1-1

How do we know if a math formula is true?

How do we know if an algorithm (like Euclid's GCD) works?

correct effective

Does an algorithm exist?

What is an algorithm?

Does a program exist? ⟵ problems

What is a program? ⟵ models

1-2] A set is "a bunch of stuff"

$\emptyset$ – nothin' in it

$\forall x, x \notin \emptyset$

$\{$pen, phone$\}$ $\{$phone, pen$\}$

$\{$  ,  $\}$

pen $\in$ $\{$pen, phone$\}$

$\forall x, \quad x \in \{y\}$ iff $x = y$

union – $\cup$ u u

$\forall x. \ x \in A \cup B$ iff $x \in A$ or $x \in B$

$\{$pen, phone$\}$ = $\{$pen$\} \cup \{$phone$\}$

1-3) "The set of all true math formulas"

A set IS its membership

"$1+1=2$" $\in$ TS $\uparrow$?

"Is there a god?"

"Will Buffy be remade?"

All sets "constructed" via $\emptyset$, $\{x\}$, $\cup$ are finite.

$x \in \{$  $\} \cup \{$  $\}$

The <u>Universe</u> ($\mathcal{U}$)

$A \subseteq B$ (subset) iff $\forall x,\ x \in A \Rightarrow x \in B$

1-4) Our universe is made of strings
and strings are sequences of characters
and chars are elements of an alphabet
an alphabet is a finite set

$\Sigma = \{0, 1\}$ $\{0, 1, 😃, \$, +\}$

chars chars

"0100001" = a string = s

length = 7 $s(0) = 0$ $s(1) = 1$ $s(2) = 0$

$U = \Sigma^*$ ← special notation

$A^* = \{\varepsilon\} \cup A \circ A^*$

epsilon = "" = the string w/ no characters

$x \in A \circ B$ iff $x(0) \in A$ and $x(1..) \in B$

$\{0,1\} \circ \{0,1\} = \{00, 01, 10, 11\}$

$\{1\} \circ \{0\} = \{10\}$

L5/ #1. Decide a data type to represent alphabets and characters.

```
Alphabet  = List<Character>
Character = Object    / void*
```

we need equality

#2. Decide a data type for Strings

```
interface String { }
class MtString implements String { .. }
class OneString impl String {
    OneString ( Char c, String s ) { ... } }
Zero = new BasicChar('0');   One = new BC('1');
010 = new OneS( Zero, new OneS(One, new
        OneS( Zero, new MtS())));
```

{Σ}

1-6) Every alphabet has a lexicographical ordering of the strings in Σ^*

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

" $= \mathcal{B}$

newList<Chars>

{ Zero, One};

0	1	2	3
ε	0,	00,	000
	1,	01,	001
		10,	010
		11,	011,
			100
			101
			110
			111
2^0	2^1	2^2	2^3

$|A|^i$ where i = layer I'm looking at

$\text{lexi} : \Sigma \times \mathbb{N} \to \Sigma^*$

$\text{lexi}\ \mathcal{B}\ 0 = \varepsilon$

$\text{lexi}\ \mathcal{B}\ 1 = 0$

$\text{lexi}\ \mathcal{B}\ 2 = 1$

$\text{lexi}\ \mathcal{B}\ 6 = 1\cancel{0}1$

2-1 "1+1" $\longrightarrow$ "2"

"1+1 = 2" $\in$ Truth

"1+1 = 3" $\notin$ Truth

$\emptyset$ $\{x\}$ $A \cup B$

Alphabet Σ Universe Σ^*

$\{0, 1\}$ $\{\, \varepsilon,\ 0110,\ 000001,$  $\}$

$\mathbb{P}(A)$ 2^A

$x \in P(A)$ iff $x \subseteq A$ ($x \subseteq A$ iff $\forall y \in x.\ y \in A$)

$A = \{0, 1, 2, 3\}$

$\emptyset \in P(A)$ $\emptyset \subseteq A$ 0110 $\{1, 2\}$

$\{0\} \in$ ↗ $\{2, 3\} \in$ $\{0, 2, 3\}$ ↓ $\underline{0}12\underline{3}$

$P(\Sigma^*)$ $\Sigma^* = \{\ \varepsilon, 0, 1, 00, 111111, \ldots \}$

$\emptyset \in P(\Sigma^*)$

$\{\varepsilon\} \in P(\Sigma^*)$ 0011, ...

all even lengthed strings $\in P(\Sigma^*)$ $= \{\varepsilon, 00, 11, 01, \ldots\}$

GIFs $\in P(\Sigma^*)$ {GIFs of me} $\in P(\Sigma^*)$

JPGs w/ a cat in them $\in P(\Sigma^*)$

2-2/ $ALL = P(\Sigma^*)$ FIN = the set of finite sets

$\boxed{\text{ALL}}$
- True math
- GIFs
- Even strings

$\boxed{\text{FIN}}$

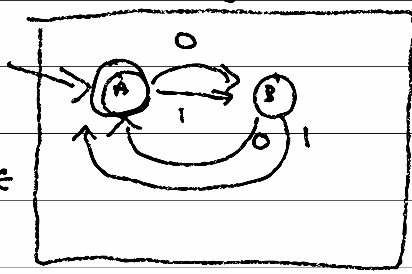
$\emptyset \in FIN$

$\forall x \in \Sigma^* . \{x\} \in FIN$

$A \in FIN \land B \in FIN$

$\rightarrow A \cup B \in FIN$

All even strings =

$x \in$ 

ε 01 ~~11~~

DFA - a deterministic finite automata

○ - states $\{A, B\}$

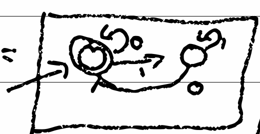
→□ - start state A

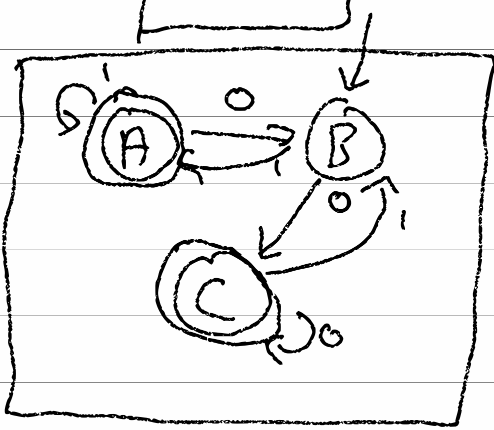
◎ - accepting states $\{A\}$

$o \xrightarrow{x} o$ - transition

	0	1
A	B	B
B	A	A

x - labels are Σ

even nums = 



$\{A, B, C\}$ B

$\{A, C\}$ $\Sigma = \{0, 1\}$

	0	1
A	B	A
B	C	A
C	C	B

2-3) $x \in DFA\ (Q,\ \Sigma,\ q_0 \in Q,\ F \subseteq Q,\ \delta : Q \times \Sigma \to Q)$

states, alphabet, start, accepting, transitions

DFA configuration $= Q \times \Sigma^*$

$[q]\ w$

config update function : config $\times$ DFA $\to$ config

$[q]w \to [q']w'$

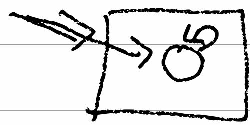
$[q_i]\ xy \to [q_j]\ y$ iff $\delta(q_i, x) = q_j$

$x \in$ DFA iff $[q_0]x \to \to \to \to \to [q_f]\varepsilon$ and $q_f \in F$

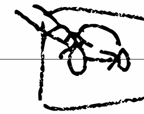
0110 ∈ EvenLen iff $[A]0110 \to [B]110 \to [A]10 \to [B]0 \to [A]\varepsilon$ $A \in \{A\}$ ✓

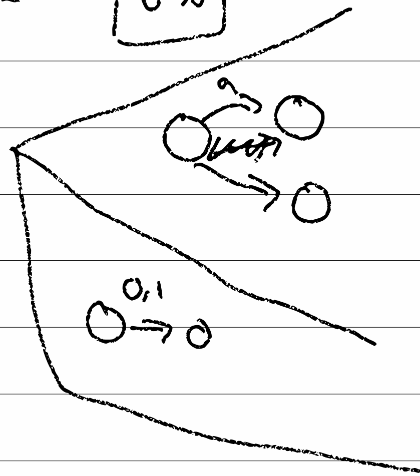
```
class DFA {
  ... Q, Σ, F, q0, δ ...
  public bool accepts (String x) {
    State qi = q0;
    while (!x.~~~~ empty()) {
      qi = δ(qi, x.first());
      x = x.rest(); }
    return F.in(qi); } }
```

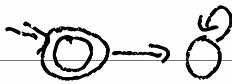
2-4



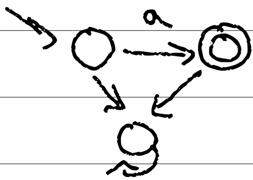
$= \emptyset =$







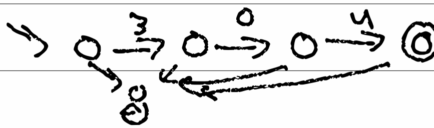
$= \{\varepsilon\}$



$= \{a\}$

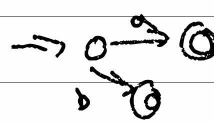
$=$

$\{304\} =$

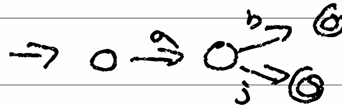


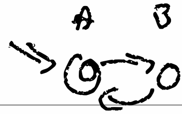
$A \cup B \in$ DFA (if $A \in$ DFA and $B \in$ DFA)

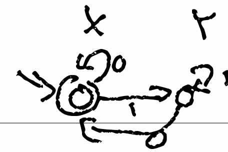
$\{a\} \cup \{b\}$



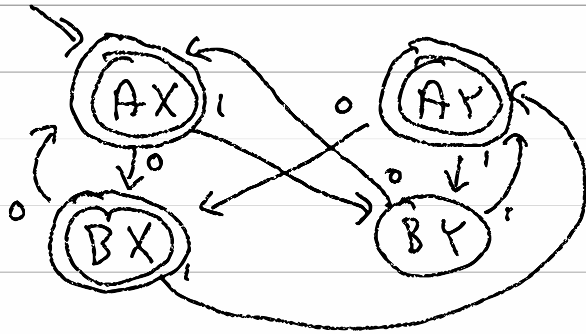
$\{ai\} \cup \{ab\}$



2-5] EvenLen 

IsEven 

$\cup$



ε ✓

00 ✓

11 ✓

0 ✓

110 ✓

$\cap$

$(x,y) \in A \times B$

iff $x \in A \wedge y \in B$

$A = (Q_A, \Sigma, q_{0A}, \delta_A, F_A)$

$B = (Q_B, \Sigma, q_{0B}, \delta_B, F_B)$

$X = A \cup B$

$Q_X = Q_A \times Q_B$

$q_{0X} = (q_{0a}, q_{0b})$

$F_X = F_A \times F_B$ — $\cap$

$F_A \times Q_B \cup Q_A \times F_B$ — $\cup$

$\delta_X((q_a, q_b), c) = (\delta_a(q_a, c), \delta_b(q_b, c))$

$x \in A \cap B$ iff $x \in A \wedge x \in B$

2-6/ $x \in A^c$ iff $x \notin A$ $(x \in U)$

Even Len   odd len

 $\Longrightarrow$ 

$F = \{A\}$ $F' = Q - F$

or F^c (wrt Q)

complement

Algorithm for $X \subseteq Y$ if X & Y are DFAs

3-1// DFA ⇒ example or false

DFA:

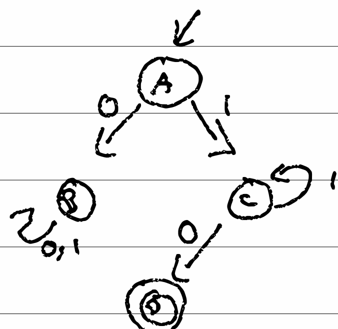
Q: State → Bool

Σ: list of characters

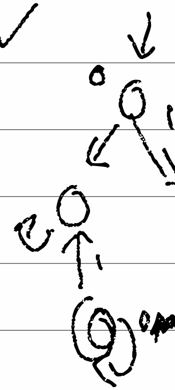
q_0: state

δ: (State × Char) → State

F: State → Bool



10 ✓



Q	∅	q_0
{A, B, C, D}	{}	[A]
{B, C, D}	{A}	[B, C]
{C, D}	{A, B}	[C]
{D}	{A, D, C}	[D]
{}	{A, B, C, D}	[ ]

A ⇒ ε

B ⇒ A, 0

C ⇒ A, 1

Yes, it's possible.'

D ⇒ C, 0

or if not, No!

10

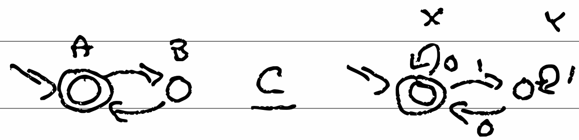
3-2/ subset

$\exists x \in U.\ x \in A \wedge x \notin B$

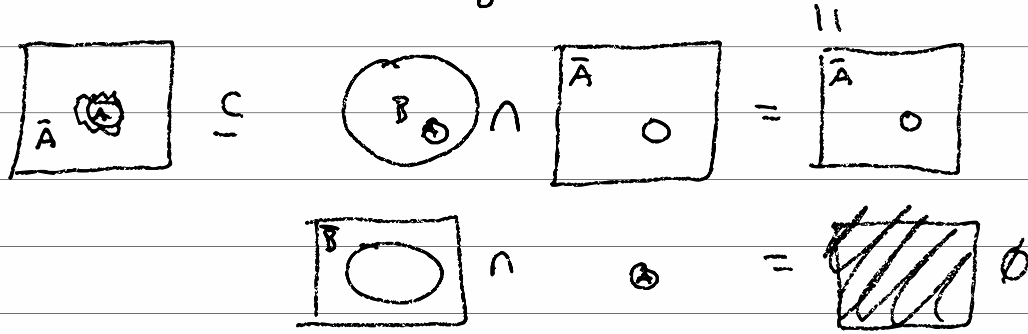
$A \subseteq B$ iff $\neg$ $\forall x \in U.\ x \in A \Rightarrow x \in B$

$\{a, b\} \subseteq \{a, b, c\}$ $\quad U = \{a, b, c\}$

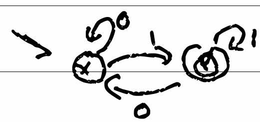
→ finite means naive works!

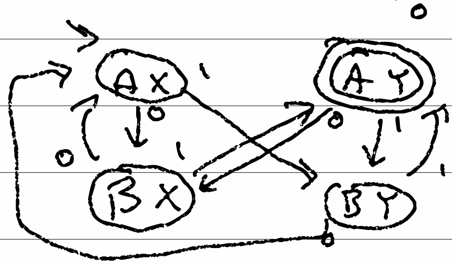


100



$\overline{\text{EvenNum}}$ =





→ 01

~~x~~ ∅

model checking

Soundness : model ⊆ theory

completeness : theory ⊆ model

model = theory

3-3/ 0, 1, 2, -1, 5   Z, P, N

$\{P\} + \{N\} = \{P, Z, N\}$

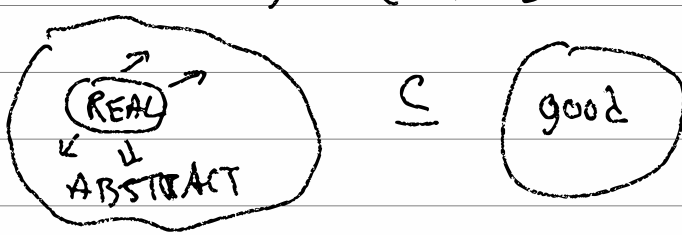
if $x > 0$ then

A $y = 5 \Rightarrow \{P\}$

o.w

B $y = 0 \Rightarrow \{Z\}$

$\Rightarrow$ assume $y = \{P, Z\}$



3-4) Finite = $\emptyset \quad \{x\} \quad A \cup B \quad A^c \quad A \cap B$ $\in$ DFA $\quad A \circ B$

Infinite = A^*

$x \in \Sigma^* \wedge y \in \Sigma^*$ then $x \circ y \in A \circ B$ iff $x \in A \wedge y \in B$

$\varepsilon \circ y = y \qquad$ if $a \in \Sigma$, $(a \circ x) \circ y = a \circ (x \circ y)$

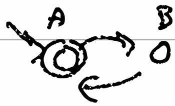
$abcd = a \circ b \circ c \circ d$

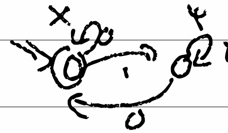
$\{j\} \circ \{m, n\} = \{jm, jn\}$

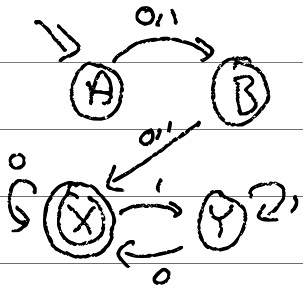
$x \in A^*$ iff $x = x_0 \circ x_1 \circ \ldots \circ x_n$ for $n \in \mathbb{N}$ and $x_i \in A$

$\{jm\}^* \ni \varepsilon,\ jm,\ jmjmjmjm$

3-5/ Even Len $\circ$ Even Num



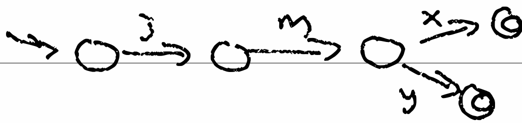


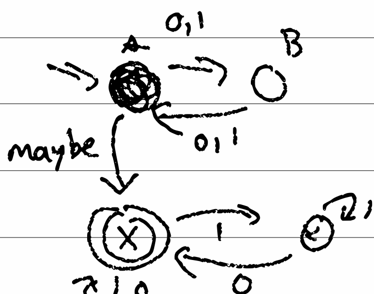


$\underbrace{00}\underbrace{110}$ ✓

0011 ×

$\{jm\} \circ \{x, y\}$

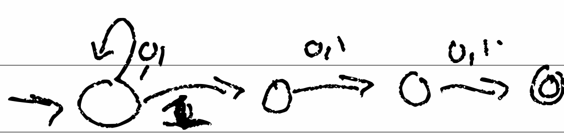




$\Sigma = \{0, 1\}$

$x \in$ DFA iff

there is some path from q_0 to $q_f \in F$ labelled w/ x



"3rd from end is 1"

3-6/ NFA = non-deterministic

finite automata

old world: the next step was obvious

$\delta: Q \times \Sigma \to Q$

new world: crazy options

- do you even read a char?
- which path do you take?

$\delta': Q \times \{\text{maybe}\} \cup \Sigma \to P(Q)$

$\delta'(A, 1) = \{A, B\}$

$\delta'(A, \text{maybe}) = \{C\}$

epsilon

$\varepsilon \quad \Sigma \quad e \quad \varepsilon$