Documentation

Github: https://github.com/913-Herlea-Stefan-Alexandru/LFTC

Fa:

- The constructor takes in the file name from which it reads the finite automate
- generate_fa() reads from the given file and returns the states, alphabet, start state, end states and transitions of the finite automate
- verify_sequence(sequence: String) returns True if the given sequence is valide for the fa or False if it isn't

Fa input file:

- Line 1: the states
- Line 2: the alphabet
- Line 3: the start state
- Line 4: the end states
- Line 5+: tranzitions (first character from the line is the state from which we begin, the last character is the stat where we go and the characters in between are the characters from the alphabet through which the transition goes)

Identifiers:

- a sequence of letters, underlines and digits, the first characer being a letter or underline

- rules:

identifier ::= lettandline{charseq}

charseq ::= (lettandline | digit){lettandline | digit}

letter ::= "a" | "b" | ... | "z" | "A" | "B" | ... | "Z"

lettandline ::= letter | "_"

digit ::= "0" | "1" | ... | "9"

Constants:

- integer rule:

nzdigit ::= "1" | "2" | ... | "9"

int ::= "0" | ["+" | "-"]nzdigit{digit}

Integer file:

- symbol ::= “+” | “-” | digit
- alphabet ::= symbol {symbol}
- state ::= letter
- states ::= state {state}
- initialState ::= state
- finalStates ::= states
- transition ::= state alphabet state