1. Turing Machine

    Finite State Machines
      - Regular Languages
      - Input String (iterated once)
    Pushdown Automata
      - Context-free Languages
      - Stack
    Turing Machines
      → Decidable, Turing Recognizable
      and Unrecognizable Languages
      → Tape
        - symbols from alphabet Σ
        - special blank symbol, $\sqcup \notin \Sigma$
        - infinite in one direction (blanks)
        - current position (can go back)
      → Operations on "Tape"
        → read current symbol
        → update (write) same cell
        → move exactly one cell
      → Structure of "Tape"
        ○ $\xrightarrow{a \to b, R}$ ○   read a, write b, move right
      exactly 2 final states: Accept and Reject

Languages:
→ Decidable (Recursive/Computable/Solvable)
  When given input string, TM will always halt.
    → Accept if it is in L.
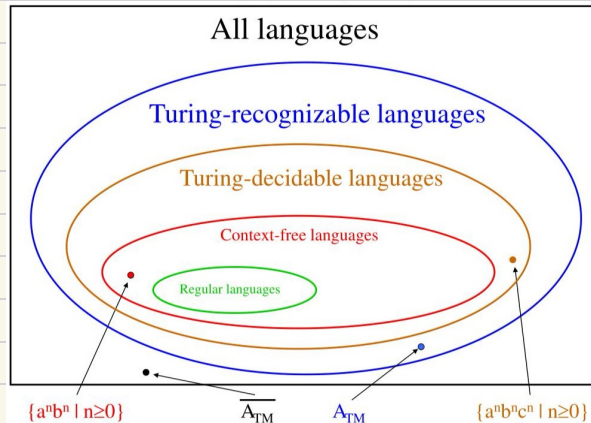   → Reject if it is not in L.
→ Turing Recognizable (Recursively Enumerable/Partially Decidable/Semi-Decidable)
  When given string in the language, TM always halts and accepts.
  When given string not in the language, TM either loops or rejects
→ Not Turing Recognizable (Not Recursively Enumerable/Not Partially Decidable)
  Can't even recognize members reliably

All languages

Turing-recognizable languages

Turing-decidable languages

Context-free languages

Regular languages

$\{a^n b^n \mid n \geq 0\}$ $\overline{A_{TM}}$ A_{TM} $\{a^n b^n c^n \mid n \geq 0\}$

<u>Formal Definition</u> of Turing Machine
$T = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$
→ Q - set of states
→ Σ - input alphabet, $\sqcup \notin \Sigma$
→ Γ - tape alphabet, $\sqcup \in \Gamma$, $\Sigma \subseteq \Gamma$
→ q_0 - initial state, $q_0 \in Q$
→ $q_{accept}, q_{reject} \in Q$
$\delta: Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$

2. Nondeterministic Turing Machine

$$\delta: Q \times \Gamma \to \mathcal{P}(Q \times \Gamma \times \{L, R\})$$

Each moment in computation, there can be $\geq \underline{1}$ successor configuration.

Tree representation of computation of nondeterministic TM:

→ Accept if at least one branch accepts.

→ Reject if all branches reject.

→ Loop if some branch is infinite

Th] Every nondeterministic TM has an equivalent deterministic TM.

⎛ Input tape - initial input, never modified ⎞
⎜ Simulation tape - perform branch computation ⎟ BFS on decision tree
⎝ Address tape - control BFS on decision tree ⎠

3-tape deterministic TM

Th] Every Multitape TM has an equivalent Single-tape TM.

3 tapes → #tape1 #tape2 #tape3 #⊔... - single tape

Def.] Turing Recognizable language - language recognized by nondeterministic TM

Def] Decidable Language - language decided by nondeterministic TM. (no loop)