Push Down Automata

Beulah A.
AP/CSE

Introduction

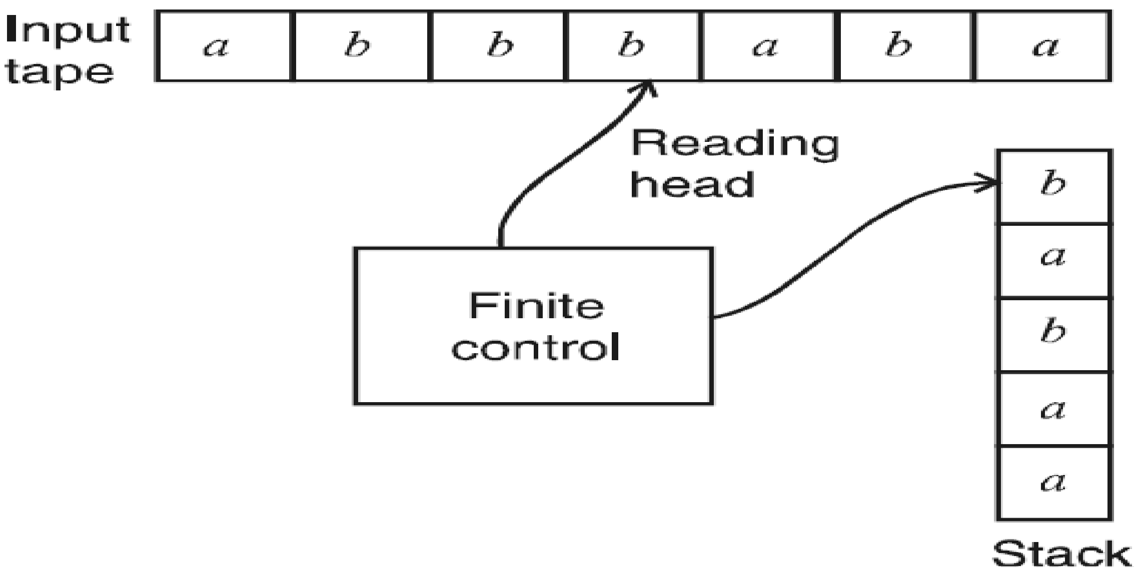
- The regular languages → the finite automaton.
- Context free language → push down automata.
- Finite automata cannot recognize all languages. Because some languages are not regular.
- Finite automata have strictly finite memories, whereas recognition of contextfree language may require storing an unbounded amount of information.
- Push down automata is a machine similar to finite automata that will acceptcontext free languages, except more powerful.

Example

- $L=\{a^n b^n : n \geq 0\}$
- $L = \{ww^R : w \in \{a,b\}^*\}$

Push Down Automata

- Finite automaton with control of both an input tape and a stack (or) Last in-first out (LIFO) list.



Definition

- $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$, where
- Q is a finite set of states.
- Σ is finite set of alphabet
- Γ is finite set of stack alphabet
- $q_0 \in Q$ is the start state (or) initial state
- z_0 in Γ is a particular stack symbol called start symbol.
- $F \subseteq Q$ is the set of final (or) favorable states.
- $\delta \rightarrow Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma^* \rightarrow 2^{Q \times \Gamma^*}$

Compare FA and PDA

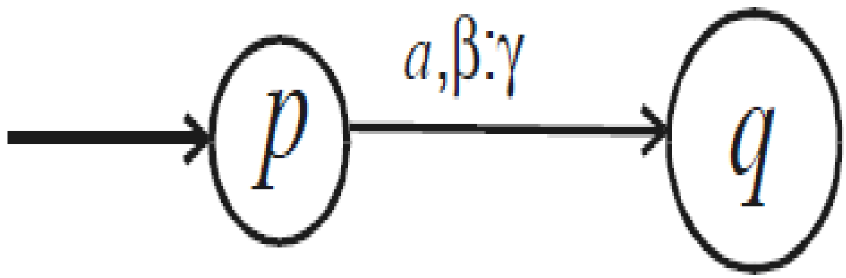
- $\delta(p, a) = q$ → M is in state p, then on reading 'a' from input tape go to state q.
- $\delta(p, \varepsilon) = q$ → M is in state p, goes to state q, without consuming input.

Compare FA and PDA

- $\delta(p, a, \beta)=\{(q,y)\} \rightarrow$ M is in state p, the symbol read from input tape is 'a',and β is on top of stack, goes to state q, and replace β by y on top of stack.
- $\delta(s, a, e)= \{(s, a)\} \rightarrow$ M is in state s, reads 'a', remains in state s and push a onto stack (e-empty stack).
- $\delta(s, c, e) = \{(f, e)\} \rightarrow$ if read 'c' in state s and stack is empty, goes to final state f and nothing to push onto stack.
- $\delta(s, e, e) = \{(f, e)\}$
- $\delta(p, q, a) = \{(p, \varphi)\}$
- PDA's are non-deterministic.

Transition Diagram

- $\delta(p, a, \beta) = \{(q, \gamma)\}$



Instantaneous Description (ID)

- An ID is a triple (q, w, γ) where
  - q is the current state
  - w is the remaining input
  - γ is the stack contents.
- $P = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$
- The instantaneous descriptions of pushdown automata is such that

  $(q_1, aw, bx) \mid\!- (q_2, w, yx)$ is possible if and only if $(q_2, y) \in \delta(q, a, b)$

Instantaneous Description (ID)

- $(q, x, \alpha) \mid\!-^{*} (q_1, y, \beta)$ represents n moves, we write $(q, x, \alpha) \mid\!-^{n} (q_1, y, \beta)$

- In particular $(q, x, \alpha) \mid\!-^{0} (q, x, \alpha)$.

Two types of transitions

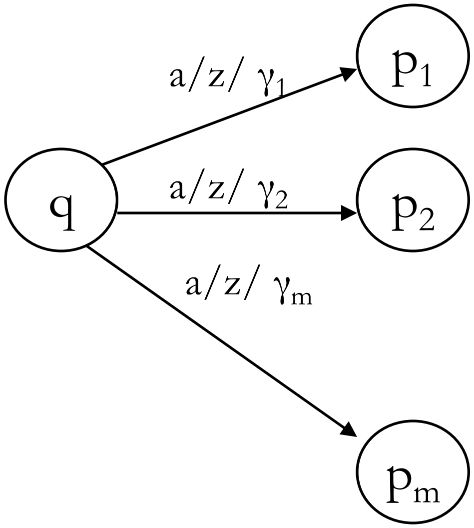
- $\delta(q, a, z) = \{(p_1, \gamma_1), \ldots\ldots\ldots (p_m, \gamma_m)\}$

  q and p_i, $1 \leq i \leq m$ are states,

  $a \in \Sigma$

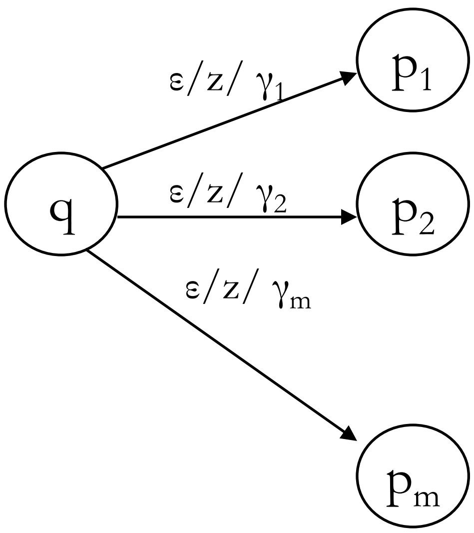
  $z \in \Gamma$

  $\gamma_i \in \Gamma^*$ $1 \leq i \leq m$,



Two types of transitions

- $\delta(q, \varepsilon, z) = \{(p_1, \gamma_1)\ (p_2, \gamma_2), \ldots\ldots.\ (p_m, \gamma_m)\}$



Languages of PDA

- Acceptance by empty stack
  - Let $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ be a PDA.
  - The language accepted by empty stack is denoted by $L_E(M)$

$$L_E(M) = \{w \mid (q_0, w, z_0) \vdash^* (p, \varepsilon, \varepsilon) \text{ for some } p \text{ in } Q\}$$

Languages of PDA

- Acceptance by final state
  - Let $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ be a PDA.
  - The language accepted by final state is denoted by $L_F(M)$

$L_F(M) = \{w \mid (q_0, w, z_0) \mid\!\!—^* (p, \varepsilon, \gamma)$ for some p in F and γ in $\Gamma^*\}$

Languages of PDA

- Acceptance by final state and empty stack
  - Let $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ be a PDA.
  - The language accepted by empty stack and final state is denoted L(M)

$L(M) = \{w \mid (q_0, w, z_0) \vdash^* (p, \varepsilon, \varepsilon)$ for some p in $F\}$