Q1 A finite automata can be used only to accept regular languages whereas pushdown automata is a finite automatce 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. $ 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

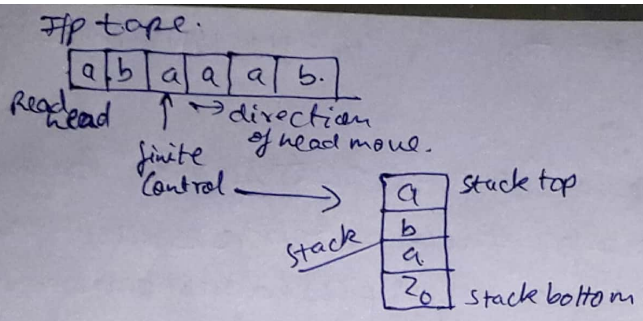
* PDA can be defined as a collection of 7 components:

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

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

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

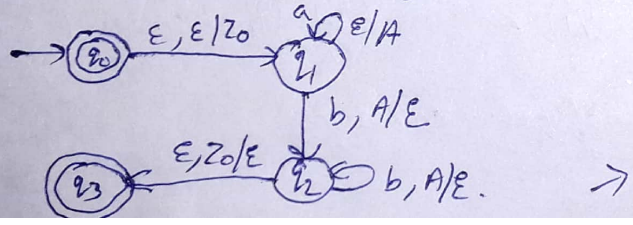
$\delta(q_0, \varepsilon, \varepsilon) = (q_1, Z_0)$

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

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

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

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



→

for string aabb

$(q_0, a, \varepsilon) \to (q_1, AZ_0)$

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

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

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

$(q_2, \varepsilon, Z_0) \to (q_3, \varepsilon)$

PDA halts & accepts it.

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

Now

$S \to aSa \Rightarrow aa$ (↳ $S \to \varepsilon$)

$S \to aSa \Rightarrow aaa$ (↳ $S \to a$)

$S \to aSa \Rightarrow aba$ (↳ $S \to b$)

$S \to bSb \Rightarrow bb$ (↳ $S \to \varepsilon$)

$S \to bSb \Rightarrow bab$ (↳ $S \to a$)

$S \to bSb \Rightarrow bbb$ (↳ $S \to b$)

$S \to a$.

$S \to b$

$S \to \varepsilon$.

∴ 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) $S \to bb\,A$
$A \to Bb$
$B \to aAa \mid \varepsilon$.

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

$S \to bb\,A \Rightarrow \underline{bbb}$
* ↳ $A \to Bb$.
↳ $B \to \varepsilon$

* ↳ $A \to Bb$. $\Rightarrow abab \Rightarrow \underline{bbabab}$.
↳ $B \to aAa$. $\Rightarrow aba$.
↳ $A \to Bb$. $\Rightarrow b$
↳ $B \to \varepsilon$

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

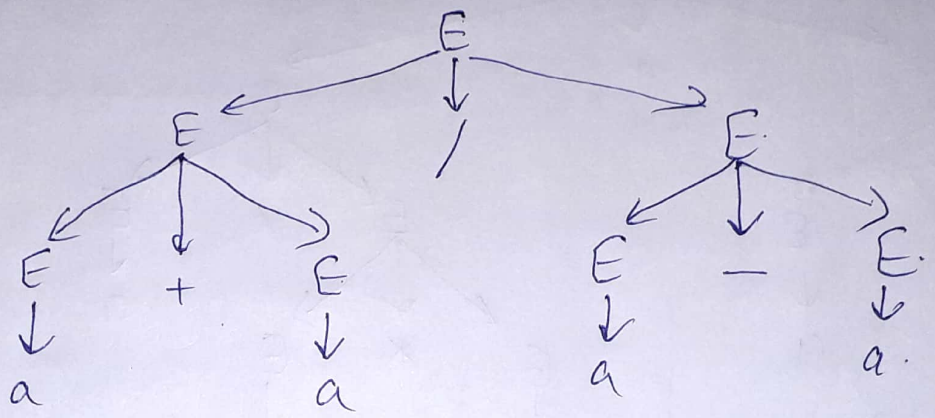
bb aabaab.
$S \to bbA$ (1) $\Rightarrow S \to bbBb$ (2) $\Rightarrow S \to bbaAab$ (3)
↳ $A \to Bb$ ↳ $B \to aAa$ ↳ $A \to Bb$

$\Rightarrow S \to bbaBbab$ (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 bb aabaab is not generated by this grammar.

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

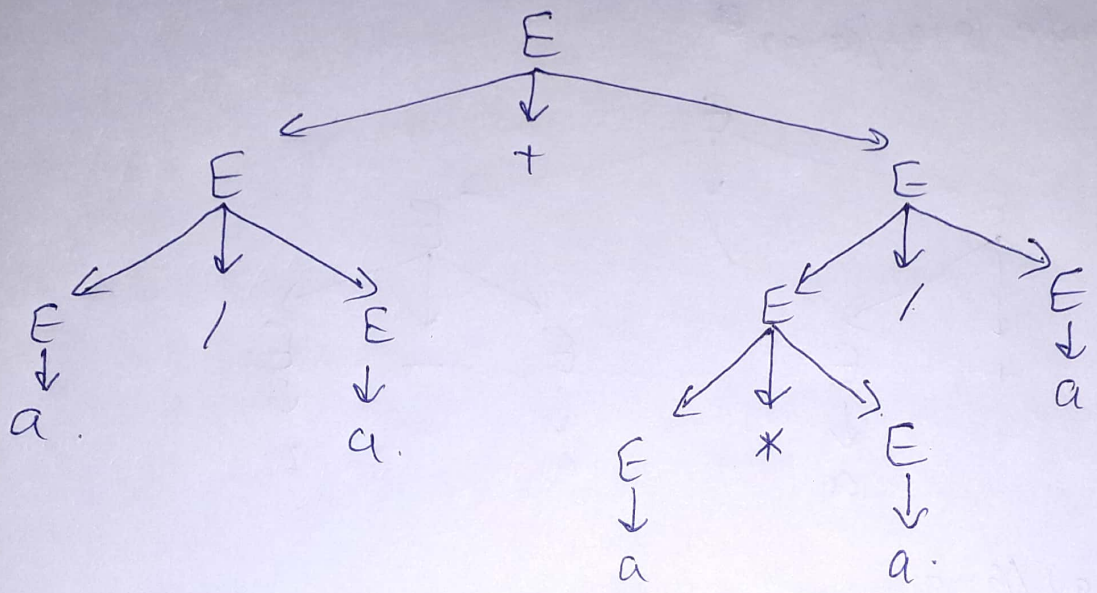
(a) generate (a+a)/(a-a)



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

(b)

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



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

Q4

Given:

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

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

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

(A change has been made here →

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

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

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

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

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

(a) for string aba.

a b a

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

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

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

for string abb.

a b b

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

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

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

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

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

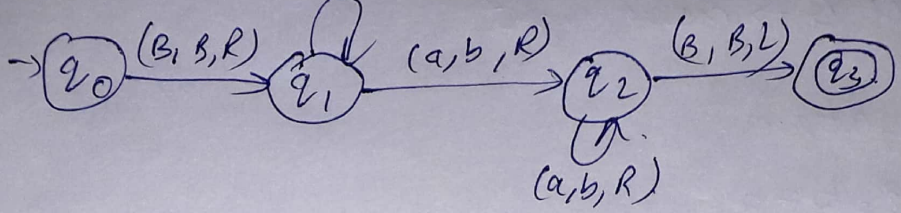
$\delta(q_f, a, a) = (q_f, \varepsilon)$

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

It is acceptable.

Hence string baa & baaaa are acceptable by given PDA.

Q5 (i)



(ii). aabb →

$\delta(q_0, B) \rightarrow \delta(q_1, B, R)$

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

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

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

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

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

as it reach final state so accepted.

abab

Initial → q_0 abab → move right.

q_1 abab

q_1 a q_2 ab

q_1 a q_2 ab.

machine halts at q_2 state and not reach q_3

∴ not accepted.

bbaa.

Initially at B → move right

$q_0 b b a a$

$q_0 a q_1 b a a$

$q_2 a q_3 a a$

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

(c) The language recognized by M is $L = \{(a^n b^m)^* \mid n \geq 2, m \geq 2\}$

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

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