Github: https://github.com/pukoarmin/Formal-Langauges-and-Compiler-Design.git

Finite Automata - Documentation:
Language used: Python

=== FiniteAutomata ===
Q: List<String> => Contains the set of states
E: List<String> => Contains the alphabet
S: List<Pair<Pair<String, String>, String>> => Contains the set of states
q0: String => Represents the first state
F: List<String> => Contains the set of final states
---
is_state(String): Boolean => Returns True if the parameter is a state, False otherwise
show_states(): Void => Prints the states
show_final_states(): Void => Prints the final states
show_alphabet(): Void => Shows the alphabet
get_transitions_for(String): returns the set of transitions for the parameter
show_transition_for(String): Shows the set of transitions for the parameter
is_DFA(): Boolean => Returns true if the FA is DFA, False otherwise
isAccepted(String): Boolean => Returns True if the FA is DFA and the parameter is an accepted sequence, False otherwise

BNF:
q0::= q1
q1::= q1 | q3
q2::= q2 | q1
q3::= q3 | q2