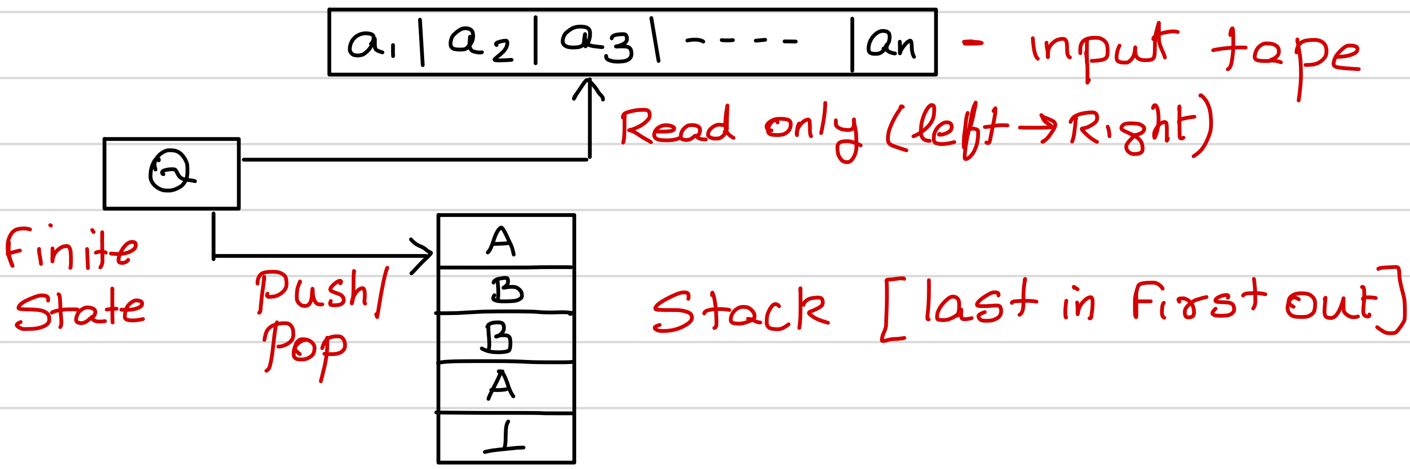
Lecture 21

Non deterministic pushdown automata (NPDA)

Finite automata + a stack.



Working of the machine

- pops the top symbol off the stack.
- Makes a transition based on the top of the stack, input symbol and current state.

Transition: push a sequence of symbols onto the stack, change state, move the read head one cell to the right.

ϵ-transitions are allowed: Machine can pop and push without reading an input symbol or moving the input head pointer.

Stack can store unbounded information but access is limited.

Definition of a nondeterministic PDA.

$$M = (Q, \Sigma, \Gamma, \delta, s, \bot, F)$$

Q - finite set of states, Σ - finite set : input alphabet

$s \in Q$ - start state, $F \subseteq Q$: set of final / accept states.

Γ - finite set : stack alphabet

$\bot$ - initial stack symbol.

$$\delta \subseteq (Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma) \times (Q \times \Gamma^*)$$

$((p, a, A), (q, \underset{\text{last}}{B_1}, \ldots \underset{\text{First}}{B_k})) \in \delta$: Example 1

$((p, \epsilon, A), (q, B_1 \ldots B_k)) \in \delta$: Example 2

$M = (Q, \Sigma, \Gamma, \delta, s, \perp, F)$

Configurations: A configuration of M is an element of $Q \times \Sigma^* \times \Gamma^*$.

Q ↓ current state

Σ^* ↓ part of the input yet unread

Γ^* ↳ current content of the stack.

Start Configuration: $(s, x, \perp)$

Define: 1 step next configuration relation $\xrightarrow[M]{1}$

if $((p, a, A), (q, \gamma)) \in \delta$ then for any $y \in \Sigma^*$ and $\beta \in \Gamma^*$

$$(p, ay, A\beta) \xrightarrow[M]{1} (q, y, \gamma\beta)$$

if $((p, \epsilon, A), (q, \gamma)) \in \delta$ then for any $y \in \Sigma^*$ and $\beta \in \Gamma^*$

$$(p, y, A\beta) \xrightarrow[M]{1} (q, y, \gamma\beta)$$

Let $\xrightarrow[M]{*}$ denote the reflexive transitive closure of $\xrightarrow[M]{1}$.

Acceptance. Two types: By final state and empty stack.

By final state: M accepts x by final state if

$$(s, x, \perp) \xrightarrow[M]{*} (q, \epsilon, \gamma) \text{ for some } q \in F, \gamma \in \Gamma^*$$

↳ can be any string.

By empty stack. M accepts x by empty stack if

$$(s, x, \perp) \xrightarrow[M]{*} (q, \epsilon, \epsilon) \text{ for some } q \in Q.$$

↳ can be any state

$L(M)$ - set of all strings $x \in \Sigma^*$ accepted by M.

Example : Set of balanced parentheses.

NPDA - accepting by empty stack.

$Q = \{q\}$, $\Sigma = \{[,]\}$, $\Gamma = \{\perp, [\}$ $s = q$.

$((q, [, \perp), (q, [\perp)) \in \delta$.

$((q, [, [), (q, [[)) \in \delta$

$((q,], [), (q, \epsilon)) \in \delta$.

$((q, \epsilon, \perp), (q, \epsilon)) \in \delta$.