5 Automata

Recap:

Mathematical Languages
- ▷ Specification - RegEx
- ▷ Recognition - Automata

Spec:

$$\mathcal{L} = (+|-)?[0-9]+(.[0-9]+)?e(+|-)?[0-9]+$$

Rec:

$$27.3e-7 \in \mathcal{L}? \quad \text{Yes!}$$

- **Automata**
  - a mathematical machine

  DEF:
  - A set of states
    - initial state
    - subset of "accept" states
  - Transition function
    $$(\text{state, symbol}) \to \text{state}.$$
    ↳ from the alphabet Σ

E.g. $\Sigma = \{a, b\}$
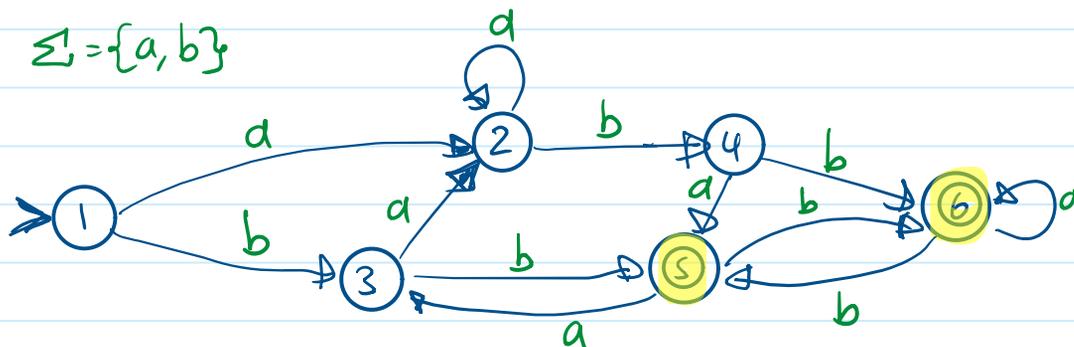
How to recognize?

Given a word w
- start in the initial state   follow the transition function

Given a word w
- start in the initial state, follow the transition function for each symbol in w e.g   abaabb
                                                                                              245356
- If the automata ends in an "accept" state then $w \in L$ otherwise $w \notin L$

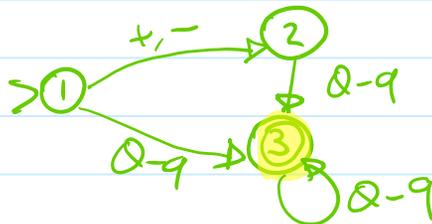$L = \{ bb, abb, aba, abbaaaa, aaaaaabb, \ldots\ldots\ldots \}$

THEOREM

For every language specified by a RegEx there exists an Automata that Recognizes it.
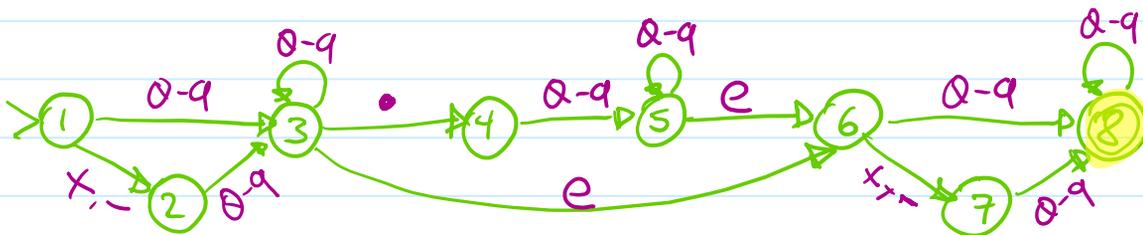
E.g. • Regex for integer Literals

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

$0 \qquad 000$
$-17 \qquad +121$

E.g.

$(+|-)?[0-9]+(.[0-9]+)?e(+|-)?[0-9]+$
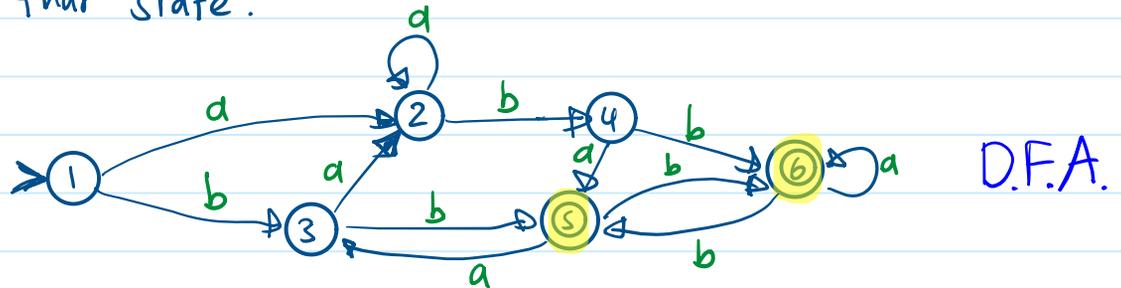

$27.3e-7 \qquad -27.52e+86$

• Coding an Automata

  • Loop over every character in w
  • a every iteration use a switch statement to

- Loop over every character in w
- a every iteration use a **switch** statement to decide the next state
    - each **case** correspond to a state in the automata
    - inside each case, consider all transitions from that state.

E.G.

D.F.A.

Pseudocode:

```
FUNCTION recognize ( string w ) : BOOLEAN
VAR state, i, c, acc
BEGIN
  state := 1;   i:=0;   acc:=False;

  WHILE i < len(w) DO
    c := w[i]
    SWITCH state OF
      CASE 1 : IF c = a THEN state := 2
               IF c = b THEN state := 3
      CASE 2 : IF c = a THEN state := 2
               IF c = b THEN state := 4
      CASE 3 : IF c = a THEN state := 2
               IF c = b THEN state := 5
      CASE 4 : IF c = a THEN state := 5
               IF c = b THEN state := 6
      CASE 5 : IF c = a THEN state := 3
               IF c = b THEN state := 6
      CASE 6 : IF c = a THEN state := 6
               IF c = b THEN state := 5
    END
    i := i + 1
  END
  IF state = 5 OR state = 6 THEN
    acc := True
  RETURN acc
END.
```
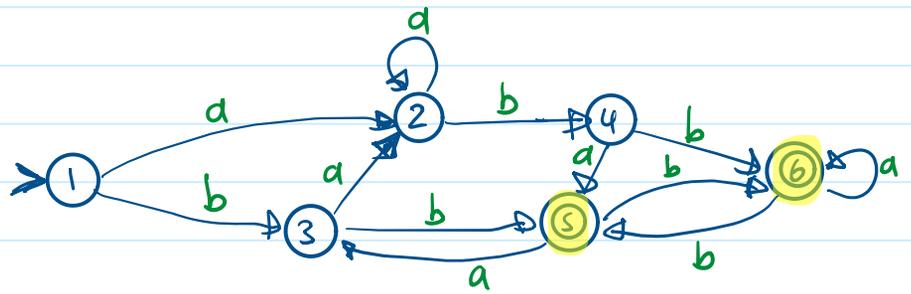
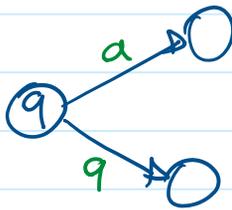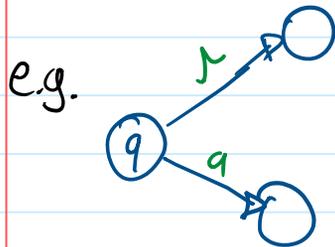w: abbaaaa

state
1 2 4 6 6 6 6 6

We could also:
- simplify the swich statement
- add **fail** and **sink** states
    - return early
- turn the swich statement into a **lookup table**
    - a b

– turn the switch statement into a lookup table

	a	b
1	2	3
2	2	4
3	2	5
4	5	6
5	3	6
6	6	5

– many other simplifications

e.g.

← non-deterministic Automata.
finite.

• This is HW #1