322

3/29/2010

SETS

Natural nos: $N = \{1, 2, 3, \ldots\}$

Integers: $\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$

Empty set: $\emptyset$

USING RULES TO DEFINE SETS:

$\{n \mid \text{rule about } n\}$

$\text{Even} = \{n \mid n \,\%\, 2 = 0\}$

$= \{n \mid n = 2m, \text{ where } m \in N\}$

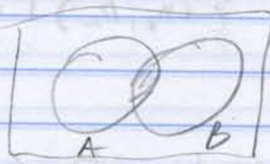
$\text{Rational} = \{n \mid \exists\, m, k \in \mathbb{Z} \text{ s.t. } n = \frac{m}{k}\}$

[Irrational: π, $\sqrt{2}$, e]

UNION $A \cup B$

VENN DIAGRAMS

INTERSECTION $A \cap B$



COMPLEMENT $\overline{A} = U - A$

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

SUBSET, PROPER SUBSET - not $\emptyset$, not itself

POWER SET OF A

$A = \{0, 1\}$

$2^A = \{\emptyset, \{0\}, \{1\}, \{0, 1\}\}$

$\text{len}\,|2^A| = 2^{\text{len}|A|}$

HE DIDN'T EXPLAIN WHY

SEQUENCE = ordered list

= (2, 4, 6, 4,)

FINITE SEQUENCE = "Tuple"

k-tuple:

2-tuple is a pair

CARTESIAN PRODUCT

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

$A = \{\emptyset, 1\}$ $B = \{x, y, z\}$

$A \times B = \{(\emptyset, x), (\emptyset, y), (\emptyset, z), (1, x), (1, y), (1, z)\}$

Not limited to 2 sets:

$A \times B \times A = \{(\emptyset, x, \emptyset), \ldots\}$

$A \times B \times C \times D \times E$

A^k

e.g. $N^2 = \{(n_1, n_2) \mid n_1, n_2 \in N\}$

FUNCTION: maps elements of 1 set (domain) to elements of another set (range)

$a \xrightarrow{f} b$ f: domain $\rightarrow$ range

$f(a) = b$

e.g. $abs(x) = x \quad x \geq 0$

$\quad -x \quad x < 0$

2-tuple

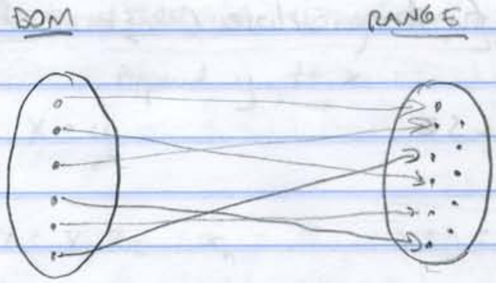
$add(x, y) = x + y$

$add: Z \times Z \rightarrow Z$

3/29/2010

DOM RANGE

ONE TO ONE:



Not all elements of range have to be mapped to

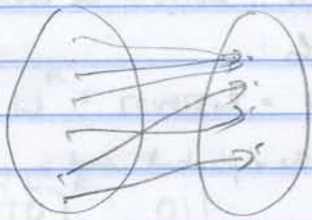
$\frac{1}{x}$ → be careful how you define domain

Function: All elements of domain must be mapped (like a computer program - Must run on all inputs)

one-to-one: every $d \in D$ mapped to one and only one $r \in R$

$f(x) \neq f(y)$ for all $x, y \in D$

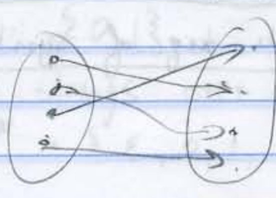
ONTO: every element in range must be mapped to.



OK for $f(x) = f(y)$

BIJECTION

BOTH ONE-TO-ONE AND ONTO [correspondence]



WITH BIJECTION,

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 w/ Dom = a set of k-tuples

[binary relation D = set of pairs]

"<"   "="

set of all tuples that satisfy the relation

Relation < = {(1,2), (1,4), (2,4), ... }

aRb = True , R(a, ... ak) = True

DOMAIN: The set of tuples for which the relation is TRUE

3/31/2010

ALPHABET = a finite set of symbols

EXAMPLES

$\Sigma_1 = \{0, 1\}$ binary: the alphabet used by computers

ASCII

$\Sigma_2 = \{a, b, c, \dots z\}$ English

STRING is defined over some alphabet

STRING OVER Σ = $\boxed{\text{finite seq. of symbols}}$ or the empty string

$w = w_1 w_2, \dots w_n \quad w_i \in \Sigma$

or

ε (empty string)

Length of String w = |w| = # of symbols

SUBSTRING Z of W = some contiguous seq of symbols from W or ε = $\boxed{\exists \text{ STRINGS } x \text{ and } y \text{ s.t. } W = xZy}$

FORMAL

CONCATENATION of strings x, y

Append y to x

$x \circ y$

$x = x_1 x_2 \dots x_n$ $\quad$ $x \circ y = x_1 x_2 \dots x_n y_1 y_2 \dots y_n$

$y = y_1 y_2 \dots y_n$

not limited to 2 strings

$x^k = \underbrace{x \circ x \circ x \dots \circ x}_{k \text{ copies}}$ $\quad$ Concatenate x w/ itself k times

$x^0 = \varepsilon$ $\qquad$ $\emptyset^k = \varepsilon$

eg $(10)^3 = 101010 = 42_d$

STRING $w = w_1 w_2 \dots w_n$

w^R = reversal $= w_n w_{n-1} \dots w_1$

eg $(110)^R = 011$

$\varepsilon^R = \varepsilon$

$1^R = 1$

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

set of all strings over Σ

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

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

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

$\bigcup_{i=0}^{N} \Sigma^i$ = set of all strings of length up to N

$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \dots$

Lexicographical ORDER OF STRINGS OVER Σ

→ order based on $\boxed{\text{length}}$ + $\boxed{\text{dictionary}}$
length ↓ First; dictionary ↓ SECOND

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

LEX ORDER

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

(WE ARBITRARILY ORDER THE SYMBOLS OF OUR ALPHABET IN THIS CASE 0 before 1)

CONFUSION ε vs $\emptyset$

ε Empty String : string of length $\emptyset$

$\emptyset$ Set of no elements

SET $A = \{\varepsilon\}$
SET $B = \{\}$
not the same

Language over Σ

= set of strings over Σ

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

$L_1 = \{0^n 1^n \mid n \geq 0\}$

eg ε, 01, 0011, 000111 ...

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

ε, 01, 0101, 010101

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

A superset of Both L_1 and L_2 (and includes other strings not in either)

L

$L_{322} = \{ w \mid |w| = 322 \}$ easy to write a program that recognizes

$L_{PRIMES} = \{ 0^p \mid p \text{ is prime} \}$ hard to " " "

$\Sigma = \{ \text{ASCII CHARS} \}$

$CSE322 = \{ w \mid w \text{ is the name of a student in } 322 \}$ EASY

$L_{BA} = \{ w \mid w \text{ is the name of a Ben Affleck movie} \}$

$L_{GOOD\text{-}BA} = \emptyset$

JUST LOOK UP RULES

$L_{JAVA} = \{ P \mid P \text{ is a syntactically correct JAVA program} \}$ EASIER

valid once you define what "syntactically correct" means
so compiler is actually checking to see if your program is in this set.

Any program is just a string of ASCII characters

** Any problem can be reduced to a language. Then ask how easy or hard is it to recognize this language?

$L_{NOLOOP} = \{ P \mid P \text{ does not go into an infinite loop} \}$ HARDER

IS GUARANTEED TO HALT

SIMULATE PROGRAM

CAN THIS EVEN BE DONE?

PROOF TECHNIQUES

1. Proof by counter-example
used all the time in daily life

eg
PRIMES = $\{0^p \mid p \text{ is prime}\}$
ODD = $\{0^n \mid n \text{ is odd}\}$

STMT: PRIMES $\subseteq$ ODD
FALSE
COUNTEREXAMPLE: $0^2 \in$ Primes
$0^2 \notin$ ODD

4/2/2010

2 Proof by Contradiction

Thm: let S be any finite subset of $\mathbb{Z}$
Then, $\bar{S}$ is infinite

A is finite if $\exists\, n \in \mathbb{N} \cup \{0\}$ the set of whole numbers
st. $|A| = n$
cardinality or size of A = # of elements

B is infinite if B is not finite

P

PF

suppose $\bar{S}$ is finite

go back to definition

$\Rightarrow |\bar{S}| = n$ s.t. $n \in \{0\} \cup N$

what else are you given?

S is finite

$\Rightarrow |S| = m$ s.t. $m \in \{0\} \cup N$

$S \cup \bar{S} = Z$

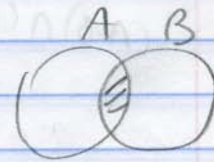
$\emptyset$

$|S \cup \bar{S}| = |S| + |\bar{S}| - |S \cap \bar{S}|$

$= n + m \in \{0\} \cup N$

$\Rightarrow S \cup \bar{S}$ is finite

$\Rightarrow Z$ is finite



Explain & add'n rule

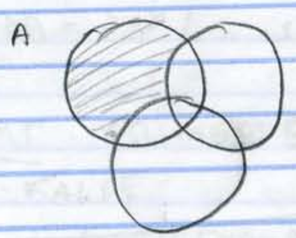
MINUS = NOT

3. Proof of Set Equality $A = B$

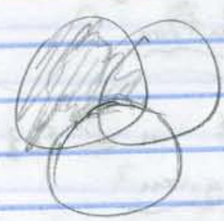
<u>SHOW</u>: $A \subseteq B$

$B \subseteq A$

<u>THM</u> $A - (B \cup C) = (A - B) \cap (A - C)$



A - B



A - C

VENN'S ARE JUST TO UNDERSTAND, NOT FOR PROOF

$(A-B) \cap (A-C)$

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

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

Need to show $L \subseteq R$ and $R \subseteq L$

<u>$L \subseteq R$</u>: let $x \in L \Rightarrow$

$x \in A \wedge x \notin B \cup C$

$\Rightarrow x \in A \wedge x \notin B \wedge x \notin C$

$x \in (A-B) \wedge x \in (A-C)$

$\Rightarrow x \in (A-B) \cap (A-C) = R$

$\Rightarrow L \subseteq R$

4/12/2010

Then show $R \subseteq L$

USE THIS TO PROVE ONE LANGUAGE IS SAME AS ANOTHER
(EQUALITY OF 2 SETS OF STRINGS)

4) IFF $\Longleftrightarrow$

STATEMENT 1 $\overset{\text{iff}}{\Longleftrightarrow}$ STATEMENT 2

ONLY IF $\Longrightarrow$

IF $\Longleftarrow$

USUALLY 1 SIDE EASY,
OTHER SIDE HARD

THM: Let x be a real no.

Floor $\lfloor x \rfloor$ = closest integer $\le x$

Ceiling $\lceil x \rceil$ = " " $\ge x$

$\lfloor 1.25 \rfloor = 1$

$\lceil 1.25 \rceil = 2$

$$\boxed{\lceil x \rceil = \lfloor x \rfloor \Leftrightarrow x \in \mathbb{Z}}$$

PF. ONLY IF

Suppose $\lceil x \rceil = \lfloor x \rfloor$

use definition

$\lceil x \rceil = z_1 \in \mathbb{Z}$

$\lfloor x \rfloor = z_2 \in \mathbb{Z}$

$\underbrace{z_2 \le x \le z_1}_{= \ \lceil x \rceil = \lfloor x \rfloor}$

$z_2 = x = z_1$

$x \in \mathbb{Z}$

PF. IF

Suppose $x \in \mathbb{Z}$

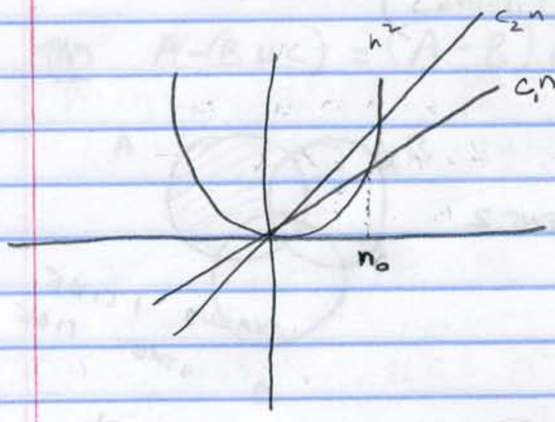
$\lceil x \rceil = x$

$\lfloor x \rfloor = x$

$\lceil x \rceil = \lfloor x \rfloor$

5) Proof By Construction

THM $\forall c \in N \ \exists n_0 \in N$ s.t. $n^2 > cn \ \forall n \geq n_0$



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

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

An asymptotic statement

4/5/2010

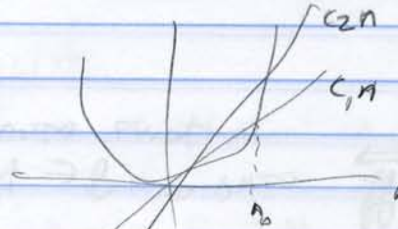
PROOFS: FOR HW:

You can use anything we proved in class.

" " " in book.

PROOF BY CONSTRUCTION

Thm: $\forall c \in N \; \exists n_0 \in N$ s.t. $n^2 > cn \quad \forall n \geq n_0$



Pf. GIVEN ANY VALUE FOR C, NEED $n_0 \in N$

s.t. $n^2 > cn \quad \forall n \geq n_0$

$n_0^2 > cn_0 \Rightarrow n_0 > c$

Let $n_0 = c+1$

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

$n = c + j \quad j = 1, 2, 3 \ldots$

$n^2 = (c+j)^2$

$= (c+j)(c+j) \quad \cancel{\ldots}$

$> (c+j)c = cn$

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 $k+1$

THM: $1+2+\ldots+n=\underbrace{\sum_{i=1}^{n} i}_{\text{LHS}}=\underbrace{\frac{n(n+1)}{2}}_{\text{RHS}}$

PF. Basis Step: $n=1$ $\sum_{i=1}^{1} i=1$ (LHS)

$\frac{1(1+1)}{2}=1$ (RHS)

IND HYP: SUPPOSE TRUE FOR k: $\sum_{i=1}^{k} i=\frac{k(k+1)}{2}$

ALWAYS TRY TO REWRITE USING k

IND STEP: SHOW TRUE FOR $k+1$

LHS $\sum_{i=1}^{k+1} i=\sum_{i=1}^{k} i+k+1=\frac{k(k+1)}{2}+k+1$

$= (k+1)\left(\frac{k}{2}+1\right)$

$= \frac{(k+1)(k+2)}{2}$

NO NEED TO EXPLICITLY SHOW THAT THIS IS THE RHS

PIGEON HOLE TECHNIQUE

DEF. 1-1 Function

each input maps to unique output (not necessarily ONTO)

$f: A \to B$ is 1-1 $\iff$ iff $\forall x, y \in A$

if $x \neq y, \Rightarrow f(x) \neq f(y)$

DEF ONTO Function

$f: A \to B$ is ONTO $\iff$ iff $\cancel{\forall y \exists x \text{ s.t. } f(x) = y}$

$\forall b \in B \ \exists a \in A$ s.t. $f(a) = b$

DEF BIJECTION = 1-1 and ONTO

PIGEONHOLE PRINCIPLE

EX: 5 students $\to$ 4 chairs

EX: 400 people $\to$ guaranteed that some share a birthday

400 $\to$ (366 days)

~~If cardinality~~ $\forall$ sets A, B:

IF $|A| > |B|$, THEN $\nexists$ 1-1 $f: A \to B$

i.e. $\exists a_1, a_2 \in A$ s.t. $a_1 \neq a_2$

AND $f(a_1) = f(a_2)$

DOVETAILING

$|A|$ = size of A = No. of elements in A

Compare A, B

ex $A = \{a, b, c\}$
$B = \{1, 2, 3\}$ f: 1-1, ONTO guarantees they have same size

$\exists$ bijection $f: A \to B$
$\underset{\text{iff}}{\Longleftrightarrow}$ $|A| = |B|$

Useful for comparing infinite sets

1) A set A is countably infinite iff $\exists$ bijection $f: N \to A$

2) A set A is countable if it is finite or countably infinite

3) A set is uncountable if it is not countable

ex $A = N - \{1\}$

$\subsetneq$ proper subset

$A \subsetneq N$

is $|A| < |N|$?

NO B/C

N		A
1	$\to$	2
2	$\to$	3
3	$\to$	4
4	$\to$	5

[illegible]

$f(n) = n+1$

ex set of integers

$Z = \ldots -3, -2, -1, 0, 1, 2, 3 \ldots$

$N = 1, 2, 3 \ldots$

is Z bigger ?

N	Z
1	
2	
3	
4	
5	

Think of $\mathbb{Z}$ as:

DOVETAILING

SAW SHAPE

4/7/2010

DOVETAILING

technique for showing a set is countably infinite by mapping to N

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

$1 \to 0$
$2 \to 1$
$3 \to -1$
$4 \to 2$
$5 \to -2$
$\vdots$

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

What about $N \times N$? $\{(n_1, n_2) \mid n_1 \in N \text{ and } n_2 \in N\}$

1-1, but not onto

$\{1\} \times N$ (1,1), (1,2) (1,3)
$\{2\} \times N$ (2,1) (2,2) (2,3)
$\{3\} \times N$ (3,1) (3,2) (3,3)

$= \{1\} \times N \cup \{2\} \times N \cup \{3\} \times N \cup \ldots$

a Union of an infinite number of infinite sets

$\square$ LEAVING ACTUAL FUNCTION AS AN EXERCISE

A countably infinite union of countably infinite sets is countable

so $N \times N \times N \times N \times \ldots$ is countable

FOR $\mathbb{Z}$

the union of 2 countably infinite sets
so $\mathbb{Z} \times \mathbb{Z}$ is countable

Power Set

$Pow(N) = 2^N$ = set of all subsets of N

includes infinite subsets

Can't be ordered by size b/c size is infinite

Suppose $Pow(n)$ is countably infinite
$\Rightarrow \exists$ 1:1, onto $f: N \to Pow(N)$

$f(4)$	N_4	{ n	y	y	n	n	...
$f(3)$	N_3	{ y	y	n	y	y	...
$f(2)$	N_2	{ n	n	y	n	y	...
$f(1)$	N_1	{ y	n	n	y	n	...
	$\vdots$						
	N	1	2	3	4	5	...

$D = \{ n \mid n \in N_n \}$

Differs from every set in our list by at least one element

$D \subseteq N$? yes $\Rightarrow$ has to be a set in the list

$D \subseteq N \Rightarrow D \in Pow(N)$

$\Rightarrow D = N_i$ for some i

Doesn't have to be i, could be 1

Is $i \in D$?

YES : $i \in D \Rightarrow i \in N_i$ CONTRADICTS $D = \{ n \mid n \notin N_n \}$ $i \in D \Rightarrow i \notin N_i$

NO : $i \notin D \Rightarrow i \notin N_i \Rightarrow i \in D$

Method is called Diagonalization

R can be rational or irrational

Real no's between $(0,1)$

Assume 1-1, onto f

N	f	R
1	$\longrightarrow$	0.57602
2	$\longrightarrow$	0.1247
3	$\longrightarrow$	0.00157
4	$\longrightarrow$	0.091779

create new number r that differs from each element in the list for at least 1 digit.

e.g. add 1

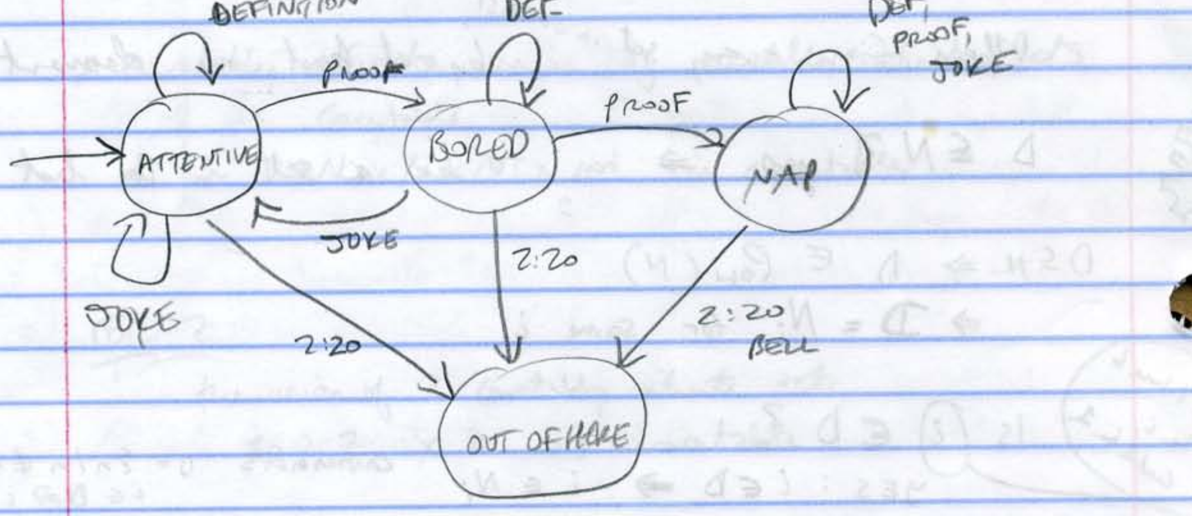
$r = .6328$

4/9/2010

☐ Pigeon Hole can be proved by Induction
(on size of range?)

1) REVIEW OF PROOF TECHNIQUES → ON SLIDES

2) FINITE AUTOMATA: CSE 322 STATES OF MIND

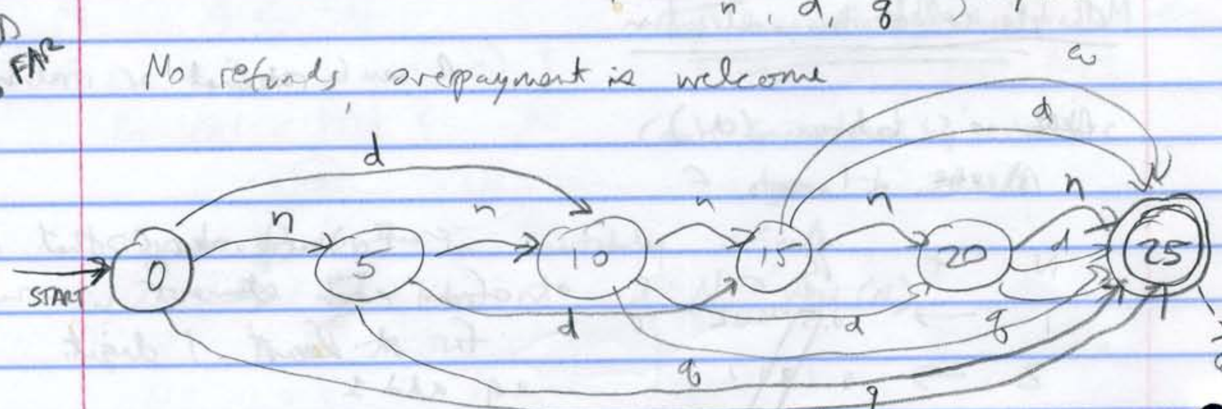


VENDING MACHINE

Gum Costs $0.25
Accepts 5, 10, 25 } inputs
n, d, q

STATES REPR HOW MUCH DEPOSITED SO FAR

No refunds, overpayment is welcome



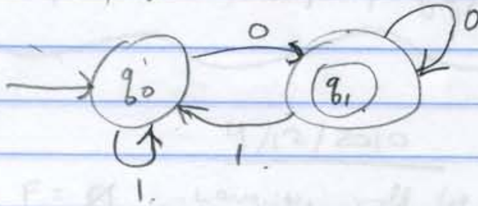
DO ALL INPUTS FOR ALL STATES

Input String: "nnunn" ACCEPT
"nn" REJECT
"q" ACCEPT

STATES = "0" "5", "10" ,..
INPUTS = string over Alphabet $\Sigma = \{n, d, q\}$
TRANSITIONS = "ARROWS"
START STATE : "0" state
SET OF ACCEPT STATES:

EXAMPLE OF A "FINITE AUTOMATON" Call it M1

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

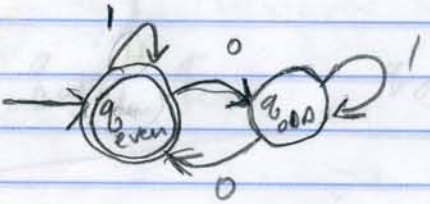


ACCEPTS ALL STRINGS THAT END WITH 0.

"Automaton" == Computer

language recognized by this machine = $\{w \mid w \text{ ends in } 0\}$ $L(M_1)$

M_2
$L(M_2) = \{ w \mid w$ contains an even num of 0's$\}$



Parallel? NOT THESE (DFA's)
NFA's do Parallel

Process 1101 with M_2

CONFIGURATIONS

$w = 1101$

$q_E \underline{1}101$

$1 q_E \underline{1}01$

$11 q_E \underline{0}1$

$110 q_{ODD} \underline{1}$

$1101 q_{ODD}$ Reject w

$w = 1010$

$q_E \underline{1}010$

$1 q_E \underline{0}10$

$10 q_O \underline{1}0$

$101 q_O 0$

$1010 q_E$ → Accept w

FORMAL DEF. OF A F.A

- Set of States Q
- Alphabet Σ.
- Transition Function : $\delta : Q \times \Sigma \rightarrow Q$

$$(q, a) \longrightarrow q'$$

STATE INPUT

- 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 LOOP FOREVER
→ NO B/C INPUT IS FINITE

δ	0	1
q_{ev}	q_{odd}	q_{even}
q_{odd}	q_{even}	q_{odd}

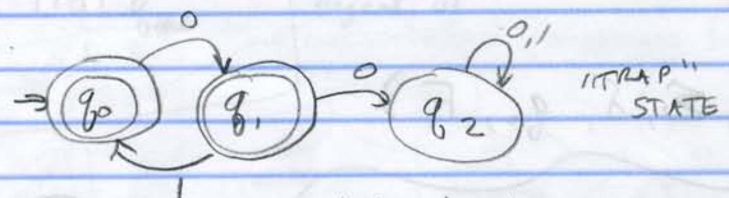
ONE WAY TO SHOW δ

Computation by Finite Automaton

$w_i \in \Sigma$

F.A. M accepts INPUT $w = w_1 w_2 w_3 \ldots w_n$
iff $\exists$ $n+1$ states $r_0, r_1, \ldots r_n \in Q$
s.t.
1. $r_0 = q_0$
2. $\delta(r_i, w_{i+1}) = r_{i+1}$ $\forall i \in \{0, 1, 2, \ldots, n-1\}$
3. $r_n \in F$

EXAMPLES



$L(M) = \{w \mid w$ DOES NOT CONTAIN $00\}$

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

TO recognize complement $\overline{L(M)} = \{w \mid w \text{ contains } 00\}$
$F = \{q_2\}$

General Rule: Flip the accept + reject states

$L(M_3) = \{w \mid w$ has even # of 0's and odd # of 1's$\}$

WHAT ARE POSSIBLE COMBO'S?

even 0's, even 1's
$\boxed{\text{even 0's, odd 1's}}$
odd 0's, odd 1's
odd 0's, even 1's

one way to look at this problem

Figure out which combo is the start state ($q_{e,e}$)
" " " accept "

What about $\overline{L(M_3)}$: switch reject & accept

$L(M') = \{w \mid w \text{ contains EVEN \# of 0's}\}$
$F = \{q_{e,e}, q_{e,o}\}$

How can we reduce to 2 states?

There's no unique machine for any language.

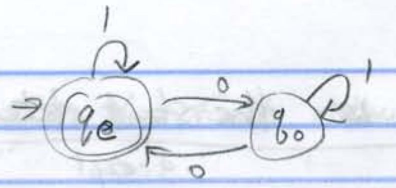
Definition: A language L is regular iff some F.A. M recognizes it.
i.e. $L(M) = L$

eg. $L(M_2) = \{w \mid w \text{ contains even \# of 0's}\}$

$L_1 = \{w \mid w \text{ cont. even \# of 0's}\}$
$L_2 = \{w \mid w \text{ ENDS IN } 0\}$

$L_1 \cap L_2$ Regular? ← CLOSURE OPERATION

M_1



M_2

$L_1 \cap L_2 = \{ w \mid w$ has even 0's AND ends in 0$\}$

<u>closure</u>

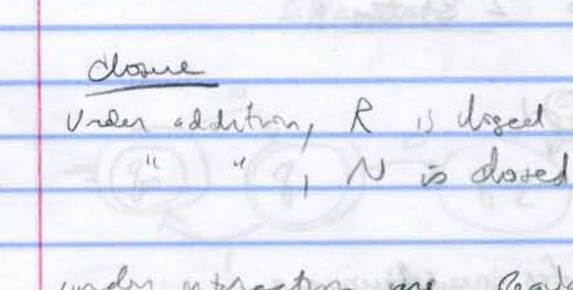
Under addition, R is closed

" " " , N is closed

under intersection are Regular Languages closed?

We already showed under complementation, regular lang's are closed

Make states for each pair of states

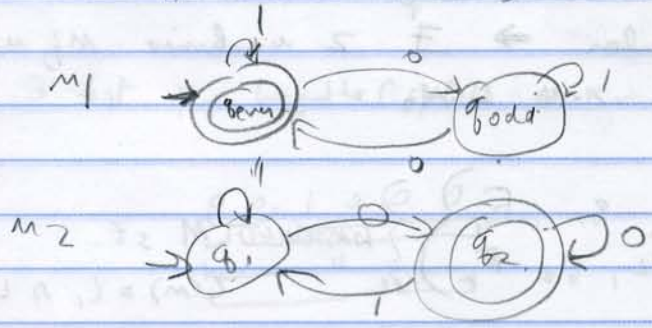


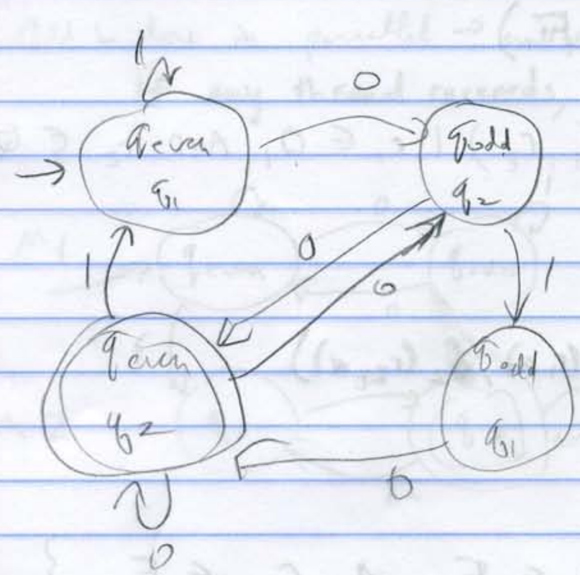
4/14/2010

4f → 000 →
Should not accept 000

$L_1 = \{w \mid w$ cont. even # of 0's$\}$
$L_2 = \{w \mid w$ ends in $0\}$
$L = L_1 \cap L_2$
L_1, L_2 REG ⇒ $L_1 \cap L_2$ REG?

BUILD A NEW THAT SIMULATES M_1 + M_2 in parallel





$L = L_1 \cup L_2$

states are same - just change set of accept states

(add 2 more)

L_1, L_2 reg $\Rightarrow L_1 \cup L_2$ is reg.

$F = \{ (q_{even}, q_1), (q_{odd}, q_2), (q_{even}, q_2) \}$

THM: Reg Lang's ARE CLOSED UNDER $\cap$ (and $\cup$)

i.e. $L1, L2$ reg $\Rightarrow L1 \cap L2$ REG

PF $L1, L2$ are regular $\Rightarrow \exists$ 2 machines M_1, M_2 s.t. $L(M_1) = L_1$ AND $L(M_2) = L_2$

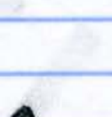
$M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$
$M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$

CONSTRUCT M s.t. $L(M) = L_1 \cap L_2$

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

$Q = Q_1 \times Q_2 = \{ (r_1, r_2) \mid r_1 \in Q_1 \text{ AND } r_2 \in Q_2 \}$

$\Sigma = \Sigma$



$\delta((r_1, r_2), a) = (\delta_1(r_1, a), \delta_2(r_2, a))$

$q_0 = (q_1, q_2)$

$F = \{ (r_1, r_2) \mid r_1 \in F_1 \wedge r_2 \in F_2 \}$

$= F_1 \times F_2$

For Union:

$$F = \{(r_1, r_2) \mid r_1 \in F_1 \lor r_2 \in F_2\}$$

For concatenation

$$L_1 \circ L_2 = \{w_1 w_2 \mid w_1 \in L_1 \land w_2 \in L_2\}$$

w = 0010

can it be split to be $\in L_1 \circ L_2$

$\underbrace{00}_{w_1}\underbrace{10}_{w_2}$ or $\underbrace{\varepsilon}_{w_1}\underbrace{0010}_{w_2}$

w = 001

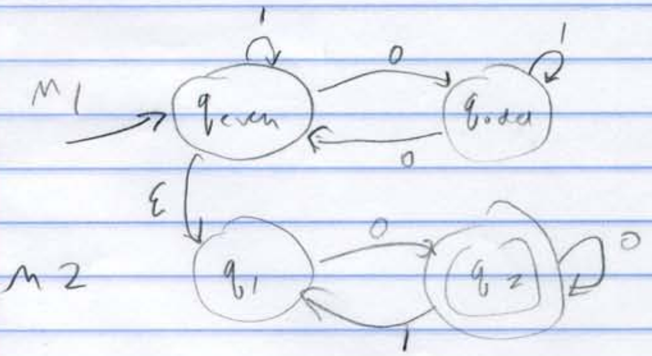
no way to split that works

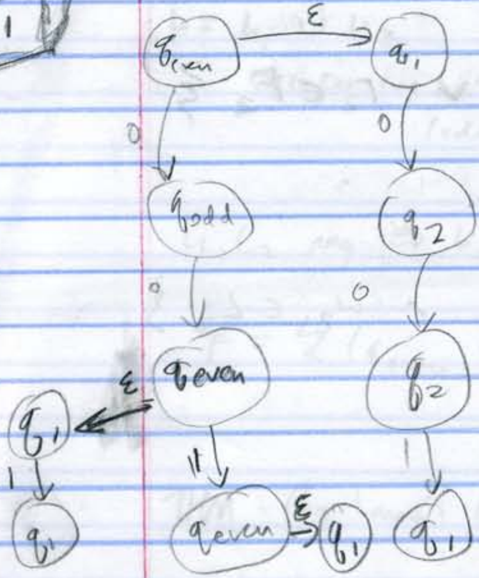
To test you have to split w in diff. ways.

0 0 1 0 0 0

Could be done in parallel → multiple threads

If any thread succeeds, accept





branch every time you enter q_{even}, even if a self-loop

DON'T ACCEPT B/C ALL 4 FINAL STATES ARE REJECTS

Each branch does a different split

$\underbrace{001}$

$\underbrace{00}\,1$