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$\}$

$\forall x,\ x \in \{y\}$ 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> ($\mathcal{U}$)

subset

$A \subseteq B$ 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

$\overset{\downarrow 0}{0}\overset{\downarrow 1}{1}\overset{\downarrow 2}{0}$

$\underbrace{\text{"0100001"}}_{\text{length} = 7}$ = a string = s

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

$U = \Sigma^*$ ← 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^*

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

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

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

lexi B 0 = ε

lexi B 1 = 0

lexi B 2 = 1

lexi B 6 = 1~~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 $\{1, 2\}$

$\{0\} \in$ " $\{2, 3\} \in$ $\{0, 2, 3\}$ $\underline{0}12\underline{3}$

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

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

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

all even lengthed strings $\in P(\Sigma^*)$ $= \{\varepsilon, 00, 11, 01, 0011, \dots\}$

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

ALL - True math
- GIFs
- Even strings

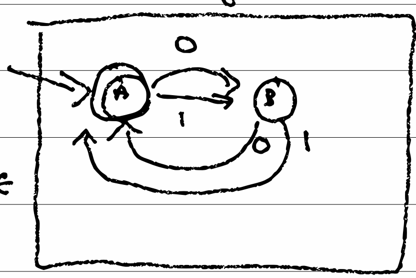
FIN

$\emptyset \in FIN$

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

$A \in FIN \wedge B \in FIN \rightarrow A \cup B \in FIN$

All even strings =



$x \in$

ε 01 ~~11~~

DFA - a deterministic finite automata

○ - states $\{A, B\}$

→□ - start state A

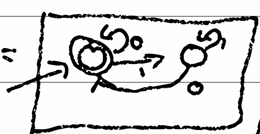
◎ - accepting states $\{A\}$

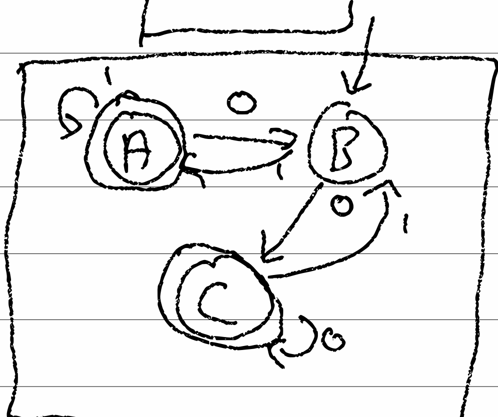
$○ \xrightarrow{x} ○$ - transition

x - labels are Σ

	0	1
A	B	B
B	A	A

even nums =





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

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

	0	1
A	B	A
B	C	A
C	C	B

2-3) $x \in DFA\,(\overset{\text{states}}{Q}, \overset{\text{alphabet}}{\Sigma}, \overset{\text{start}}{q_0 \in Q}, \overset{\text{accepting}}{F \subseteq Q},$

$\delta : Q \times \Sigma \Rightarrow Q$ — transitions)

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

$[q]\,w$

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

$[q]w \longrightarrow [q']w'$

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

$x \in$ DFA iff $[q_0]\,x \Rightarrow\Rightarrow\Rightarrow\Rightarrow\Rightarrow [q_f]\,\varepsilon$

and $q_f \in F$

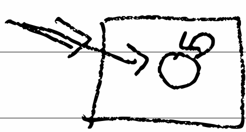
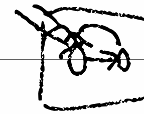
0110 $\in$ EvenLen iff $[A]0110 \Rightarrow [B]110 \Rightarrow [A]10$

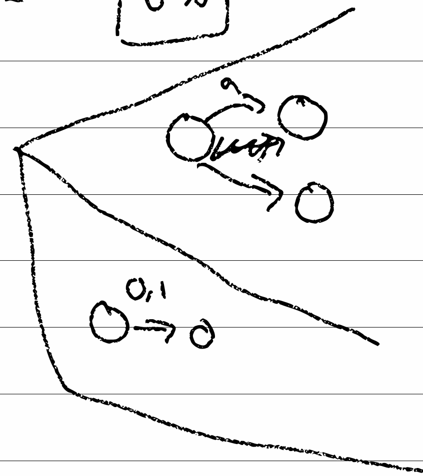
$\Rightarrow [B]0 \longrightarrow [A]\,\varepsilon$ $A \in \{A\}$

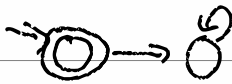
✓

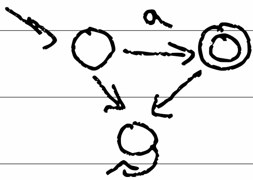
```
class DFA {
   ... Q, Σ, F, q0, δ ...
   public bool accepts (String x) {
      State qi = q0;
      while (!x.empty()) {
         qi = δ(qi, x.first());
         x = x.rest(); }
      return F.in(qi); } }
```

2-4

 $= \emptyset =$ 

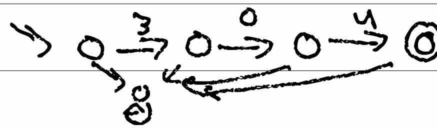


 $= \{\varepsilon\}$

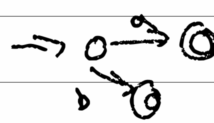


$= \{a\}$

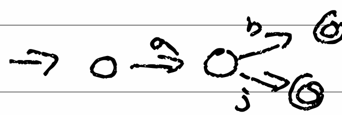
$=$ 

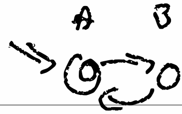
$\{304\} =$ 

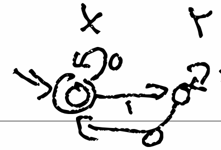
$A \cup B \in DFA$ (if $A \in DFA$ and $B \in DFA$)

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

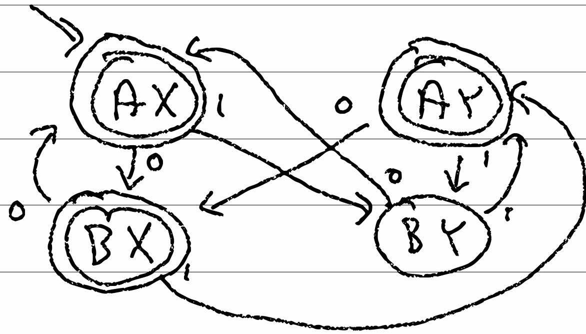
$\{a;\} \cup \{ab\}$



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

Even Len

$F = \{A\}$

odd len

$\Rightarrow$

$F' = Q - F$

or F^c (wrt Q)

complement

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