking that each transition from the transitions list has the length <= 1.
2. Text

   Description automatically generatedText

   Description automatically generatedText

   Description automatically generatedText

   Description automatically generatedText

   Description automatically generatedCan check if an input sequence is a valid one for the FiniteAutomata: starting from the initial state, we go through each character of the given sequence and we check that we have a next state that we can go to corresponding to the current state and value of the character. If we can’t find a next state, then the sequence is not accepted by the FA and the algorithm stops, returning false. When we reach the end of the sequence if the current state is part of the list of finalStates of the FA, then the sequence is accepted by the FA and the method returns true.

Text

Description automatically generatedText

Description automatically generated

Integration of FA with the Scanner

* Use FA to detect tokens and in the scanner program
  + Made a FA for identifier and integer constant and used them in the scanner function to classify a token