UNIT - IV

Push Down Automata.

* Let us consider $L=\{a^n b^n / n \geq 1\}$. This is a context free language but not Regular.
* A Finite Automata cannot Accept 'L' i.e, strings of the form $a^n b^n$ as it has to remember the no. of a's in a string

2/18 * $(q_0, x, z_0) \vdash (q_f, \epsilon, \gamma)$ for Final state

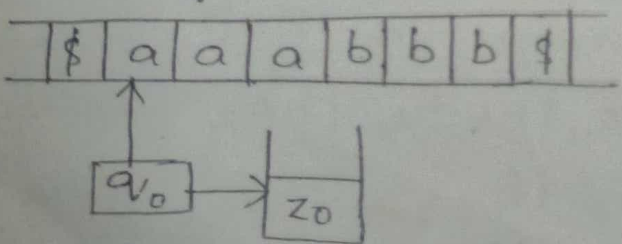
* $(q_0, x, z_0) \vdash (q, \epsilon, \epsilon)$ for empty stack.

1/18 Design a PDA which accepts $L=\{a^n b^n / n \geq 1\}$.

$L=\{ab, aabb, aaabbb, \text{---------}\}$.

$w = aaabbb$.

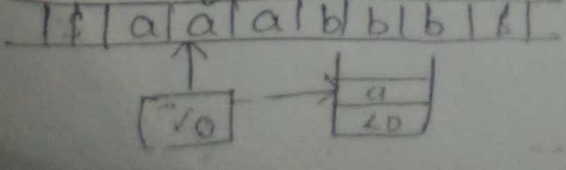
1) Reading of 1^{st} a.



push a into stack.

$\delta(q_0, a, z_0) = (q_0, az_0)$

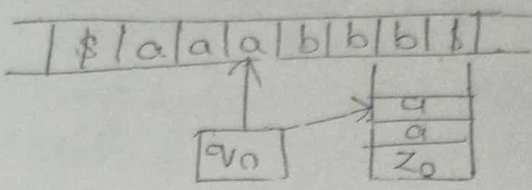
2) Reading of 2^{nd} a



push a into stack.

$\delta(q_0, a, a) = (q_0, aZ_0)$.

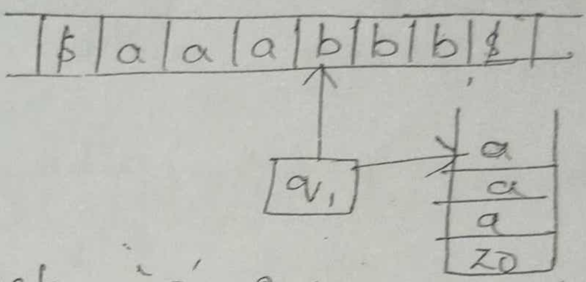
3) Reading of 3rd a.



push 'a' into sta

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

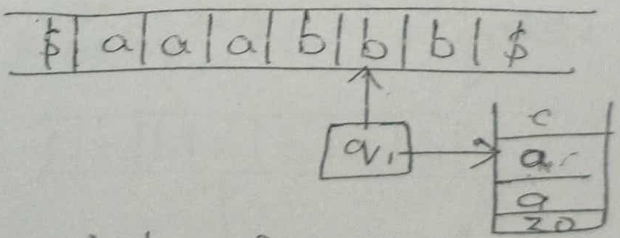
4) Reading of ~~1st~~ b



~~push~~ pop 'a' ~~into~~ from stack.

$\delta(q_0, b, a) = (q_1, \epsilon)$

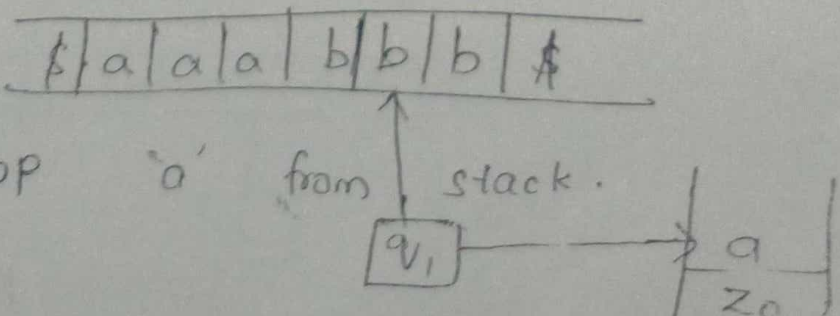
5) Reading of 2nd b.



Pop 'a' from stack

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

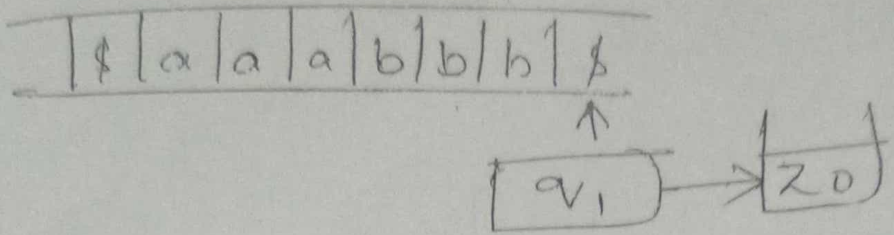
6) Reading of 3rd b.



Pop 'a' from stack.

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

7) ~~Read~~:



$\delta(q_1, \epsilon, Z_0) =$ ~~Sto~~ $(q_f, Z_0) \rightarrow$ for final state

$\delta(q_1, \epsilon, Z_0) = (q_2, \epsilon) \rightarrow$ for empty stack.

String Acceptability:-

Consider the string aabb.

$(q_0, aabb, Z_0) \vdash (q_0, abb\ aZ_0) \vdash$

$\delta(q_0, a, Z_0) = (q_0, aZ_0)$ $\quad\quad$ $\delta(q_0, a, a) = (q_0, aa)$

$(q_0, bb, aaZ_0) \vdash (q_1, b, aZ_0) \vdash (q_1, \epsilon, Z_0)$

$\delta(q_0, b, a) = (q_1, \epsilon)$ $\quad\quad$ $\delta(q_1, b, a) = (q_1, \epsilon)$

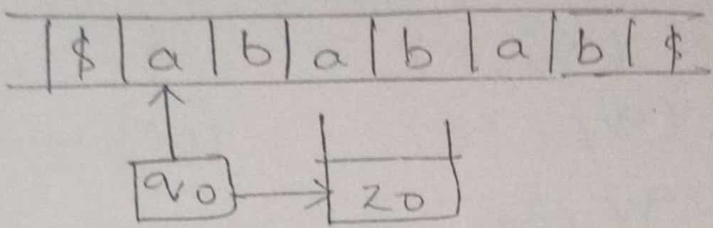
$\vdash (q_2, \epsilon, \epsilon)$.

Hence the string is Acceptable.

x Design a PDA for Equal no. of a's and Equal no. of b's.

$L = \{ababab\}$.

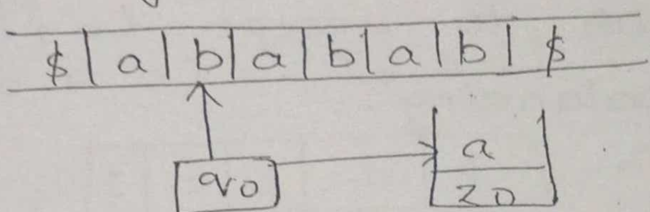
1) Reading of 1^{st} a.



push a into stack.

$\delta(q_0, a, z_0) = (q_0, az_0)$

2) Reading of 1^{st} b.

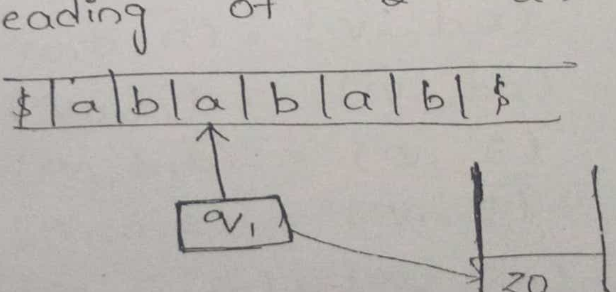


$z_0)$

Pop a from stack.

$\delta(q_0, b, a) = (q_1, z_0)$

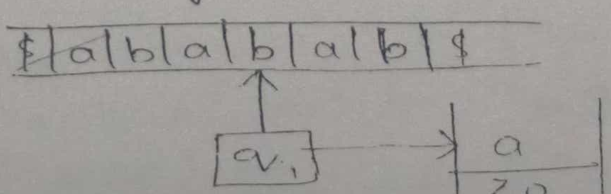
3) Reading of 2^{nd} a.



$\delta(q_1, a, z_0) = (q_2, az_0)$

4) Reading of 2^{nd} b.

$\delta(q_2, b, a) \cdot (q_3,$



1) Design a PDA to accept $L = \{()()()\}$ Paranthesis balancing.

$\delta(q_0, (, z_0) = (q_0, (z_0)$

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

$\delta(q_1, (, z_0) = (q_1, (z_0)$

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

$\delta(q_0, \epsilon, z_0) = (q_f, z_0)$

$\delta(q_1, \epsilon, z_0) = (q_f, z_0)$

$(q_0, ()(), z_0) \vdash (q_0,)(), (z_0)$

$\vdash (q_1, (), z_0) \vdash (q_1,), (z_0) \vdash (q_1, \epsilon, z_0)$

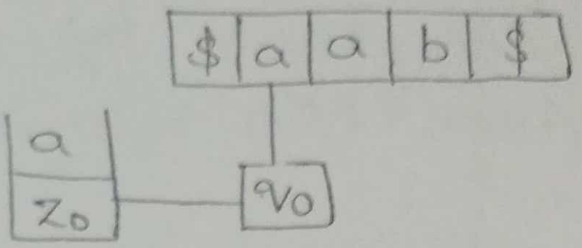
$\vdash (q_f, \epsilon, z_0)$.

2) Design a PDA to accept $L = \{((()))\}$ Paranthesis balancing.

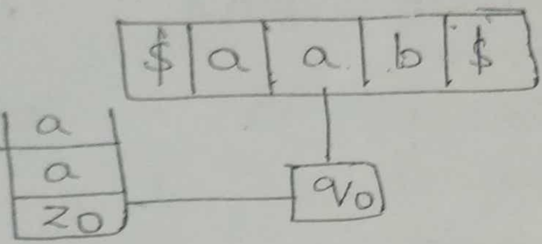
* Design a PDA to accept $L = \{wcw^R\}$ (palindrome) :·

Sol: consider $w = aab$

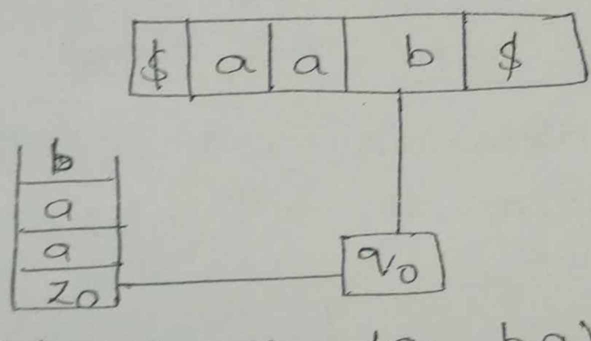
$w^R = baa$



$\delta(q_0, a, Z_0) = (q_0, aZ_0)$



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



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

$\delta(q_0, c, b) = (q_1, ba)$

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

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

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

check The acceptability of string aabcbac

$(q_0, aabcbaa, Z_0) \vdash (q_0, abcbaa, aZ_0)$
$\vdash (q_0, bcbaa, aa) \vdash (q_0, cbaa, ba)$
$\vdash (q_0, baa, ba) \vdash (q_1, aa, \epsilon) \vdash$
$(q_1, a, \epsilon) \vdash (q_f, \epsilon, Z_0)$.

3.16 **Equivalence between PDA and CFG:**

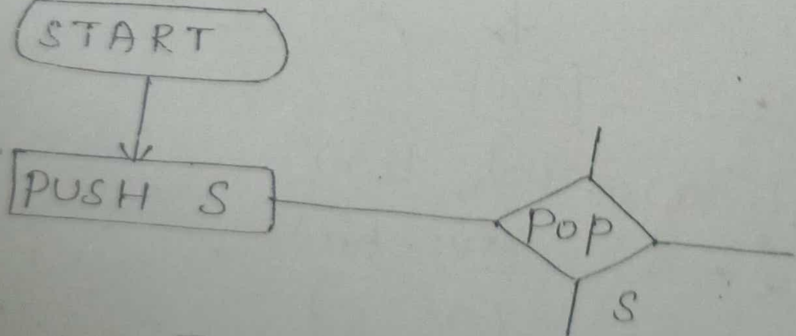
There is an Equivalence between PDA and CFG i.e, if PDA is given we construct equivalence CFG and if CFG is given Then we construct Equivalent PDA. CFG is not directly acceptable by PDA, So we convert The given CFG into either CNF or GNF.

* CFG to PDA < CNF - PDA, GNF - PDA
* PDA to CFG.

CNF to PDA:-

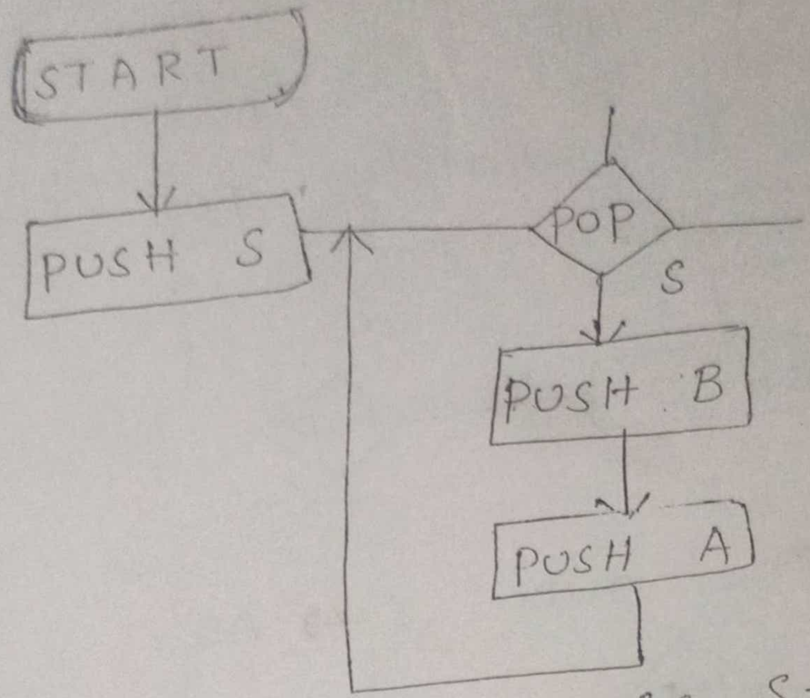
Step-1:- construct CNF

Step-2:- Starts with The start symbol and push This start symbol onto The stack. Pop The Start symbol S.

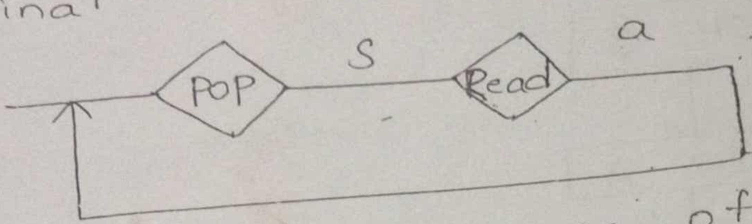


Step-3:- At The time of popping The start symbol consider The rule for 'S'.

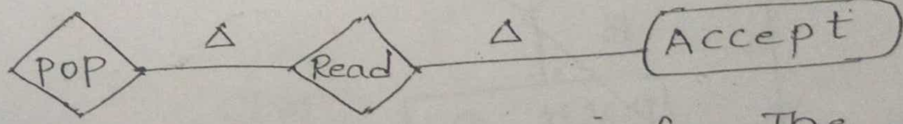
1) If The rule is $S \rightarrow AB$ Then push The Symbols in reverse order.



Step-4:- If The rule is $S \rightarrow a$ Then Th terminal is read.



Step-5:- For Acceptance of The string the Δ acceptability is used. The Δ acceptability is define as follows.



Step-6:- construct The transitions for The above design.

1) construct PDA for The following CFG

$S \rightarrow AB$

$B \rightarrow CD$

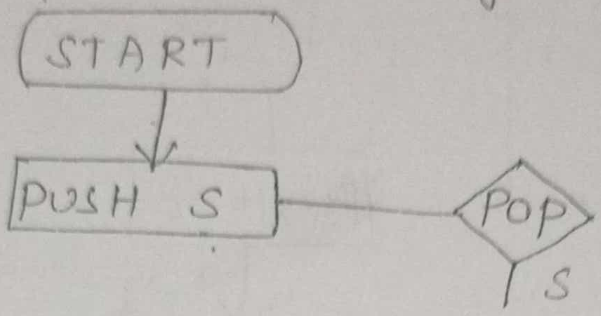
$A \rightarrow a$

$C \rightarrow b$

$D \rightarrow a$

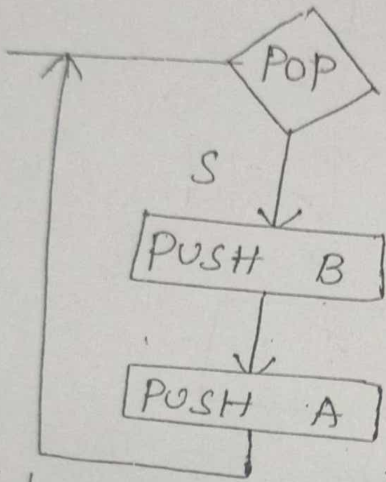
Step-1:- The given grammar is already in CNF.

Step-2:- Push The start symbol S onto The stack and immediately POP S.

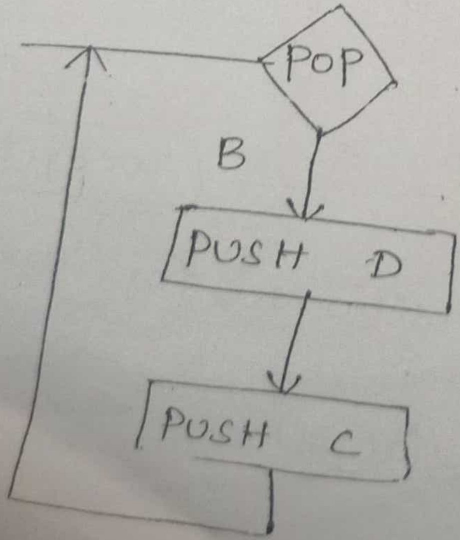


Step-3:-

*consider The rule $S \rightarrow AB$

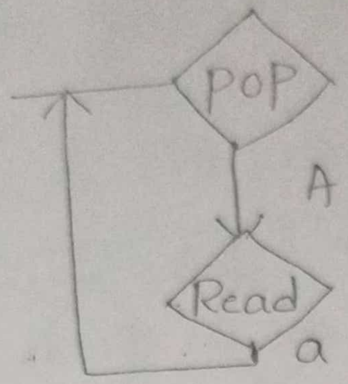


*consider The rule $B \rightarrow CD$

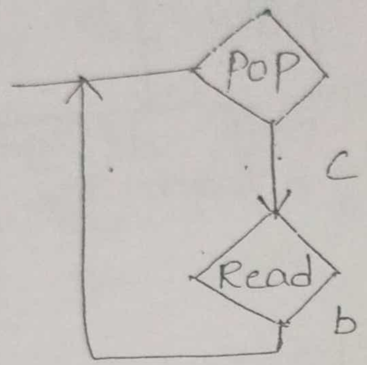


Step-4:-

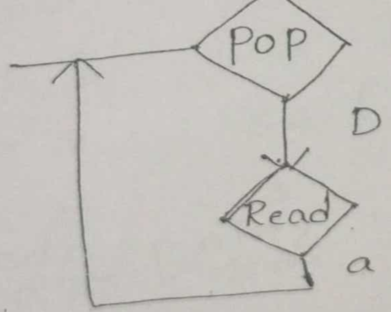
*consider The rule $A \rightarrow a$.



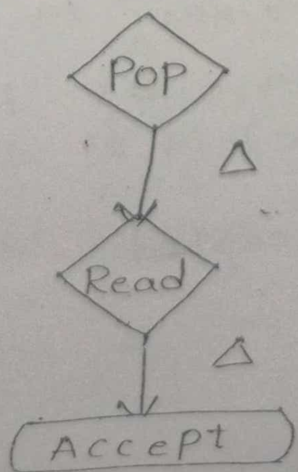
* consider The rule $C \longrightarrow b$:



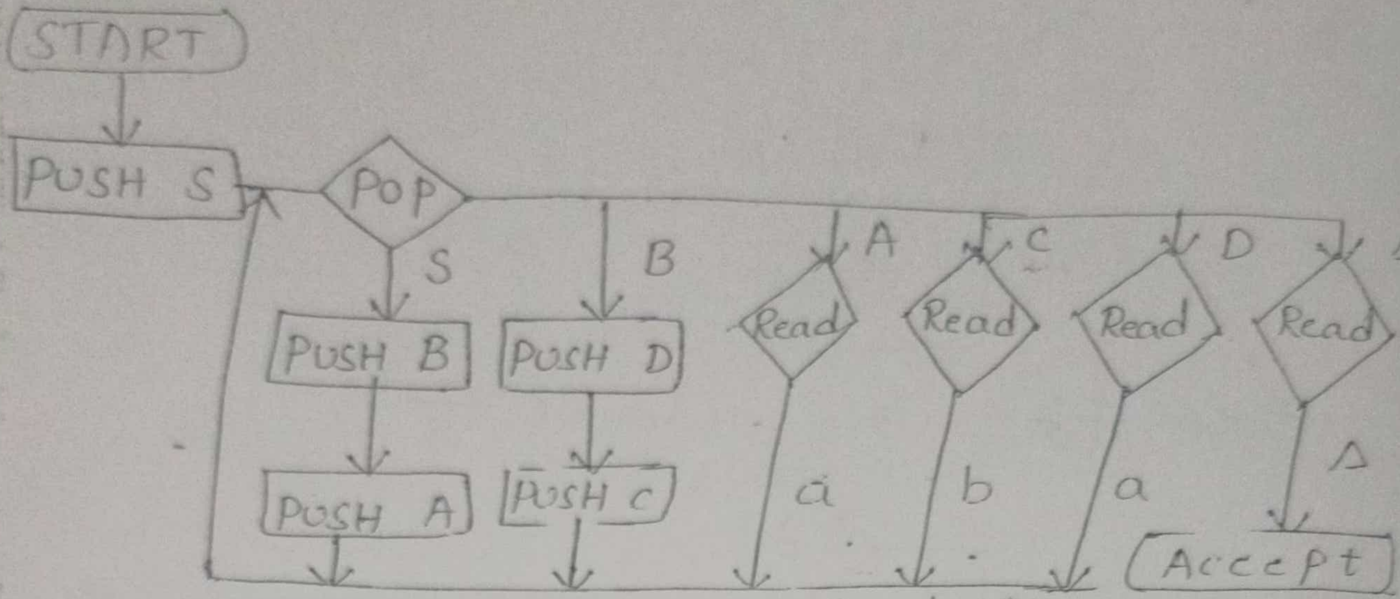
* consider The rule $D \longrightarrow a$



Step-5:



∴ The overall design is



PDA design.

The generated string from the grammar is

$S \rightarrow AB$

$S \rightarrow aB$ [∵ $A \rightarrow a$]

$S \rightarrow aCD$ [∵ $B \rightarrow CD$].

$S \rightarrow abD$ [∵ $C \rightarrow b$]

$S \rightarrow aba$ [∵ $D \rightarrow a$]

∴ The string is aba.

Acceptability of The string aba:-

$\omega = aba$.

$(q, aba, S) \vdash (q, aba, AB) \vdash (q, ba, B)$

$\delta(q, aba, s) = (q, AB)$. $(q, a, A) = (q, \epsilon)$

$\vdash (q, ba, CD) \vdash (q, a, D) \vdash (q, \epsilon, \epsilon)$

$\delta(q, \text{[illegible]}, B) = (q, CD)$ $\delta(q, b, c) = (q, \epsilon)$ $\delta(q, a, D) = (q, \epsilon)$

∴ The string is accepted.

3/18 PDA design:-

* If there is a rule $S \rightarrow AB$, Then PDA is $\delta(q, \omega, S) = (q, AB)$
* If there is a rule $S \rightarrow a$, Then PDA is $\delta(q, a, S) = (q, \epsilon)$.
* $S \rightarrow AB$. The PDA is $\delta(q, \omega, S) = (q, AB)$
* $B \rightarrow CD$ The PDA is $\delta(q, \omega, B) = (q, CD)$
* $C \rightarrow b$ The PDA is $\delta(q, b, C) = (q, \epsilon)$
* $D \rightarrow a$ The PDA is $\delta(q, a, D) = (q, \epsilon)$
* $A \rightarrow a$ The PDA is $\delta(q, a, A) = (q, \epsilon)$.

Stack Acceptability:-

Input	Stack	Action
		start
aba Δ	Δ	push S
aba Δ	S Δ	pop S
aba Δ	Δ	push B
aba Δ	B Δ	push A
aba Δ	AB Δ	pop A
	B Δ	pop B.
ba Δ	Δ	push D
ba Δ	D Δ	push C
ba Δ	CD Δ	pop C
ba Δ	D Δ	pop D
a Δ	Δ	pop Δ
Δ	–	Accept.
–		

B)
, ϵ)
ϵ)
(q, ϵ).

Convert The following CFG into PDA

$$S \longrightarrow \overset{A}{a}\underbrace{S\overset{B}{b}}_{C}$$

$$S \longrightarrow \underset{A}{a}\underset{B}{b}.$$

Given grammar is not in CNF. so,
convert it into CNF.

$S \longrightarrow AC$

$S \longrightarrow AB$

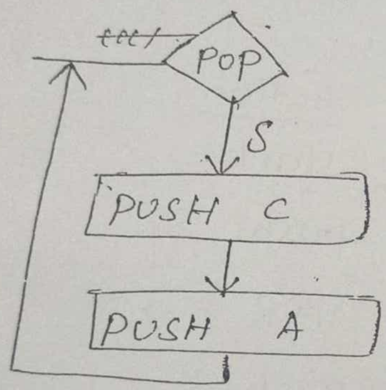
$C \longrightarrow SB$

$A \rightarrow a.$

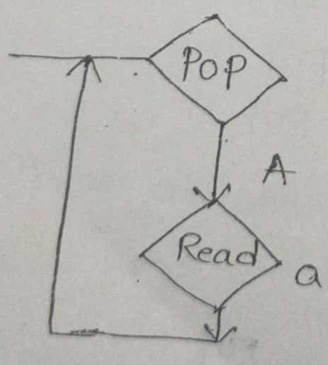
$B \rightarrow b.$

Now, The grammar is in CNF.

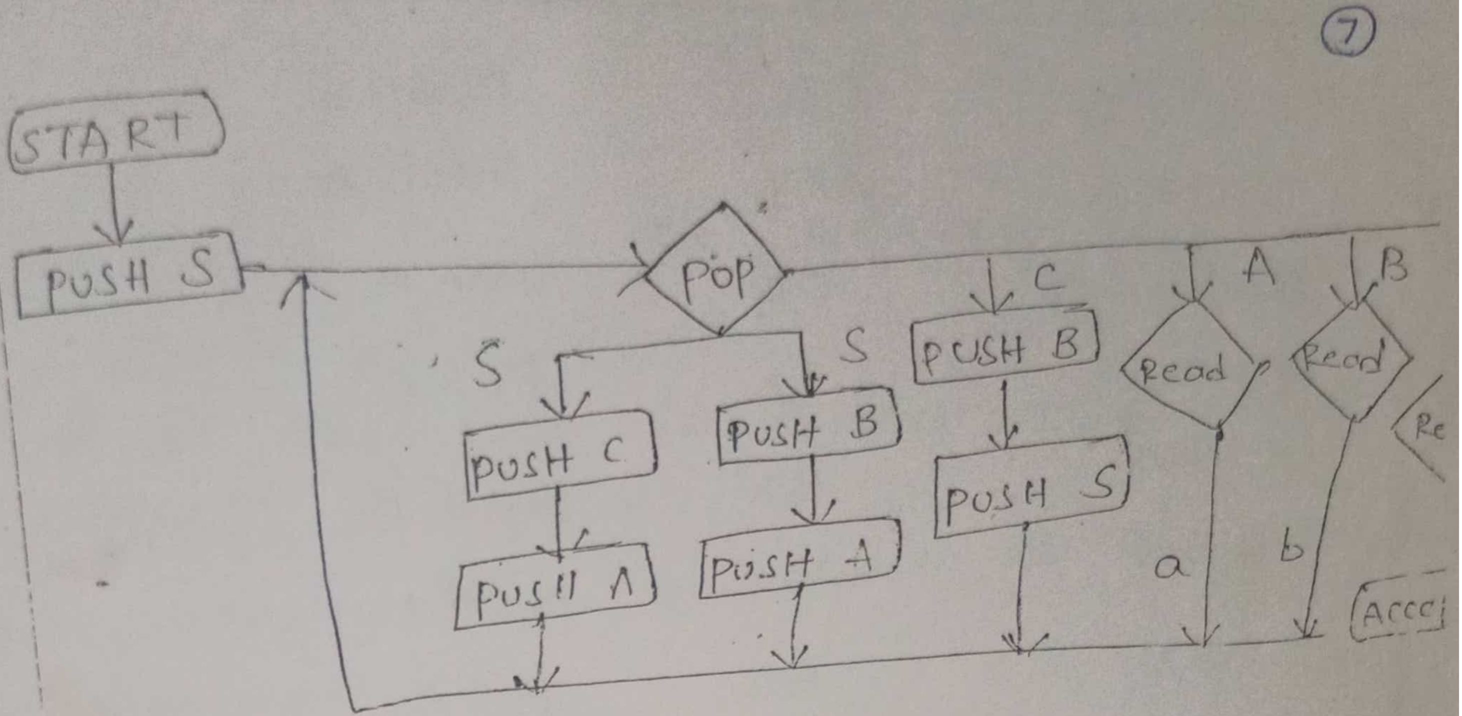
$S \rightarrow AC.$



$A \rightarrow a.$



The overall design is



* PDA design:-

* $S \to AC$ The PDA is $\delta(q, \omega, S) = (q, AC)$
* $S \to AB$, The PDA is $\delta(q, \omega, S) = (q, AB)$
* $C \to SB$, The PDA is $\delta(q, \omega, S) = (q, SB)$
* $A \to a$, The PDA is $\delta(q, a, A) = (q, \epsilon)$
* $B \to b$, The PDA is $\delta(q, b, B) = (q, \epsilon)$.

String generation:- $S \to AC$

$S \to aSb$

$S \to ab$

$S \to$ ~~ab~~ aabb.

∴ The string is aabb.

String Acceptability:-

consider the string $\omega = aabb$.

$(q, \cancel{\omega}\ aabb, S) \vdash (q, \cancel{\omega}\ aabb, AC) \vdash (q, abb, C) \vdash$
$(q, abb, \cancel{A}B B) \vdash (q, bb, BB) \vdash (q, bb, B) \vdash$
(q, ϵ, ϵ).
$(q, b, B) \vdash (q, \epsilon, \epsilon)$.

stack Acceptability:-

Input	Stack	Action
	Δ	start
aabbΔ		push S
	SΔ	
aabbΔ		

aabbΔ	Δ	pop S
aabbΔ	CΔ	push C
aabbΔ	ACΔ	push A
aabbΔ	CΔ	POP A
aabbΔ	Δ	POP C
abbΔ	BΔ	PUSH B
abbΔ	ABΔ	PUSH A
bbΔ	BΔ	POP A

$S \to AC$

$\to aSB$

$\to aABB$

$\to aaBB$

$\to aabB$

$\to aabb$

2/3/18 **GNF to PDA:-** If The Grammar is in The form of

* If The rule is $A \to a\alpha$ Then PDA is $\delta(q, a, A) = (q, \alpha)$.
* If The rule is $A \to a$ Then PDA is $\delta(q, a, A) = (q, \epsilon)$.

* construct a PDA Equivalent to The following Grammar:-

$S \to aAA,$
$A \to aS \mid bS \mid a.$

Sol:- * The Grammar is already in GNF
* consider The rule $S \to aAA$ is in The form of $A \to a\alpha$ Then The PDA is
$\delta(q, a, S) = (q, AA)$.
* $A \to aS$, It is in The form of $A \to a\alpha$ Then The PDA is $\delta(q, a, A) = (q, S)$
* $A \to bS$, It is in The form of $A \to a\alpha$ Then The PDA is $\delta(q, b, A) = (q, S)$
* $A \to a$, It is in The form of $A \to a$ Then The PDA is $\delta(q, a, A) = (q, \epsilon)$

~~CFG to~~ **PDA to CFG:-**
Suppose M is PDA, Then There is a Grammar G such That $L(G) = L(M)$.
* To convert The PDA to CFG, we use The following Three rules.

Rule 1:- The productions for start symbol S are given by

$S \rightarrow [q_0, z_0, q]$ for each state q in Q

Rule 2:- Each move That pops a symbol from stack with transitions as

$\delta(q, a, z_i) = (q', \epsilon)$ induces a production as $[q, z_i, q'] \rightarrow a$ for q' in Q.

Rule 3:- Each move That does not pop symbol from stack with transition as $\delta(q, a, z_0) = (q', z_1 z_2 z_3 \text{------})$ induces a production as $[q, z_0, q_m] \rightarrow a[q_0', z_1, q_2][q_2, z_2, q_3][q_3, z_3, q_4] \text{-----} [q_{m-1}, z_m, q_m]$.

Note:- This procedure is applicable for empty stack approach. 23/3

* Give The Equivalent CFG for The following PDA

$\delta(q_0, b, z_0) = (q_0, zz_0)$

$\delta(q_0, \epsilon, z_0) = (q_0, \epsilon)$

$\delta(q_0, b, z) = (q_0, zz)$

$\delta(q_0, a, z) = (q_1, z)$

$\delta(q_1, b, z) = (q_1, \epsilon)$

$\delta(q_1, a, z_0) = (q_0, z_0)$

Here There are two states q_0 and q_1, two stack symbols z and z_0 and two input symbols a and b.

* The productions for start symbol S is

$S \longrightarrow [q_0, z_0, q_0]$

$S \longrightarrow [q_0, z_0, q_1]$ $\quad S \to [q_0, z_0, q_2]$

* consider $\delta(q_0, \epsilon, z_0) = (q_0, \epsilon)$

According to rule-2

$\delta(q_0, \epsilon, z_0) = (q_0, \epsilon)$

$\delta(q, a, z_i) = (q', \epsilon)$ Then $[q, z_i, q'] \to a$

$$\boxed{[q_0, z_0, q_0] \longrightarrow \epsilon}$$

* consider $\delta(q_1, b, z) = (q_1, \epsilon)$ (from rule-2

$\delta(q, a, z_i) = (q', \epsilon)$ Then $[q, z_i, q'] \to$

$[q_1, z, q_1] \longrightarrow b.$

23/3/18

According to rule -3.

$\delta(q, a, z_0) = (q', z_1 z_2 \text{-------})$

$[q, z_0, q_m] \longrightarrow a[q', z_1, q_2][q_2, z_2, q_3]$

$[q_3, z_3, q_4] \text{------} [q_{m-1}, z_m\, q_m].$

1) $\delta(q_0, b, z_0) = (q_0, zz_0)$ apply rule 3.

no. of states $= 2 = \{q_0, q_1\}$ (3)

no. of stack symbols $= 2$ $\{z, z_0\}$ (3)

no. of rules derived are $= 2 \times 2 = 4$

so, The no. of rules $2 \times 3 = 6.$

1) $\delta(q_0, b, z_0) = (q_0, zz_0)$

$[q_0, z_0, q_0] \to b[q_0, z, q_0][q_0, z_0, q_0]$

$[q_0, z_0, q_0] \to b[q_0, z, q_1][q_1, z_0, q_0]$

$[q_0, z_0, q_1] \to b[q_0, z, q_0][q_0, z_0, q_1]$

$[q_0, z_0, q_1] \to b[q_0, z, q_1][q_1, z_0, q_1]$

$q_0 \quad z_0 \quad q_2 \to b$

$q_0 \quad z_0 \quad q_2 \to z_0)$

3) $\delta(q_0, b, z) = (q_0, z z)$ apply rule

$[q_0, z, q_0] \rightarrow b\ [q_0, z, q_0]\ [q_0, z, q_0]$

$[q_0, z, q_0] \rightarrow b\ [q_0, z, q_1]\ [q_1, z, q_0]$

$[q_0, z, q_1] \rightarrow b\ [q_0, z, q_0]\ [q_0, z, q_1]$

$[q_0, z, q_1] \rightarrow b\ [q_0, z, q_1]\ [q_1, z, q_1]$

4) $\delta(q_0, a, z) = (q_1, z)$

$[q_0, z, q_0] \rightarrow a\ [q_1, z, q_0]$

$[q_0, z, q_1] \rightarrow a\ [q_1, z, q_1]$

Sol:-

6) $\delta(q_1, a, z_0) = (q_0, z_0)$

$[q_1, z_0, q_0] \rightarrow a\ [q_0, z_0, q_0]$

$[q_1, z_0, q_1] \rightarrow a\ [q_0, z_0, q_1]$.

→ Rename as

$[q_0, z_0, q_0]$ as A

$[q_0, z_0, q_1]$ as B

$[q_1, z, q_1]$ as C

$[q_1, z_0, q_0]$ as D

$[q_1, z_0, q_1]$ as E

$[q_0, z, q_0]$ as F

$[q_0, z, q_1]$ as G

$[q_1, z, q_0]$ as H

The Grammar is

$S \rightarrow A | B | C$

$A \rightarrow bFA | bGD$

$B \rightarrow bAB | bGE$

$D \rightarrow aA$

$E \rightarrow aB$

$F \rightarrow bFF | bGH | aH$

$G \rightarrow bAB | bGC | aC$

* construct CFG from the following PDA

1) $\delta(q_0, \epsilon, z_0) = (q_1, \epsilon)$ r-2
2) $\delta(q_0, 0, z_0) = (q_0, 0z_0)$
3) $\delta(q_0, 0, 0) = (q_0, 00)$
4) $\delta(q_0, 1, 0) = (q_0, 10)$
5) $\delta(q_0, 1, 1) = (q_0, 11)$
6) $\delta(q_0, 0, 1) = (q_1, \epsilon)$ r-2
7) $\delta(q_1, 0, 1) = (q_1, \epsilon)$ r-2
8) $\delta(q_1, 0, 0) = (q_1, \epsilon)$ r-2
9) $\delta(q_1, \epsilon, z_0) = (q_1, \epsilon)$ r-2

Sol:- Step-1:- The rule for start symbol's is

$$S \rightarrow [q_0, z_0, q_0] \mid [q_0, z_0, q_1]$$

rule-2:-

1) consider $\delta(q_0, \epsilon, z_0) = (q_1, \epsilon)$

$$[q_0, z_0, q_1] \rightarrow \epsilon$$

6) $\delta(q_0, 0, 1) = (q_1, \epsilon)$

$$[q_0, 1, q_1] \rightarrow 0$$

7) $\delta(q_1, 0, 1) = (q_1, \epsilon)$

$$[q_1, 1, q_1] \rightarrow 0$$

8) $\delta(q_1, 0, 0) = (q_1, \epsilon)$

$$[q_1, 0, q_1] \rightarrow 0$$

9) $\delta(q_1, \epsilon, z_0) = (q_1, \epsilon)$

$$[q_1, z_0, q_1] \rightarrow \epsilon$$

rule-3:-

2) $\delta(q_0, 0, z_0) = (q_0, 0z_0)$

$\delta(q_0, 0, z_0) = (q_0, 0z_0)$

$$[q_0, z_0, q_0] \rightarrow 0\,[q_0, 0, q_0]\,[q_0, z_0, q_0]$$

$[q_0, z_0, q_0] \rightarrow 0\ [q_0, 0, q_1]\ [q_1, z_0, q_0]$

$[q_0, z_0, q_1] \rightarrow 0\ [q_0, 0, q_0]\ [q_0, z_0, q_1]$

$[q_0, z_0, q_1] \rightarrow 0\ [q_0, 0, q_1]\ [q_1, z_0, q_1]$

3) $\delta(q_0, 0, 0) = (q_0, 00)$

$[q_0, 0, q_0] \rightarrow 0\ [q_0, 0, q_0]\ [q_0, 0, q_0]$

$[q_0, 0, q_0] \rightarrow 0\ [q_0, 0, q_1]\ [q_1, 0, q_0]$

$[q_0, 0, q_1] \rightarrow 0\ [q_0, 0, q_0]\ [q_0, 0, q_1]$

$[q_0, 0, q_1] \rightarrow 0\ [q_0, 0, q_1]\ [q_1, 0, q_1]$

4) $\delta(q_0, 1, 0) = (q_0, 10)$

$[q_0, 0, q_0] \rightarrow 1\ [q_0, 1, q_0]\ [q_0, 0, q_0]$

$[q_0, 0, q_0] \rightarrow 1\ [q_0, 1, q_1]\ [q_1, 0, q_0]$

$[q_0, 0, q_1] \rightarrow 1\ [q_0, 1, q_0]\ [q_0, 0, q_1]$

$[q_0, 0, q_1] \rightarrow 1\ [q_0, 1, q_1]\ [q_1, 0, q_1]$

5) $\delta(q_0, 1, 1) = (q_0, 11)$

$[q_0, 1, q_0] \rightarrow 1\ [q_0, 1, q_0]\ [q_0, 1, q_0]$

$[q_0, 1, q_0] \rightarrow 1\ [q_0, 1, q_1]\ [q_1, 1, q_0]$

$[q_0, 1, q_1] \rightarrow 1\ [q_0, 1, q_0]\ [q_0, 1, q_1]$

$[q_0, 1, q_1] \rightarrow 1\ [q_0, 1, q_1]\ [q_1, 1, q_1]$

Rename,

$[q_0, z, q_0]$ as A

$[q_0, z_0, q_1]$ as B

$[q_0, 0, q_0]$ as C

$[q_0, 0, q_1]$ as D

$[q_0, 1, q_0]$ as E

$[q_0, 1, q_1]$ as F

$[q_0, z_0, q_1]$ as G

$[q_0, 1, q_0]$ as H

$[q_1, 1, q_1]$ as I

$[q_1, 0, q_1]$ as J

$[q_1, z_0, q_1]$ as K

The Grammar is

$S \rightarrow A|B$

$A \rightarrow 0|$

5. Turing Machines

* Alan Turing is father of such a Model which has computing capability of general purpose computer. Hence This Model is popularly known as Turing Machine. It has The following features.
* It has external Memory which remembers Arbitarily long sequence of Input.
* It has unlimited Memory capability.
* The Model has a facility by which Input at left (or) right on The can be read easily
* The Machine can produce certain output based on Input. Some times It may be required That The Same input has to be used to generate The output. So, In This Machine The distinction between Input and output has been removed Thus, a common set of Alphabets can be used for The Turing Machine.

Basic Model (or) Model of Turing Machine:-

The turing Machine can be modled with The help of The following representation.

Input tape:- The Input tape having infinite, number of cells, Each cell