18-1

Church - Turing Thesis

Alonzo Church ⇒ lambda - calculus
first programming language

Alan Turing ⇒ Turing machine
first "elegant" general-purpose computer

Church - λ-calculus - software
Turing - TM - hardware

"Any computation can be implemented w/ λ-calculus or TMs"

"computation" $\xleftrightarrow{\text{implement}}$ λ

"computation" $\xleftrightarrow{\text{implement}}$ TM

λ = TM
1. λ ⇒ TM ?
2. TM ⇒ λ ✓

λ = software = all PLs
TM = hardware = all real computers

TMs = Deciders & Recognizers

Deciders: may ~~not~~ diverge → "useful"

Recognizers: may diverge → "not useful"

ALL - not possible
Σ_1 - not useful
Σ_0 - useful

18-2

Properties of TMs

Σ_0 (deciders)

Closed under union?

$A \in \Sigma_0$ and $B \in \Sigma_0$ then $A \cup B \in \Sigma_0$?

totally possible

$w\#w \in \Sigma_0$ $\quad a^n b^n c^n \in \Sigma_0$ $\quad (w\#w) \cup (a^n b^n c^n) \in \Sigma_0$?

Σ_0 ✓

```
TM union (TM x, TM y) = bool (word w) {
    if (x(w)) then return true;
    else if (y(w)) then return true;
    else return false; }.
```

Σ_1 ✓ run 1 step of x, then 1 step of y, return true if any does

---

Concatenation

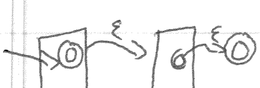
Σ_1 ✓ Σ_0 ✓

$uv \in A \circ B$ if $u \in A$ and $v \in B$

input: w

for all i between 0 and $|w|$
divide w at i into u and v,
then run x(u) && y(v)
} Σ_0

NFAs



non-deterministically choose i } Σ_1

Kleene

$w \in A^*$ means $w = w_0 \ldots w_n$ where $w_i \in A$

$A = w\#w$ $\quad A^* = (w\#w)^*$ $\quad$ "dog#dog Jay#Jay ☺#☺" $\quad n = 2$

$w_0 =$ dog#dog $\in A = w\#w$

$w_1 =$ Jay#Jay $\in A$

$w_2 =$ ☺#☺ $\in A$

$m \in [0, |w|]$

Σ_0 ✓ $\quad \Sigma_1$ ✓

Intersect

18-3

Complement

$w \in A^c$ if $w \notin A$ f is a TM for A

✓ Σ_0: input: w   do: if(f(w)) { ret f } else{ret t}

✗ Σ_1:

future claim

if ($\underbrace{f(w)}$) {
ret F
ow
ret T

if $w \in A$, then it rets T → ✓
if $w \notin A$, then it ~~rets~~ rets F → ✓
OR diverges

not possible
1. figure that it is diverging
2. stop, and say yes

---

if Σ_1 were closed under complement
then $\Sigma_0 = \Sigma_1$