SETS
Natural nos: \( N = \{1, 2, 3, \ldots\} \)
Integers: \( \mathbb{Z} = \ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\)  
Empty set: \( \emptyset \)

Rules to define sets:
- Even = \( \{n \mid n \text{ even}\} = \{n \mid n = 2m, \text{ where } m \in N\} \)
- Rational = \( \{n \mid n \text{ rational}\} = \{n \mid n = \frac{m}{k}, \text{ where } m, k \in \mathbb{Z}\} \)

\( \text{Integers: } \sqrt{2}, \sqrt{3}, \pi, \) etc

UNION \( A \cup B \)

INTERSECTION \( A \cap B \)

COMPLEMENT \( A = U - A \)

depends on the universe \( U \)
of items you're choosing your sets from

POWERSet of \( A \)
\( A = \{10, 13\} \)
\( 2^A = \{\emptyset, \{10\}, \{13\}, \{10, 13\}\} \)
SEQUENCE = ordered list
= (2, 4, 6, 8)
FINITE SEQUENCE = "tuple"

k-tuple:
2-tuple is a pair

Cartesian product
A \times B = \{ (a, b) \mid a \in A, b \in B \}

\text{eg. } A = \{1, 2, 3\}, B = \{r, s\}
A \times B = \{ (1, r), (1, s), (2, r), (2, s), (3, r), (3, s) \}

Not limited to 2 sets:
A \times B \times C \times D \times E

A^n
\text{eg. } N^2 = \{ (n_1, n_2) \mid n_1, n_2 \in N \}

FUNCTION: maps elements of one set (domain) to elements of another set (range)

a \rightarrow b
f(a) = b

\text{eg. } a \rightarrow x (x) = x \quad x \geq 0
\quad -x \quad x < 0

add (x, y) = x + y
add: \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}
Function: All elements of domain must be mapped
(like a computer program - Must run on all inputs)

- One-to-one: every $x \in D$ mapped to one and only one $r \in R$
  $f(x) \neq f(y)$ for all $x, y \in D$

- Onto: every element in range must be mapped to

- $f(x) = -f(y)$

Bijection

- With one-to-one and onto, correspondence

Injection:

Set $A$ is "isomorphic" to set $B"
Predicate or Property: A function whose range is \{True, False\}

Function: True or False

e.g.

Even (4) = true

Even (3) = false

Relation = Property \cup \text{ Domain} = \text{ k-tuples}

\[
\begin{array}{c}
\{1, 2, 3, 4\} \\
\{0, 1, 2, 3, 4\}
\end{array}
\]

\[
\langle 1, 2 \rangle, \langle 1, 4 \rangle, \langle 2, 4 \rangle, \langle 0, 1 \rangle, \langle 0, 3 \rangle
\]

aRb = True \quad \forall (a, b) \text{ where } aRb = True

\[
\text{Domain: The set of tuples for which the relation is True}
\]

3/31/2010

**ALPHABET** = a finite set of symbols

**EXAMPLES**

\[\Sigma_2 = \{0, 1\}\] binary: the alphabet used by computers

ASCII

\[\Sigma_2 = \{a, b, c, \ldots, z\}\] English

String is defined over some alphabet

**STRING OVER** \[\Sigma = \text{ finite seq of symbols or the empty string}

\[w = w_1 \ldots w_n \quad w_i \in \Sigma\]

or

\[\epsilon = \text{ (empty string)}\]

Length of string \[w = |w| = \# \text{ of symbols}\]

**SUBSTRING** \[\Sigma \text{ of } w \text{ -- some contiguous seq of symbols from } w \text{ or } \epsilon \]

\[
\text{form: } w = x \Sigma y \quad x, y \in \Sigma
\]

FORMAL
CONCATENATION of string $x, y$

$x \cdot y$

$x = x_1 x_2 \ldots x_n$
$y = y_1 y_2 \ldots y_n$

$x \cdot y = x, x_2, \ldots x_n, y_1, y_2, \ldots y_n$

Not limited to 2 strings

$x^k = \underbrace{x \cdot x \cdot x \ldots x}_k$  

Concatenate $x$ with itself $k$ times

$x^0 = \varepsilon$

$\varepsilon^k = \varepsilon$

$\varepsilon^0 = \varepsilon$

$\varepsilon^1 = 1$

Example:

$(10)^3 = 101010 = 102$

String $w = w_1 w_2 \ldots w_n$

$w^r = \text{reversal} = w_n w_{n-1} \ldots w_1$

Example:

$(110)^2 = 011$

$\varepsilon^n = \varepsilon$

$1^n = 1$

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

Set of all strings over $\Sigma$

$\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, \ldots\}$ infinite

$\Sigma^2 = \{00, 01, 10, 11\}$

$\Sigma^0 = \{\varepsilon\}$

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

$\Sigma^2 \cup \Sigma^1 \cup \varepsilon$

$\Sigma^*$ of all strings of length up to $N$
Lexicographical order of strings over $\Sigma$

Order based on length + dictionary first, second

$\varepsilon = 0, 1, 00, 01, 10, 11, 000, 001, \ldots$

We arbitrarily order the symbols of our alphabet in the one 0 before 1

Conversion $\varepsilon$ vs $\emptyset$

$\varepsilon$ empty string, string of length 0
$\emptyset$ set of no elements

Set $A = \varepsilon, 0, 1$
Set $B = \varepsilon, 1$ not the same

Language over $\Sigma$

$\Sigma = 0, 1, 13$

$L_1 = \varepsilon 0^n 1^n \mid n \geq 0$

$L_1 = \varepsilon, 01, 0011, 000111, \ldots$

$L_2 = \varepsilon (01)^n \mid n \geq 0$

$L_2 = \varepsilon, 01, 0101, 01101$

$L_3 = \varepsilon w \mid w$ contains same # of 0's and 1's

A superset of both $L_1$ and $L_2$ (and includes other strings not in either)
Can this exercise be done?

Hapbee

Problem 1:

L = 3

Does not look like an English word.

---

The existence of a language is to create the language.

My problem can be reduced to a logarithm. Know any laws?

Any grammar is just a string of ASCII characters.

This set.

Is computable is a string leading to the end of the program. In

your case you define what "subtracting a word" means.

L = 3

P is a predefined word, thus program 2.

L = 0

---

L = 6

L is the name of a boy. A nice name.

CSE 322 = 5

L is the name of a student in 322.

EASY

2 = 3 ASCII 2

---

L = 3

P is prime? Need to get "prime and number that numerals

L = 3

M = 1

L = 322

Easy to write a program that numerals
\textbf{Proof Techniques}

1. Proof by counter-example
   - Used all the time in daily life
   - Example:
     \begin{align*}
     \text{PRIMES} &= \{ p \mid p \text{ is prime} \} \\
     \text{ODD} &= \{ 2n + 1 \mid n \text{ is odd} \}
     \end{align*}

   \textbf{Statement:} PRIMES \in ODD
   - \textbf{False:}
     \textbf{Counterexample:} \( 0 \in \text{PRIMES} \), \( 0 \not\in \text{ODD} \)

2. Proof by contradiction

   \textbf{Theorem:} Let \( S \) be any finite subset of \( \mathbb{Z} \)
   - Then, \( S \) is infinite

   \textbf{Set:} \( \{ n \in \mathbb{N} \cup \{0\} \} \)
   - The set of whole numbers
   - \( |A| = n \) \( \implies \) \( \mathbb{Z} \) cardinality or size of \( A \) = \# of elements

   \textbf{B is infinite if B is not finite}

4/2/2010
Suppose \( S \) is finite.

Go back to definition.

\[ |S| = n \text{ s.t. } n \in \mathbb{Z}_+ \cup \mathbb{N} \]

What else are you given?

\( S \) is finite

\[ |S| = m \text{ s.t. } m \in \mathbb{Z}_+ \cup \mathbb{N} \]

\[ S \cup S = \varnothing \]

\[ |S \cup S| = |S| + |S| - |S \cap S| \]

\[ = n + m \text{ s.t. } n, m \in \mathbb{Z}_+ \cup \mathbb{N} \]

\[ \Rightarrow S \cup S \text{ is finite} \]

\[ \Rightarrow \mathcal{Z} \text{ is finite} \]
3. Proof of Set Equality \( A = B \)

Show: \( A = B \)
\[ E \subseteq A \]

Then \( A - (B \cup C) = (A - B) \cap (A - C) \)

\[ \text{Venn's art: } \]
\[ \text{just to understand} \]
\[ \text{not for proof} \]

Let \( L = A - (B \cup C) \)

Let \( R = (A - B) \cap (A - C) \)

Need to show \( L \subseteq R \) and \( R \subseteq L \)

\[ L \subseteq R : \text{let } x \in L \implies x \in A \land x \notin B \cup C \]

\[ \implies x \in A \land x \notin B \land x \notin C \]

\[ x \in (A - B) \land x \in (A - C) \]

\[ \implies x \in (A - B) \cap (A - C) = R \implies L \subseteq R \]
4) IFF

\[ \text{STATEMENT 1 } \iff \text{STATEMENT 2} \]

\[ \Rightarrow \]

\[ \text{ONLY IF} \]

\[ \text{IF} \]

**THEM:** Let \( x \) be a real no.

Floor \( \lfloor x \rfloor = \text{closest integer} \leq x \)

Ceiling \( \lceil x \rceil = \text{closest integer} \geq x \)

\( \lfloor 1.25 \rfloor = 1 \)

\( \lceil 1.25 \rceil = 2 \)

\[ \lfloor x \rfloor = \lfloor x \rfloor \iff x \in \mathbb{Z} \]

**PF. ONLY IF**

Suppose \( \lfloor x \rfloor = \lfloor x \rfloor \)

Use definition

\( \lfloor x \rfloor = \mathbb{Z}_1 \in \mathbb{Z} \)

\( \lfloor x \rfloor = \mathbb{Z}_2 \in \mathbb{Z} \)

\[ \mathbb{Z}_2 \leq x \leq \mathbb{Z}_1 \]

\( \lfloor x \rfloor = \lfloor x \rfloor \)

\( \mathbb{Z}_2 = x = \mathbb{Z}_1 \)

\( x \in \mathbb{Z} \)

**PF. IFF**

Suppose \( x \in \mathbb{Z} \)

\( \lfloor x \rfloor = x \)

\( \lfloor x \rfloor = x \)

\( \lfloor x \rfloor = \lfloor x \rfloor \)
5) **Proof by Construction**

**Thm**: \( A_0 \in \mathbb{N} \quad \exists n_0 \in \mathbb{N} \text{ s.t. } n^2 > c n \quad \forall n \geq n_0.

\( n \) is size of input
\( f(n) \) is running time

Algorithm in linear time always better than quadratic, even if \( c \) is very large

an asymptotic statement
Proofs: For HW:
You can use any thing we proved in class.

Proof by Construction

Thm: \( \forall n \in \mathbb{N} \exists n_0 \in \mathbb{N} \text{ s.t. } n^2 > cn \forall n \geq n_0 \)

\[ \text{Pf. Given any value } c \text{ and } n_0 \in \mathbb{N} \text{ s.t. } n^2 > cn \forall n \geq n_0. \]

\[ n_0 > c \]

Let \( n_0 = c + 1 \)

For any \( n \geq n_0 \Rightarrow n = c+1, c+2, c+3, \ldots \)

\[ n^2 = (c+j)^2 \]

\[ = (c+j)(c+j) \]

\[ > (c+j)c = cn \]
Proof by Induction

EX: \[ 1 + 2 = 3 = \frac{2(2+1)}{2} \]
\[ 1 + 2 + 3 = 6 = \frac{3(3+1)}{2} \]
\[ 1 + 2 + 3 + 4 = 10 = \frac{4(4+1)}{2} \]
\[ 1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2} \]

3 STEPS
1. Basis step
2. a. Induction Hypothesis
   b. Induction Step

Basis: Show TRUE for \( k = 0 \) (or 1)

Ind Hyp: Assume true for some \( k \)

Ind Step: Show true for \( k+1 \)

\[ \text{LHS:} \quad 1 + 2 + \ldots + n = \sum_{i=1}^{n} i = \frac{n(n+1)}{2} \]

PE: Basis step: \( n = 1 \) \[ \sum_{i=1}^{1} i = 1 (\text{LHS}) \]
\[ \frac{1(1+1)}{2} = 1 (\text{RHS}) \]

Ind Hyp: Suppose true for \( k \): \[ \sum_{i=1}^{k} i = \frac{k(k+1)}{2} \]

Ind Step: Show true for \( k+1 \)
\[ \text{LHS:} \quad \sum_{i=1}^{k+1} i = \sum_{i=1}^{k} i + k + 1 = \frac{k(k+1)}{2} + k + 1 \]
\[(k+1) \left( \frac{k}{2} + 1 \right) = \frac{(k+1)(k+2)}{2} \]

No need to explicitly show that this is the LHS.

**Pigeonhole Technique**

**DEF:** 1-1 Function

\[
f : A \rightarrow B \text{ is 1-1 if } \forall x, y \in A \quad y \neq x \implies f(x) \neq f(y)
\]

**DEF:** Onto Function

\[
f : A \rightarrow B \text{ is onto if } \forall y \in B \exists x \in A \text{ s.t. } f(x) = y
\]

**DEF:** Bijection = 1-1 and Onto

**Pigeonhole Principle**

**EX:** 5 students \( \rightarrow \) 4 chairs

**EX:** 400 people \( \rightarrow \) guaranteed that some share a birthday

\[400 \rightarrow (366 \text{ days})\]

If cardinally \( A \leq B \), then \( \exists \not\exists 1-1 f : A \rightarrow B \)

\( i.e. ~ \exists a_1, a_2 \in A \text{ s.t. } a_1 \neq a_2 \)

\[\text{and } f(a_1) = f(a_2)\]
DOVETAILING

\[ |A| = \text{size of } A = \text{No. of elements in } A \]

Comparing A, B

\[ A = \{a, b, c\} \]
\[ B = \{1, 2, 3\} \]

f: 1-to-1, onto \Rightarrow they have same size

\[ \exists \text{ bijection } f: A \rightarrow B \]
\[ \iff |A| = |B| \]

Useful for comparing infinite sets

1. A set A is countably infinite \( \iff \exists \text{ bijection } f: N \rightarrow A \)
2. A set A is countable if it is finite or countably infinite
3. A set is uncountable if it is not countable

\[ A = N - \{1, 2\} \]

A ≤ N

\[ |N| \rightarrow |A| \]

Is \(|A| < |N|\)?

\[ \text{No} \]

\[ \{1, 2, 3, ...\} \]

Set of integers

\[ \mathbb{Z} = \ldots, -2, -1, 0, 1, 2, 3, ... \]
\[ \mathbb{N} = 1, 2, 3, ... \]

Is \( \mathbb{Z} \) bigger than \( \mathbb{N} \)?
Think of $\mathbb{Z}$ as:

\[
\begin{array}{c}
0 & -1 & 1 & 2 & -3 & 3 & -4 & 4 \\
\uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow \\
1 & 2 & 3 & 4 & 3 & 2 & 1 & 0
\end{array}
\]

DOVETAILING

Sawi Shape

4/7/2010

DOVETAILING

technique for showing a set is countably infinite

by mapping to $\mathbb{N}$

should be able to say, what is the $10^{th}$ element in this set?

\[ f(n) = \begin{cases} 
\frac{n}{2} & \text{if } n \text{ is even} \\
\frac{-(n-1)}{2} & \text{if } n \text{ is odd}
\end{cases} \]

1 \rightarrow 0

2 \rightarrow 1

3 \rightarrow -1

4 \rightarrow 2

5 \rightarrow -2

What about $\mathbb{N} \times \mathbb{N}$?

$\exists (n_1, n_2) \mid n_1, n_2 \in \mathbb{N}$

\[ S = (n_1, n_2) \]
\[ \mathbb{N} \times \mathbb{N} = \{(1,1), (1,2), (1,3), \ldots\} \]
\[ \mathbb{N} \times \mathbb{N} = \{(2,1), (2,2), (2,3), \ldots\} \]
\[ \mathbb{N} \times \mathbb{N} = \{(3,1), (3,2), (3,3), \ldots\} \]
\[ = \mathbb{N} \times \mathbb{N} \cup \mathbb{N} \times \mathbb{N} \cup \mathbb{N} \times \mathbb{N} \cup \ldots \]

A union of an infinite number of infinite sets

**LEAVING ACTUAL FUNCTION AS AN EXERCISE**

A countably infinite union of countably infinite sets is countable.

\[ \mathbb{N} \times \mathbb{N} \times \mathbb{N} \times \mathbb{N} \times \ldots \text{ is countable} \]

**Proof**

The union of 2 countably infinite sets is countable.

\[ \mathbb{Z} \times \mathbb{Z} \text{ is countable} \]

**Power Set**

\[ \text{Power} \,(N) = 2^N = \text{set of all subsets of } N \]

Includes infinite subsets.

Can't be ordered by size because:

Size = infinite

Suppose \( \text{Power} \,(n) \) is countably infinite,

\[ \Rightarrow \exists \, 1-1, \text{ onto } \varphi : N \to \text{Power} \,(N) \]
\[ f(1) \mathbb{N} \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \ldots \]

\[ D = \{ n \in \mathbb{N} \mid n \geq 3 \} \]

D differs from every set in our list by at least one element.

\[ D \subseteq \mathbb{N} \text{? yes } \Rightarrow \text{ has to be a set in the list} \]

\[ D \subseteq \mathbb{N} \Rightarrow D \in \text{Pow}(\mathbb{N}) \]

\[ \Rightarrow D = \mathbb{N}_i \text{ for some } i \]

\[ \text{Is } i \in D? \]

\[ \text{Yes: } i \in D \Rightarrow i \in \mathbb{N}_i \]

\[ \text{No: } i \notin D \Rightarrow i \notin \mathbb{N}_i \Rightarrow i \notin D \]

Method is called Diagonalization.

Real no's between (0,1)

Assume 1-1 onto f

\[ N \subseteq \mathbb{R} \]

\[ 1 \rightarrow 0.1602 \]

\[ 2 \rightarrow 0.1297 \quad \text{add } 1 \]

\[ 3 \rightarrow 0.0057 \]

\[ 4 \rightarrow 0.091729 \]

\[ r = .6328 \]
4/9/2010

1) Proposition: Hare can be proven by induction.
   (or size of range?)

2) REVIEW OF PROOF TECHNIQUES → ON SLIDES

2) FINITE AUTOMATA: CSE 322 STATES OF MIND

DEF: PROOF, JOKES

DEFINITION → PROOF

ATTENTION → BOILED → NAP

VENDING MACHINE

Gum Costs $0.25
Accepts 5, 10, 25

No refunds, overpayment is welcome

For all inputs
For all states

All inputs
Input String: "nnnnn" ACCEPT
"nn" REJECT
"n" ACCEPT

STATES = "0", "s", "10"

INPUTS = string over Alphabet \( \Sigma = \{e, n, d, q\} \)

TRANSITIONS = "ARROWS"

START STATE: "0" state

SET OF ACCEPT STATES:

\[ \Sigma = \{0, 1, 2\} \]

Automaton = Computer

Language recognized by this machine = \( \{w \mid w \text{ ends in } 0\} \)

\[L(M)\]

\[L(M_2) = \{w \mid w \text{ contains an even number of } 0\}'\]

Parallel? Not THESE (DFA)
NFA's do parallel
Process 1101 with M2

\[ w = 1101 \]
\[ q_0 = 1101 \]

1. \[ q = 101 \]
   \[ b_{\text{out}} = 1 \] \[ q_0 = 01 \]

2. \[ q = 10 \]
   \[ b_{\text{out}} = 0 \]

3. Reject \( w \)

w = 1010

1. \[ q = 1010 \]
   \[ b_{\text{out}} = 1010 \]

2. \[ q = 010 \]
   \[ b_{\text{out}} = 010 \]

3. \[ 10 \]
   \[ b_{\text{out}} = 10 \]

4. \[ 101q_0 \]

5. \[ 1010 q_{\text{out}} \rightarrow \text{Accept } w \]
**Formal Def. of a F.A.**

- Set of states $Q$
- Alphabet $\Sigma$
- Transition function $\delta : Q \times \Sigma \to Q$
- Start state $q_0 \in Q$
- Set of accept states $F \subseteq Q$ (or "final")

$5$-tuple $(Q, \Sigma, \delta, q_0, F)$

---

$4/12/2010$

If $F = \emptyset$, language will be $\emptyset$.

Won't accept anything.

Student: will lose forever

3NF vs. input is finite

---

One way to show $\delta$:

- Even / Even / Even
- Odd / Even / Odd
- Odd / Odd / Odd
Computation by Finite Automaton

FA \( M \) accepts INPUT \( w = w_1 w_2 w_3 \ldots w_n \)
iff \( \exists n+1 \) states \( q_0, q_1, \ldots, q_n \in Q \)
with
1. \( q_0 = q_0 \)
2. \( \delta(q_i, w_{i+1}) = q_{i+1} \quad \forall i \in \{0, 1, 2, \ldots, n-1\} \)
3. \( q_n \in F \)

Examples

\[ L(M) = \{ w \mid w \text{ does not contain } 00^3 \} \]

Use same for rejecting any web page containing the word "cougars"

To recognize complement \( \overline{L(M)} = \{ w \mid w \text{ contains } 00^3 \} \)

\( F = \{ q_1, q_2 \} \)

General rule: Flip the accept \( \rightarrow \) reject states

\( L(M) = \{ w \mid w \text{ has even } \# \text{ of } 0's \text{ and odd } \# \text{ of } 1's \} \)

What are possible combo's?

- even 0's, even 1's
- even 0's, odd 1's
- odd 0's, even 1's
- odd 0's, even 1's

One way to look at this problem
Figure out which combo is the start state (ge,e)

What about \( L(M_2) \)? Switch reject / accept

\[ L(M_1) = \{ w \in 0^* 1 \text{ contains even # of 0's} \} \]

\[ F = \{ ge, ege \} \]

How can we reduce to 2 states?

There is no unique machine for any language.

**Definition:** A language \( L \) is regular iff

some F.A. \( M \) recognizes it.

i.e. \( L(M) = L \)

eg. \( L(M_2) = \{ w \in 0^* 1 \text{ contains even # of 0's} \} \)

\[ L_1 = \{ w \in 0^* 1 \text{ contains even # of 0's} \} \]

\[ L_2 = \{ w \in 0^* \text{ even # in 0's} \} \]

\[ L_1 \cap L_2 \text{ regular?} \leq \text{Closure Operation} \]
\[ L_1 \cap L_2 = \{ w \in \{0,1\}^* \mid w \text{ ends in } 03 \text{ AND ends in } 03 \} \]

**Closure**

Under addition, \( R \) is closed.

Under intersection, are regular languages closed?

We already showed under complementation, regular languages are closed.

Mark states for each pair of states:

- \((f_1, f_2)\) is connected to \((g_0, g_2)\).
- \((f_1, f_2)\) is connected to \((g_0, g_1)\).
4/14/2010

\[ \text{If } \Rightarrow \{000\} \Rightarrow \text{Should not accept } 000 \]

\[ L_1 = \{ \text{all } w \text{ cont. even # of 0's} \} \]
\[ L_2 = \{ \text{all } w \text{ ends in } 0^k \} \]
\[ L = L_1 \cap L_2 \]
\[ L_1, L_2 \text{ RE ?} \]

Build a new TM that simulates \( M_1 \cup M_2 \) in parallel.

\[ M_1 \]
\[ \text{Start} \]
\[ \text{Todd} \]

\[ M_2 \]
\[ \text{Start} \]
\[ \text{Todd} \]

Draw a transition diagram for the new TM.
\( L = L_1 \cup L_2 \)

States are same - just change set of accept states

(formula)

\( L_1, L_2 \) reg \( \equiv \) \( L_1 \cup L_2 \) reg

\( F = L \left( \delta_1, q_1 \right) \cup \left( \delta_2, q_2 \right) \)

\text{THM:} \text{Reg. lang's are closed under } \cup \text{ and } \cap

i.e. \( L_1, L_2 \text{ reg } \equiv L_1 \cap L_2 \text{ reg} \)

\text{If } L_1, L_2 \text{ are regular } \Rightarrow \exists \text{ 2 machines } M_1, M_2

\text{s.t. } L(M_1) = L_1 \text{ and } L(M_2) = L_2

\begin{align*}
M_1 &= (Q_1, \Sigma_1, \delta_1, q_1, F_1) \\
M_2 &= (Q_2, \Sigma_2, \delta_2, q_2, F_2)
\end{align*}

\text{Define } M \text{ s.t.}

\[ L(M) = L_1 \cap L_2 \]

\[ M = (Q, \Sigma, \delta, q_0, F) \]

\[ Q = Q_1 \times Q_2 = \left\{ (q_1, q_2) \mid q_1 \in Q_1 \text{ and } q_2 \in Q_2 \right\} \]

\[ \Sigma = \Sigma_1 \cup \Sigma_2 \]

\[ \delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a)) \]

\[ q_0 = (q_1, q_2) \]

\[ F = \left\{ (q_1, q_2) \mid q_1 \in F_1 \text{ and } q_2 \in F_2 \right\} \]

\[ = F_1 \times F_2 \]
For Union:

\[ F = \forall (r_1, r_2) \mid r_1 \in F \lor r_2 \in F \]

For Concatenation:

\[ L_1 \cdot L_2 = \{ w_1w_2 \mid w_1 \in L_1, w_2 \in L_2 \} \]

\[ w = 0010 \]

Can it be split to be \( L_1 \cdot L_2 \):

\[ \begin{align*}
00 & \quad 10 \\
W_1 & \quad W_2
\end{align*} \quad \text{or} \quad \begin{align*}
0010 & \\
W_1 & \quad W_2
\end{align*} \]

\[ w = 0010 \]

No way to split that works.

To test \( w \), have to split \( w \) in different ways.

\[ \begin{align*}
00 & \quad 1000 \\
00 & \quad 0000 \\
00 & \quad 0000
\end{align*} \]

Could be done in parallel \( \Rightarrow \) multiple threads

If any thread succeeds, accept.
Don't accept by ALL and FINAL states are rejects.

Each branch does a different split.

\( M = (Q, \Sigma, \delta, q_0, F) \)

\( Q = \{ q_1, q_2 \} \)

\( \Sigma = \{ 0, 1 \} \)

\( \delta(q_0, 0) = q_1 \)

\( \delta(q_0, 1) = q_2 \)

\( \delta(q_1, 0) = q_1 \)

\( \delta(q_1, 1) = q_2 \)

\( \delta(q_2, 0) = q_2 \)

\( \delta(q_2, 1) = q_1 \)

\( F = \{ q_1 \} \)