FLCD Lab 4 Documentation

Vasc Bogdan-Alexandru

937/2

Finite Automata

Github: https://github.com/BogdanJWK/flcd/tree/master/lab4

Structure

The Finite Automata is implemented as a class with 5 fields and several methods:

- Q: array – list of all states
- E: array – list of all alphabet symbols
- q0: array – list of all start states
- F: array – list of all final states
- S: dictionary – containing all transitions, given as a dictionary having:
  - Key: tuple with two values (state, symbol)
  - Value: next state
- isValid() – returns true if FA is valid, false otherwise
- checkSequence(seq: string) – returns true if sequence is validated, false otherwise

Validating a sequence

Sequence validation is done through a simple iteration. Initially it is assumed that the current state is one of the start states. Each consecutive symbol from the sequence is paired with the current state and if the pair is within the transitions dictionary, the next state is obtained, and it replaced the current state. If the pair is not part of the dictionary, the sequence is invalidated.

EBNF

States = word {word}

initialState = word

finalStates = word { word }

alphabet = word { word }

transitions = word word word

word = character { character }

character = “a” | “b” | … | “z” | … | “A” | “B” | … | “Z” | “0” | “1” | … | “9”