1-1

Theory of Computation

- What are computers? What can they do?
  What can't they do?

Set Theory

$\{1, 2, 3\}$ $\{1, 1, 2, 3\}$

$X \underset{\text{is a subset}}{\subseteq} Y$ iff $\forall a, a \in X \rightarrow a \in Y$

$\mathbb{N}$ – naturals $\mathbb{Z}$ – integers $\mathbb{N} \subseteq \mathbb{Z}$

$\mathbb{R}$ – reals $\mathbb{Q}$ – rationals

$a \in X \underset{\text{union}}{\cup} Y$ iff $a \in X$ or $a \in Y$

$a \in X \underset{\text{intersect}}{\cap} Y$ iff $a \in X$ and $a \in Y$

$a \in \underset{\text{complement}}{X^c} = \overline{X}$ iff $a \underset{\text{not in}}{\notin} X$ and $a \in \mathcal{U}$

Tuples

$(0, 1)$ is a two-tuple (or pair) of 0 and 1

X is a tuple, then $\pi_i(\bar{x})$ is the ith thing

$\pi_0(0, 1) = 0$

$X \underset{\text{cross product}}{\times} Y = \{ (x, y) \underset{\text{such that}}{|} x \in X \text{ and } y \in Y \}$

Relation R on X, Y, and Z is a subset of $X \times Y \times Z$

pluses : $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$

pluses 0 0 0

$\forall n.$ pluses 0 n n

$\forall n, m.$ pluses $(1+n)$ m $(1+x)$ $\Leftarrow \exists x.$ pluses n m x

-2/ A function F from $X \to Y$ is a relation on $X \times Y$ s.t.

$\forall x, y_1, y_2,\ F\ x\ y_1 \wedge F\ x\ y_2 \Rightarrow y_1 = y_2$

succ : { (0,1)
(1,2)
(3,4)
(99,100) }
...

Reflexive: $\forall x.\ R\ x\ x$

Symmetric: $\forall x,y.\ R\ x\ y \Rightarrow R\ y\ x$

Transitive: $\forall x,y,z.\ R\ x\ y \wedge R\ y\ z \to R\ x\ z$

Powerset: $\mathbb{P}(X) = P(X) = 2^X$

$a \in P(X)$ iff $a \subseteq X$

$X = \{0,1\}$ $\quad P(X) = \{ \underset{00}{\{\}}, \underset{10}{\{0\}}, \underset{01}{\{1\}}, \underset{11}{\{0,1\}} \}$

Sequence of X is a function from $N \to X$
(a string if there's also a length N) $(N, N \to X)$

A string of Σ is a finite sequence of Σ
sigma, the alphabet $\qquad |\alpha| =$ length

0110 $\qquad$ Jay

ε – epsilon, empty string

x^R – reverse, switch direction $\quad f(x) = f(|str| - x)$

$x \circ y = xy$, concatenate

$f(i) = x(i)$ if $i < |x|$
$\qquad y(i - |x|)$ o.w.

~~x^n = n copies of x~~

Strings can be ordered (lexicographically)

x^* = kleene star (not a string) a set of strings

$s = x^n$ of some n, then $s \in x^*$

Devangari

"Language of Σ" is a set of strings of Σ

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 comptute the factorial of 25.

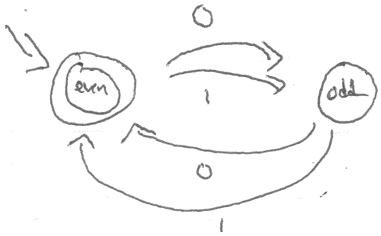
The job of the compute is "recognition"
given a string x
return Yes if x is in the set
No if is NOT in the set

~~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 , $X \Rightarrow \{Y, N\}$

2-1/ Deterministic Finite Automata

Even-ness :
over binary



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

○ is a node or a state

↘○ is the start state

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

◎ is the accept state

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

even odd even

$_a○ \xrightarrow{c,d} ○_b$ (on c or d)

$_a○ \longrightarrow ○_b$ (on anything)

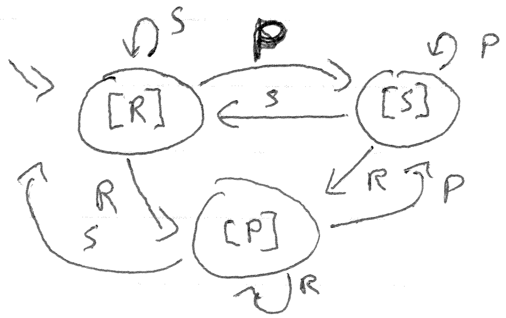
Machine with output

$c \in \Sigma'$

([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