Lecture 2

Turing Machines

Q = finite set of states
Σ = input alphabet (finite)
Γ = tape alphabet (finite)

œ ∈ Γ \ Σ

q0 ∈ Q start state
qA ∈ Q accept state
qR ∈ Q reject state

Transition function
δ: Q × Γ → Q × Γ × {L, R}

Classroom MGH 058 starts Monday

w ∈ Σ* input
1wl = n

conversion: if more than left end of tape has a L, just stay there.

eg [Ly#y: ye2012*] y because why
Formalize computation

"Consequent":

snapshot of a TM on an input at step $n$

- current state
- tape contents
- head position of TM

$Q \Gamma = \emptyset$ left $\Gamma$ right $\emptyset$
Configure C

\[ \delta(q, a) = (p, b, R) \]

\[ uqav \rightarrow_{M} ubpv \]

\[ \delta(q, a) = (p, b, R) \]

\[ qa + M pbv \]

\[ ueqav \rightarrow_{M} uqcbv \]

\[ \delta(q, a) = (p, b, R) \]

\[ uqa + M ubqo \]

Nothing from uqaev uv

uqajuv

\[ C + M D \rightarrow \text{yields in same number of steps} \]

Start configure on input uv?
\[ q_0 \omega \]

\[ \text{Set } M \text{ accepts } \omega \text{ iff } q_0 \omega \vdash^* u \text{ acc } v \text{ for some } u, v. \]

\[ M \text{ rejection } \omega \text{ iff } q_0 \omega \vdash^* u \text{ rej } v \text{ for some } u, v. \]

**Defn** \[ L(M) = \{ \omega | M \text{ accepts } \omega \} \]

Language \underline{recognized} by \( M \).

**Defn** \( M \) is a decoder iff for every \( \omega \in \Sigma^* \), \( M \text{ accepts } \omega \) or \( M \text{ rejects } \omega \).
M decides \( L \) if
\[
L(M) = \mathcal{L}
\]
and \( M \) is a decider.

Notation for Mode for each

DFA

\[
q \quad \overset{a}{\rightarrow} \quad p
\]

TM diagram

\[
\Delta(q, a) = (p, b, \lambda)
\]

\( \{0^2^n : n \geq 0\} \)

scanning to see if exactly \( n \equiv 0 \mod 0 \)

if so, accept.
A language \( L \) is \( \overline{\text{recognizable}} \) if there is a TM \( M \) such that \( L = L(M) \).

A language \( L \) is \( \overline{\text{decidable}} \) if there is a decider TM \( M \) such that \( L = L(M) \).