

Lab 4- Finite Automata – documentation

Definition: A finite automaton (FA) is a 5-tuple $M = (Q, \Sigma, \delta, q_0, F)$

where:

- Q - finite set of states ($|Q| < \infty$)
- Σ - finite alphabet ($|\Sigma| < \infty$)
- δ – transition function: $\delta: Q \times \Sigma \rightarrow P(Q)$
- q_0 – initial state $q_0 \in Q$
- $F \subseteq Q$ – set of final states

FiniteAutomata:

- Class arguments: file (String) – path of the file where the input for the finite automata is stored
- Class fields:
 - allStates (List of strings) – holds the finite list of states (Q)
 - inputSymbols (List of strings) – holds the finite alphabet (Σ)
 - initialState (String) – holds the initial state (q_0)
 - finalStates (List of strings) – holds the list of final states (F)
 - transitionFunctions (List of transitions, where a transition consists of a pair of a state and alphabet and the state it leads to)
- Class methods:
 - readLines() → private method that reads the file located at "file" path line by line. It checks if the line starts with "states=", "input_symbols=", "initial_state=", "final_states=", "transition_functions=" and for each of these predefined strings adds the elements that come after "=" to their specific field. For "initial_state" and "final_states" it also checks if the state belongs to the list of states.
 - checkSequence (sequence: String): Boolean → private method that checks if the given sequence is a deterministic finite automata by taking each character from the sequence, finding a pair with the currentState and the character in the

transitionFunctions field, starting with the initialState. If a pair is found, the method goes to the next char in sequence and the currentState becomes the state that has associated the transition. This process is repeated until there are no more items in the sequence, in which case it is checked if the sequence reached the final state and if so, the method returns true, otherwise false. However, the method can return false earlier, if there are items in the sequence that aren't in the allStates field or if there is no transition with the given state and item from sequence.

- showMenu() -> prints a list of options to the user and lets him input his option. If "1" is given as input, the method will print the finite automata. For "2" it will allow the user to input a sequence and check if it is a DFA or not. For "0" it will exit the method and for anything else it will print "Invalid input!".
- toString() - > formats the content of the class in a pretty manner
- FA.in input:
 - FA.in:= states, input_symbols, initial_state, final_states, transition_functions
 - States:= "states=" state+
 - State:=character|letter
 - Input_symbols:="input_symbols=" symbol+
 - Symbol:=character|letter
 - Initial_state:= "initial_state=" state
 - Final_states:="final_states=" state+
 - Transition_functions:="transition_functions=" transition_functions*