https://github.com/banuAndrei99/FLCD

```
# Finite Automaton
## __print_states()
    Prints the states of the automaton.
   - pre: states are present in input json
   - post: None

## __print_alphabet()
    Prints the alphabet of the automaton.
   - pre: alphabet is present in input json
   - post: None

## __print_final_states()
    Prints the final states of the automaton.
   - pre: final states are present in input json
   - post: None

## __print_transitions()
    Prints the transitions of the automaton.
   - pre: states are transitions in input json
   - post: None

## check(sequence, current_state)
    Checks if <sequence> can be obtained starting from <current_state>.
   - pre: None
   - post: None

## check_wrapper(sequence)
    Initializes the parameters needed for check() method and print a message based on
the result.
   - pre: None
   - post: None

## Example:
```

```
1. Print states
2. Print alphabet
3. Print final states
4. Print transition dictionary
5. Check sequence
6. Close

>>> 5

Enter a sequence: 110
everything OK
```

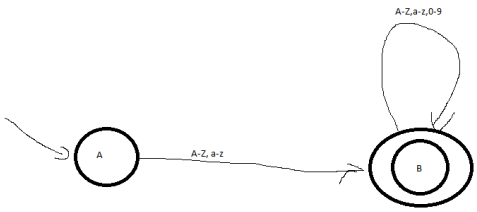
For a DFA that checks if a sequence is a valid identifier

```
identifier ::= letter | letter{letter}{digit}

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

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

BNF



FA::= states, alphabet, initial_states, transitions, final_states

State = A | B

States::= State{State}

Alphabet::= A-Z, a-z, 0-9

Final_states::= B

Transitions::= {State, “:”, Alphabet, B}