Finite Automata

Documentation

Lab4

Dora-Maria Moraru

GitHub link: https://github.com/DoraMoraru/FLCD

The finite automaton is represented as a class that contains 5 attributes: the set of states (Set<String>), the set of symbols from the alphabet (Set<String>), the initial state (String), the set of final states (Set<String>), the set of transitions which maps a pair of type (source, route) to a destination (Map<Pair<String, String>, List<String>>).

The FA class has a method that reads a FA from a file, which returns the FA created from the input file if is valid or throws an exception otherwise.

The structure of the file is the following:

1. First line: the number of states followed by the set of states
2. Second line: the number of elements from the alphabet followed by the elements of the alphabet
3. Third line: the number of transitions
4. Next lines: the transitions
5. Next line: the initial state
6. Last line: the number of final states followed by the final states

The BNF is the following:

fa = nr_states ‘ ‘ states ‘\n’ nr_alpahbet ‘ ‘ alphabet ‘\n’ nr_transitions ‘\n’ transitions ‘\n’ initial_state ‘\n’ nr_final_states ‘ ‘ final_states

nr_states = digit{digit}

states = state | state ‘ ‘ states

nr_alphabet = digit{digit}

alphabet = symbol | symbol ‘ ‘ alphabet

nr_transitions = digit{digit}

transitions = transition | transition ‘\n’ transitions

transition = state ‘ ‘ symbol ‘ ‘ state

nr_final_states = digit{digit}

final_states = state | state ‘ ‘ final_states

character = letter | digit

string = character {string}

letter = uppercase_letter | lowercase_letter

uppercase_letter = “A” | “B” | … | “Z”

lowercase_letter = “a” | “b” | … | “z”

digit = “0” | “1” | … | “9”

state = string

symbol = string

We also have a method that checks if a sequence is accepted or not by the FA. This is done by going through each symbol from the given sequence and checking that the respective point can be reached by the following the FA transitions.

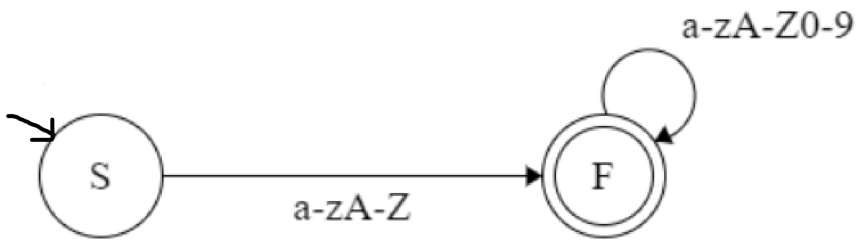
FA.in example:

```
3 S A B
2 0 1
4
S 0 A
A 1 B
A 0 S
B 0 B
S
1 B
```

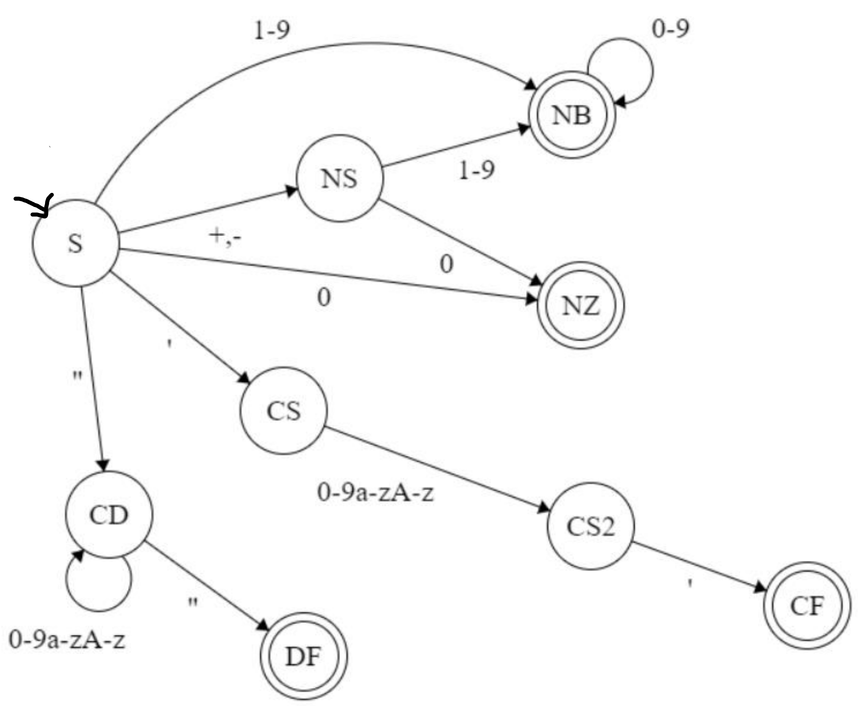
Integration with Scanner

The idea is that regex matching for constants and identifiers is replaced with checking that the FA accepts a given sequence, namely one which represents an identifier/ constant.

The following is the FA for the identifier:



The following is the FA for the constants:



1-9
0-9
NB
NS
1-9
S
+,-
0
0
NZ
'
"
CS
0-9a-zA-z
CD
CS2
'
CF
"
0-9a-zA-z
DF