5 Automata

Monday, September 11, 2023 11:01 AM

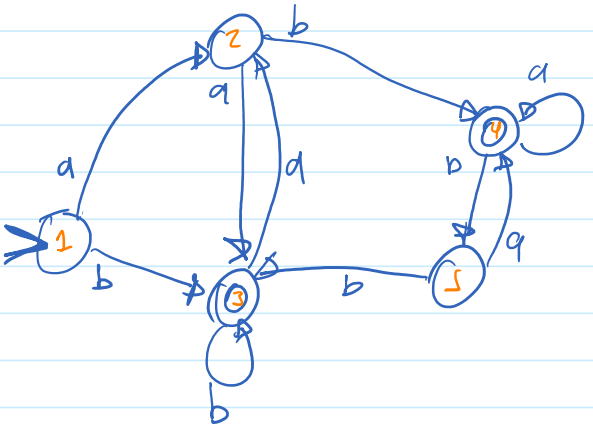
Mathematical Languages.

Problems
- Specification - regex
- Recognition. $w \in L$?

( )
[ ]
{ }
: ?
; !

- Automata.
  - Set of states
    - initial state
    - "accept" states
  - Transition function
    function from
    state, symbol → state.
    symbols from Σ

$\Sigma = \{a, b\}$



How to recognize a Language?

- Given a word w
  we start at the initial state
  and follow the transition function
  for each symbol from w in order.

w = aabbaba → 4

- If you end in an accept state
  $w \in L$!
  otherwise
  $w \notin L$!

$L = \{ab, b, bab, bbbbb\}$

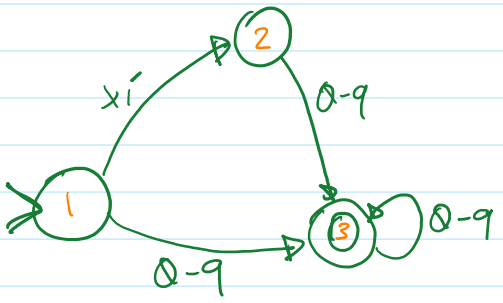
THEOREM for every Language L
Specified by a reg-ex.
There exists an automata that
recognizes L

E.G.

Integer Constants

$(+|-)?[0-9]^+$

Σ = ASCII

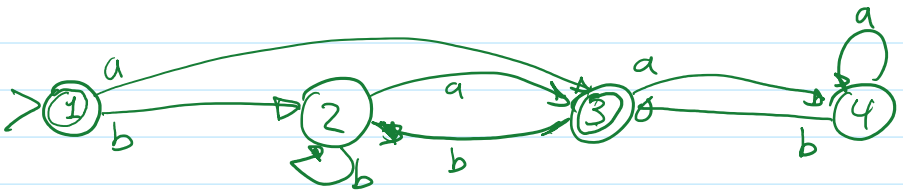


0 -9 27
+82

Note: There exist an algorithm that takes a regex and constructs an automata.

- Encoding an automata.
  - Loop over a switch statement.
  - every case of switch statement will correspond to a state in the automata.
  - the body of each switch statement should correspond to the relevant transition function.

E.G.



L { a, ba, bba }

Pseudocode

```
FUNCTION recognize( string s ) : BOOLEAN
VAR state, I : INTEGER
    c : CHAR
BEGIN
```

```
state := 1;  i := 0

WHILE i < length(s) DO
  c := s[i]
  CASE state OF
    1 : IF c = 'a' THEN state := 3
                   ELSE state := 2
    2 : IF c = 'a' THEN state := 3
                   ELSE state := 2
    3 : IF c = 'a' THEN state := 4
                   ELSE state := 2
    4 : IF c = 'a' THEN state := 4
                   ELSE state := 3
  END
  i := i + 1
END
IF state = 3 OR state = 1 THEN
   RETURN True
ELSE RETURN False
END
```

We could:
- add fail / sink states
- return early
- turn switch into lookup-table.

this algorithm is the basis of our "Lexical analyzer"

C

C++: an octopus made by nailing extra legs onto a dog. – Steve Taylor

—EOF—