1-3

A computer "solves" a problem

A problem is a language

The addition problem is a set strings over $\{0-9, =, +\}^*$

$= = + + 3 \qquad 3+3=6 \qquad 3+3=18$

"is just the ones that are true"

The even problem is the set of even-length strings over Σ

The C-factorial is the set of all C-programs that complete the factorial of 25.

The job of the compute is "recognition"

given a string x

return Yes if is in the set

No if is NOT in the set

~~A computer is a function from string $\Rightarrow$ {Yes, No}~~

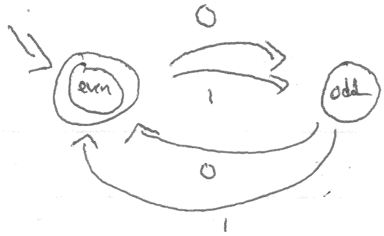
└ not actually

A computer is a finite tuple , C

A computer semantics is a finite function , $x \Rightarrow \{Y, N\}$

1-1/ Deterministic Finite Automata

Even-ness over binary :



$= \{ s \mid s \in \{0,1\}^* \text{ and } |s| = 2n \text{ for some natural} \}$

$\bigcirc$ is a node or a state

$\Rightarrow\bigcirc$ is the start state

$_a\bigcirc \xrightarrow{c} \bigcirc_b$ is a transition that goes FROM a TO b ON c

$\circledcirc$ is the accept state

$2n \rightarrow 2n+1 \rightarrow 2n+2 = 2(n+1)$

even odd even

$_a\bigcirc \xrightarrow{c,d} \bigcirc_b$ (on c or d) $_a\bigcirc \longrightarrow \bigcirc_b$ (on anything)

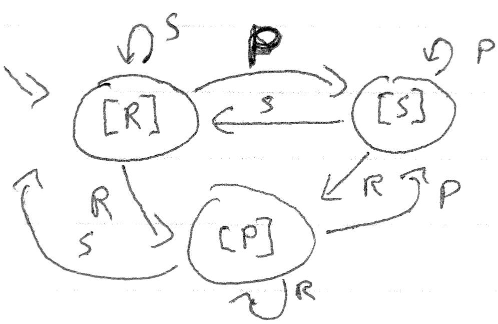
Machine with output

$c \in \Sigma'$

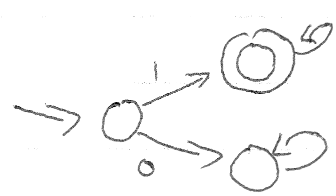
$\bigcirc$[c] when the machine reaches this state, outputs c

even[☺] Moore machine

Mealy machine $\xrightarrow{a[c]}$ outputs on transition



~~odd number~~ machine
negative float

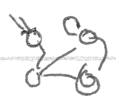


2-2

A DFA is a 5-tuple of
$(\Sigma, Q, q_0, \delta, F)$

Σ is some alphabet (finite)
Q is a finite set called states
$q_0 \in Q$ the start state
$F \subseteq Q$ the accepting states
$\delta : Q \times \Sigma \Rightarrow Q$
$\quad x \quad a \quad y$ $\qquad x \xrightarrow{a} y$

How many binary DFAs w/ 4 states?
$4 \times 2^4 \times 4 \times (4 \times 2) = 2048$



$L(\text{DFA } d)$ = the language of the DFA d
$= \{ x \mid x \text{ is accepted by DFA } d \}$
$x \in \Sigma^*$

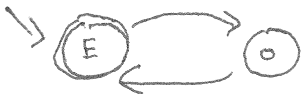
A string x is accepted by DFA d iff
$q_0 \overset{x}{\Rightarrow}{}^* q_i$ s.t. $q_i \in F$

A DFA d runs from q_i to q_j on x $(q_i \overset{x}{\Rightarrow}{}^* q_j)$
$q_i \overset{\varepsilon}{\Rightarrow}{}^* q_i \qquad q_i \overset{ax}{\Rightarrow}{}^* q_k$ iff $q_i \overset{a}{\Rightarrow} q_j$ where $q_j \in Q$
and $q_j \overset{x}{\Rightarrow}{}^* q_k \qquad a \in \Sigma$

A DFA d steps from q_i to q_j on x $(q_i \overset{x}{\Rightarrow} q_j)$
$((q_i, x), q_j) \in \delta$

$[q_i]\, x \Rightarrow [q_j]\, y \Rightarrow \ldots \Rightarrow [q_n]$

1-3/



DFA = ($\Sigma = \{0,1\}$,

$Q = \{E, O\}$,

$q_0 = E$

$\delta =$

	E	O
0	O	E
1	O	E

$F = \{E\}$)

$0110 \in L(d)$?

$E \overset{0110}{\Rightarrow^*} E$?

$E \overset{0}{\Rightarrow} O$ ✓

$O \overset{110}{\Rightarrow^*} E$?

$O \overset{1}{\Rightarrow} E$ ✓ $(O, 1, E) \in \delta$

$E \overset{10}{\Rightarrow^*} E$

$E \overset{1}{\Rightarrow} O$

$O \overset{0}{\Rightarrow^*} E$

$O \overset{0}{\Rightarrow} E$ ✓

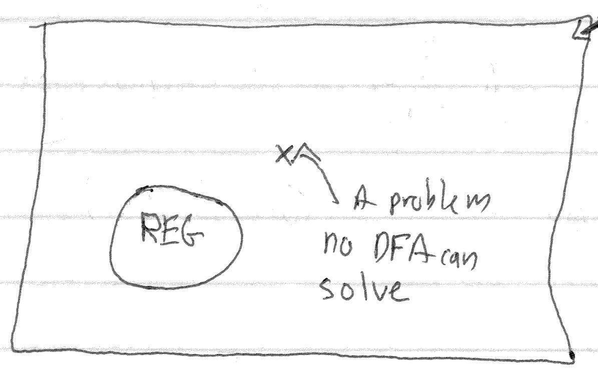
$E \overset{\varepsilon}{\Rightarrow^*} E$ ✓

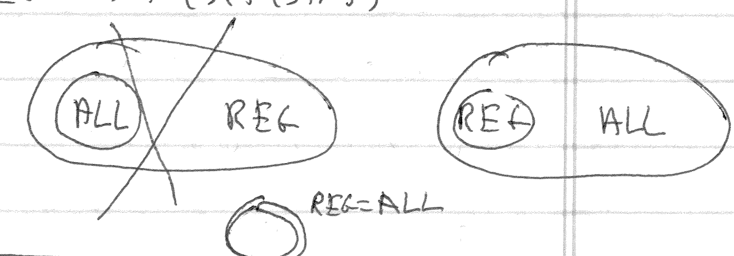
Why are Q + Σ finite?
- descriptive of reality
- rigorous foundation

Which languages are described by a DFA? Regular REG

ALL = all languages of Σ
: set(set(str))

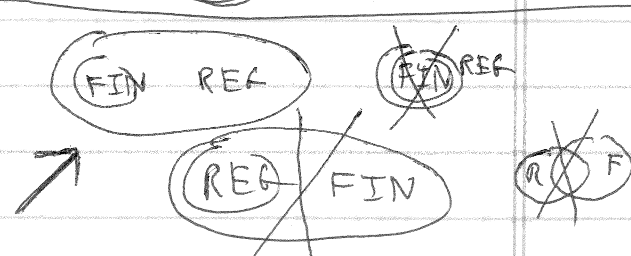
REG : set(set(str))





FIN : all languages that are finite

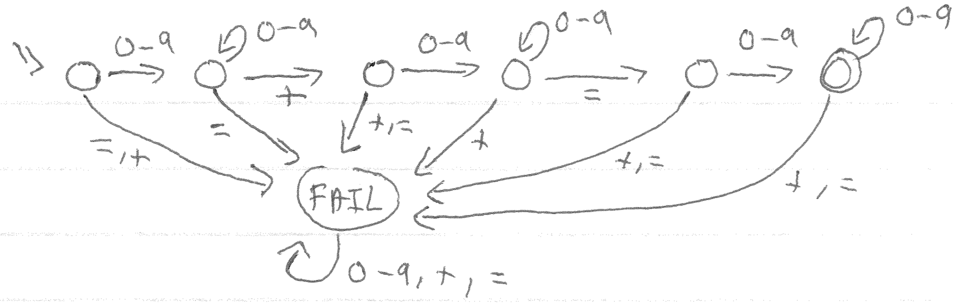
{Jay} ∈ FIN   Jay ∉ FIN



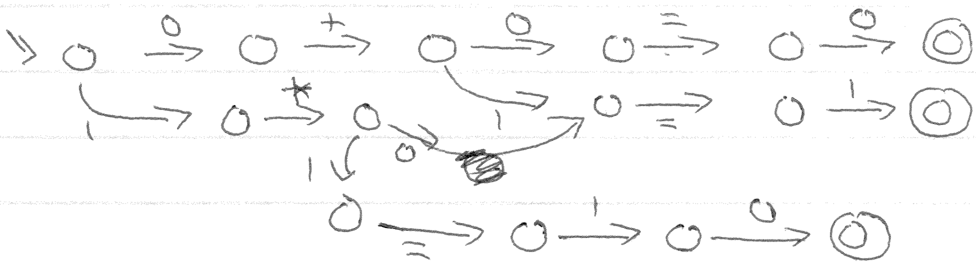
2-4/

Well-formed addition equations:

$\Sigma = 0\text{-}9, +, =$

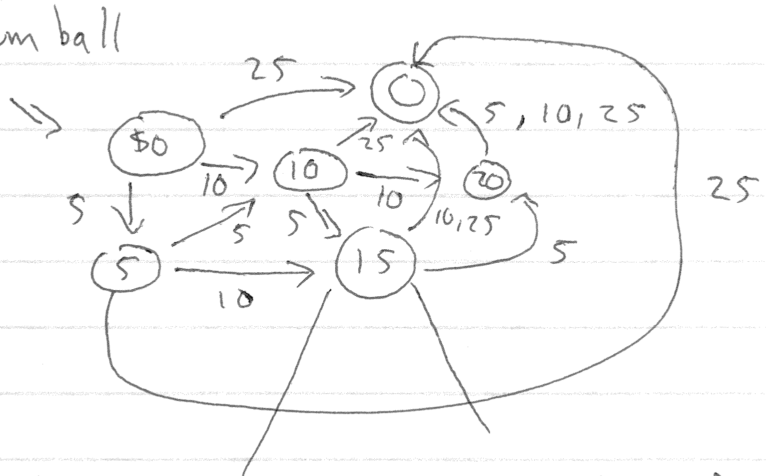


Correct, 1-bit additions



$\{0+0=0,\ 0+1=1,\ 1+0=1,\ 1+1=10\}$

Gum ball



(001) 5 ⟼ 20 (0 1)
(010) 10 ⟼ 25 (1 0)
(100) 25 ⟼ 25 (1 0)