

Github link:

→ [https://github.com/Badetto/FLCD\\_Lab2](https://github.com/Badetto/FLCD_Lab2)

## Finite Automata:

Description: Is an algorithm that iterates through the given transitions values given by the user and checks if the given sequence is valid based on the **Private variables**: "States", "Alphabet", "Transitions", "InitialState" and "FinalStates" (their names are suggestive).

→ **ReadFile**: It reads values from a file, thus initializing all 5 Private variables, also if one of the private variables is empty it will throw an exception.

→ **IsDeterministic(): bool**: It returns true if there exists only one transitions for a state-symbol pair, otherwise if there are at least 2 transitions for a state-symbol it violates one of the rules for a FA to be deterministis and it will return false.

→ **CheckSequence (sequence: list<string>): bool**: It returns true if the given sequence is valid, otherwise it will return false. A sequence is valid if you can go from a state to another one via some transitions (based on the values from the sequence taken in order); the first state must be an initial state, and the last state must be a final state.

→ **Display**: One print method for each private variable.

## BNF for input file:

<file> ::= <states> <in\_state> <out\_states> <alphabet> <transitions>

<states> ::= "states" <newline> <state\_list>

<state\_list> ::= <state> <newline> { <state> <newline> }

<in\_state> ::= "in\_state" <newline> <state> <newline>

<out\_states> ::= "out\_states" <newline> <state\_list>

<alphabet> ::= "alphabet" <newline> <symbol\_list>

<symbol\_list> ::= <symbol> <newline> { <symbol> <newline> }

<transitions> ::= "transitions" <newline> <transition\_list>

<transition\_list> ::= <transition> <newline> { <transition> <newline> }

<transition> ::= <state> "," <symbol> "," <state>

<state> ::= <character> { <character> }

<symbol> ::= <character> | "\_"

<character> ::= "a" | "b" | ... | "z" | "A" | "B" | ... | "Z" | "0" | "1" | ... | "9"

<newline> ::= "\n"