Theory of Computation (TOC) set A (18ESJCS169) Vaibhav Saran ①
4CS4-06.

Q1 A finite automata can be used only to accept regular languages whereas pushdown automata is a finite automata with an extra memory called stack which helps pushdown automata to recognize context free languages. The pushdown automata (PDA) will have i/p tape, finite control and stack.

I/p tape: It is divided in many cells. At each cell only 1 i/p symbol is placed thus certain i/p string is placed on tape.

finite control: It is a type of pointer which point to the symbol which is to be read. Δ is placed at end to indicate end of string

Stack: It is a structure in which you can add & remove the items from one end only.

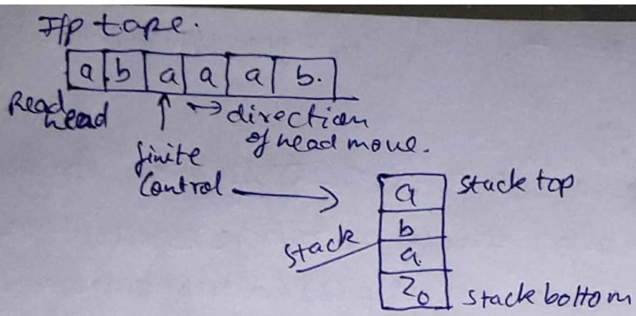
* PDA can be defined as a collection of 7 components:

(i) Q: finite set of states. (ii) Σ: set of i/ps.

(iii) Γ: stack alphabet (iv) q_0: initial stage.

(v) Z_0: start symbol in stack (vi) F: set of final states $F \subseteq Q$

(vii) δ: mapping f^n used for moving from current state to next state.



PDA is can recognize context free languages also: Say for eg. PDA for $L = \{a^n b^n \mid n \geq 0\}$

Now, $M = (Q, \Sigma, \delta, S, F, \Gamma, Z_0)$

$\Sigma = \{a, b\}$, $\Gamma = \{A, B, Z_0\}$

$Q = \{q_0, q_1, q_2, q_3\}$, $F = \{q_3\}$

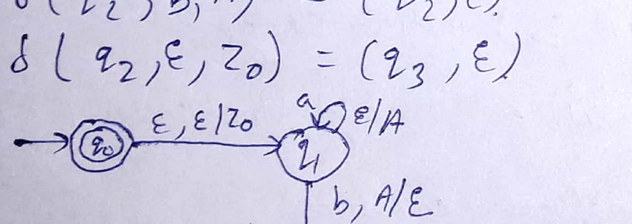
$\delta(q_0, \epsilon, \epsilon) = (q_1, Z_0)$

$\delta(q_1, a, \epsilon) = (q_1, A)$

$\delta(q_1, b, A) = (q_2, \epsilon)$

$\delta(q_2, b, A) = (q_2, \epsilon)$

$\delta(q_2, \epsilon, Z_0) = (q_3, \epsilon)$



for string aabb

$(q_0, a, \epsilon) \rightarrow (q_1, AZ_0)$

$(q_1, a, AZ_0) \rightarrow (q_1, AAZ_0)$

$(q_1, b, AAZ_0) \rightarrow (q_2, b, AZ_0)$

$(q_2, b, AZ_0) \rightarrow (q_2, Z_0)$

$(q_2, \epsilon, Z_0) \rightarrow (q_3, \epsilon)$

PDA halts & accepts it.

Q2 (a) (i) $S \to aSa \mid bSb \mid a \mid b \mid \varepsilon$.

Now

$S \to aSa$
↳ $S \to \varepsilon$ $\Rightarrow aa$.
↳ $S \to a$ $\Rightarrow aaa$.
↳ $S \to b$ $\Rightarrow aba$.

$S \to bSb$ $\Rightarrow bb$.
↳ $S \to \varepsilon$
↳ $S \to a \Rightarrow bab$
↳ $S \to b \Rightarrow bbb$

$S \to a$.
$S \to b$
$S \to \varepsilon$.

∴ language generated is $L = \{\varepsilon, a, b, aa, bb, aaa, bbb, aba, bab, \ldots\}$

(ii). $S \to aSa \mid B$
$B \to bB \mid a$.

$S \to aSa$.
↳ $S \to B \to a$ $\Rightarrow aaa$.

$S \to aSa$.
↳ $S \to B \to bB \Rightarrow ba \Rightarrow abaa$.
↳ $B \to a$.

$S \to B$ $\Rightarrow a$.
↳ $B \to a$.
↳ $B \to bB$ $\Rightarrow ba$
↳ $B \to a$.

∴ language generated is $L = \{a, ba, aaa, abaa, \ldots\}$.

(b) $S \rightarrow bbA$

$A \rightarrow Bb$

$B \rightarrow aAa \mid \epsilon$.

(18EJICS169) Vaibhav Saran (4)

To check (bb aa baab.) is accepted or not.

$S \rightarrow bbA$

* $\hookrightarrow A \rightarrow Bb$. $\Rightarrow$ bbb

$\hookrightarrow B \rightarrow \epsilon$

* $\hookrightarrow A \rightarrow Bb$. $\Rightarrow abab \Rightarrow$ bbabab.

$\hookrightarrow B \rightarrow aAa$. $\Rightarrow aba$.

$\hookrightarrow A \rightarrow Bb$. $\Rightarrow b$

$\hookrightarrow B \rightarrow \epsilon$

$\therefore L = \{bbb, bbabab, \ldots\}$

bb aabaab.

$S \rightarrow bbA$ $\Rightarrow$ $S \rightarrow bbBb$. $\Rightarrow$ $S \rightarrow bbaAab$. (3)

$\hookrightarrow A \rightarrow Bb$. (1) $\quad \hookrightarrow B \rightarrow aAa$. (2) $\quad \hookrightarrow A \rightarrow Bb$

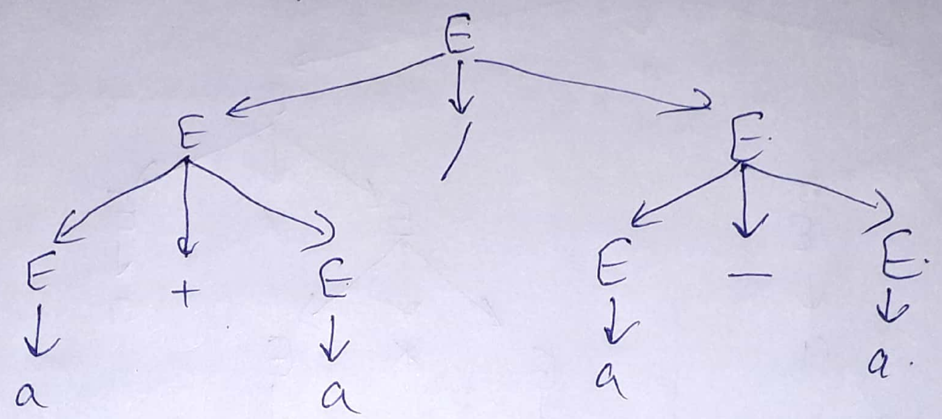
$\Downarrow$

$\Rightarrow S \rightarrow bbaBbab$ (4)

in step (4) we observe that B can be replaced by aAa or ϵ but in neither case it is the desired string, so bbaabaab is not generated by this grammar.

Q3 $E \rightarrow E+E \mid E-E \mid E*E \mid E/E \mid (E) \mid id.$

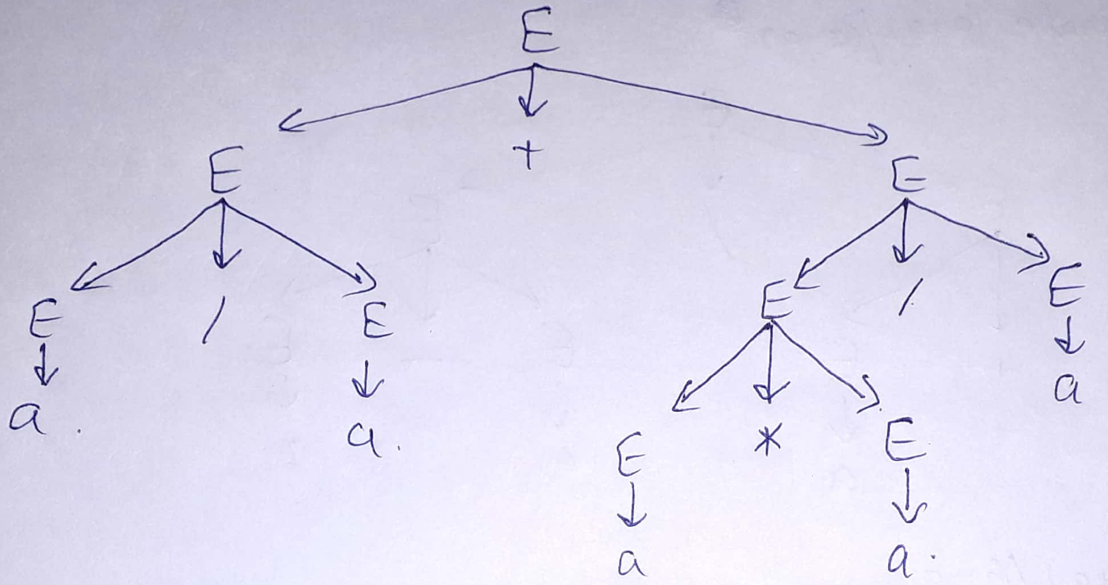
(a) generate $(a+a)/(a-a)$



$\Rightarrow (a+a)/(a-a)$

(b)

(b) generate $a/a + (a*a)/a$.



$\Rightarrow a/a + (a*a)/a$.

Given:

Q4 PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$

$Q = \{q_0, q_f\}$ $\quad \Sigma = \{a, b\}$

$F = \{q_f\}$. $\quad \Gamma = \{a\}$.

$\delta(q_0, a, \varepsilon) = (q_0, a)$

$\delta(q_0, b, \varepsilon) = (q_0, a)$

(A change has been made here → $\delta(q_0, a, a) = (q_f, \varepsilon)$

$\delta(q_f, a, a) = (q_f, \varepsilon)$

$\delta(q_f, b, a) = (q_f, \varepsilon)$

(a) for string aba.

aba

$\delta(q_0, a, \varepsilon) = (q_0, a)$

$\delta(q_0, b, a) =$ no transition available.

∴ the final state is not reached and hence aba is not acceptable by given PDA

for string abb.

abb

$\delta(q_0, a, \varepsilon) = (q_0, a)$

$\delta(q_0, b, a)$. = no transition available.

∴ the final state is not reached and hence abb is not acceptable by given PDA.

(b) for string baa.

baa.

$\delta(q_0, b, \varepsilon) = (q_0, a)$

$\delta(q_0, a, a) = (q_f, \varepsilon)$

$\delta(q_f, a, a) = (q_f, \varepsilon)$

It is acceptable.

for string baaaa.

$\delta(q_0, b, \varepsilon) = (q_0, a)$

$\delta(q_0, a, a) = (q_f, \varepsilon)$

$\delta(q_f, a, a) = (q_f, \varepsilon)$

$\delta(q_f, a, a) = (q_f, \varepsilon)$

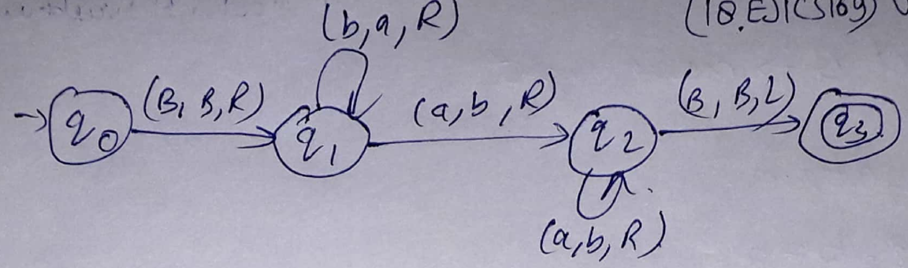
$\delta(q_f, a, a) = (q_f, \varepsilon)$

It is acceptable.

Hence string baa & baaaa are acceptable by given PDA.

(18EJICS169) Vaibhav Saran (9)

Q5 (i)



(ii). aabb $\rightarrow$ $\delta(q_0, B) \rightarrow \delta(q_1, B, R)$

$\delta(q_1, a) \rightarrow \delta(q_1, b, R)$

$\delta(q_1, a) \rightarrow \delta(q_2, a, R)$

$\delta(q_2, b) \rightarrow \delta(q_2, a, R)$

$\delta(q_2, b) \rightarrow \delta(q_2, a, R)$

$\delta(q_2, B) \rightarrow \delta(q_3, B, L)$

as it reach final state so accepted.

abab

Initial $\rightarrow$ B $\rightarrow$ move right

$q_0 \underline{a}bab$

$q_1 a\underline{b}ab$

$q_1 a q_2 ab$

$q_1 a q_2 ab.$

machine halts at q_2 state and not reach q_3

$\therefore$ not accepted.

bbaa.

Initially at B → move right.

$q_0 b b a a$

$q_0\ q_1\ q_2\ a b a a$.

$q_2\ a\ q_3\ b a a$.

∴ it is accepted as it reaches q_3 (final state)

(C) The language recognized by M is $L = \{(a^n b^m)^* \mid n \geq 1, m \geq 1\}$

i.e. any number of a's & bs together.

$L = \{ab, aabb, bbaa, aaabbb, \ldots\}$