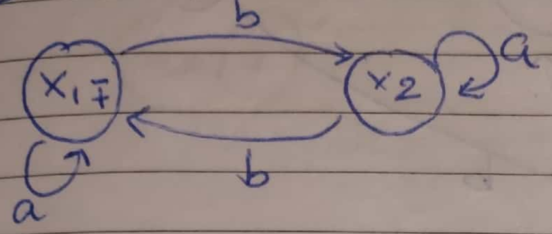
Assn2

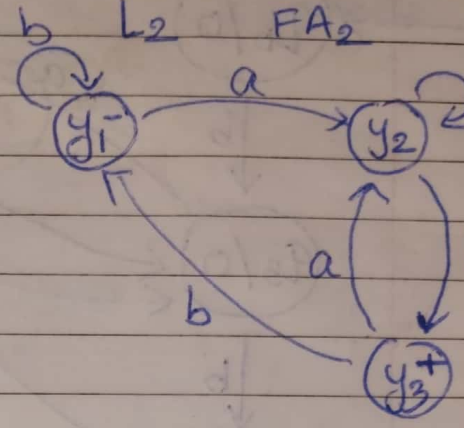
Q1) L1: language of all words with an even number of b's
L2: language of all words that end with substring ab.

$\Sigma = \{a, b\}$

(a) L1 FA1

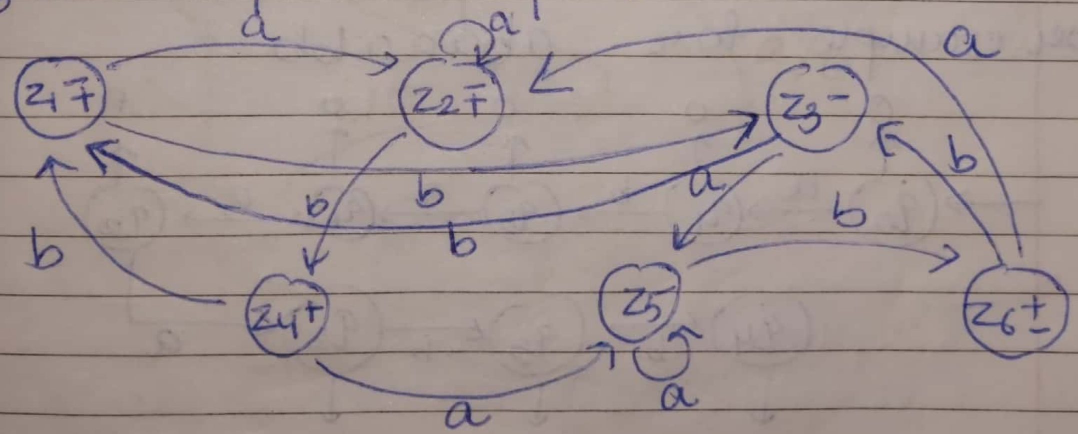


L2 FA2



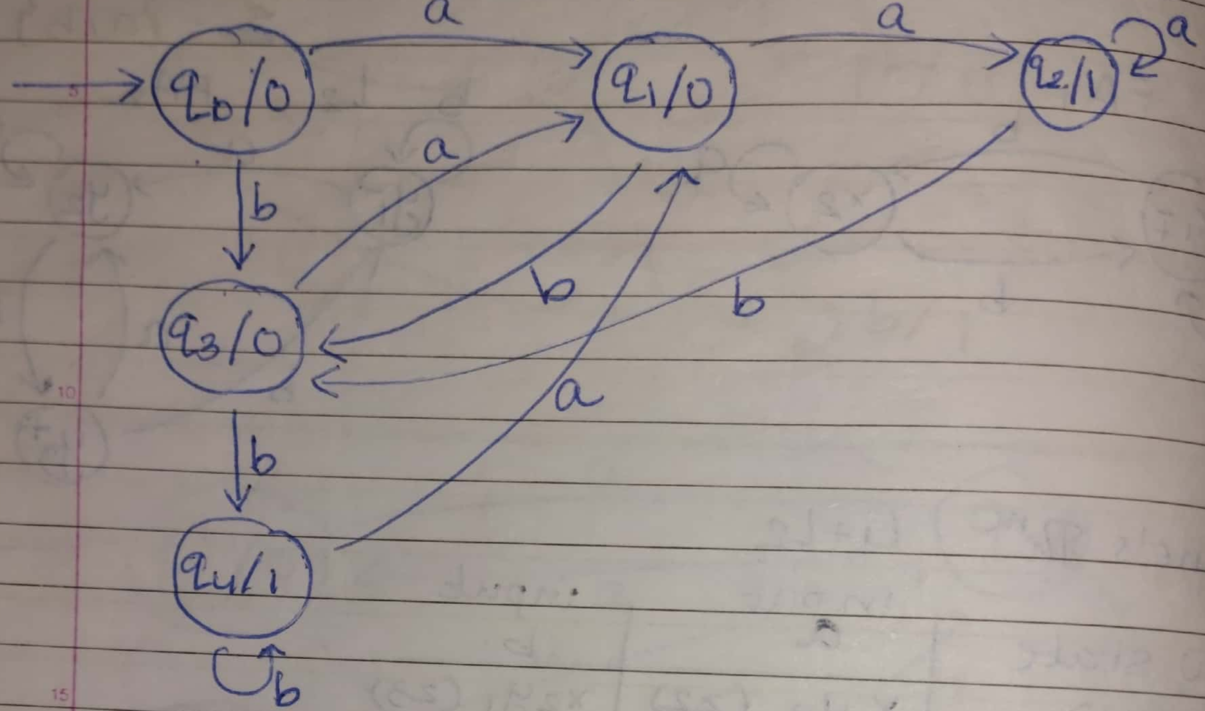
(b) Kleene's Th^m L1+L2

	new state	input a	input b
+-	x_1y_1 (z1)	x_1y_2 (z2)	x_2y_1 (z3)
+-	x_1y_2 (z2)	x_1y_2 (z2)	x_2y_3 (z4)
-	x_2y_1 (z3)	x_2y_2 (z5)	x_1y_1 (z1)
+	x_2y_3 (z4)	x_2y_2 (z5)	x_1y_1 (z1)
	x_2y_2 (z5)	x_2y_2 (z5)	x_1y_3 (z6)
++-	x_1y_3 (z6)	x_1y_2 (z2)	x_2y_1 (z3)



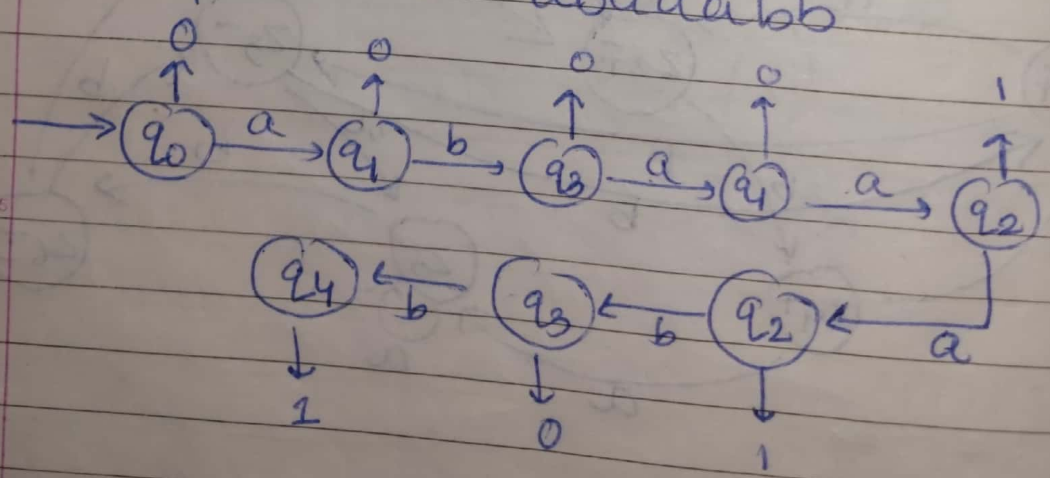
L1 + L2

Q2. Draw a Moore Machine to count the occurrences double letters (i.e. aa or bb) in words over the alphabet $\Sigma = \{a, b\}$



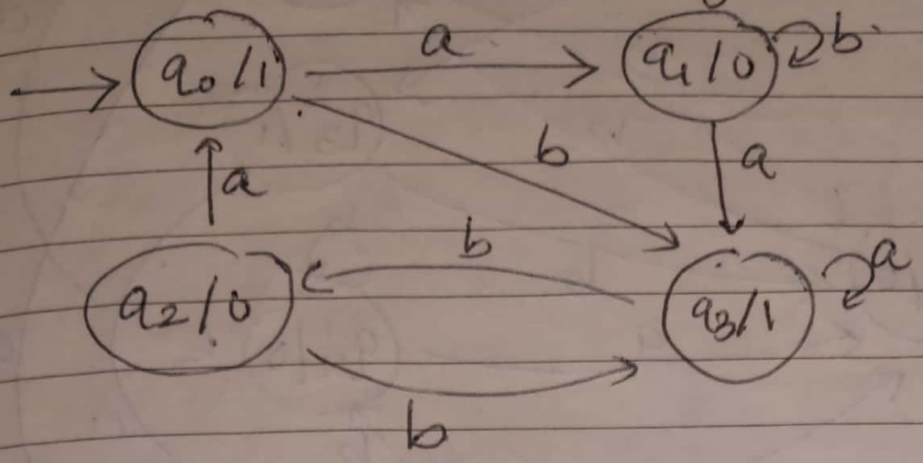
every time there is an 'a' after another 'a' or a 'b' after another 'b', 1 is produced hence counting the occurrences of double letters.

for example for abaaabb

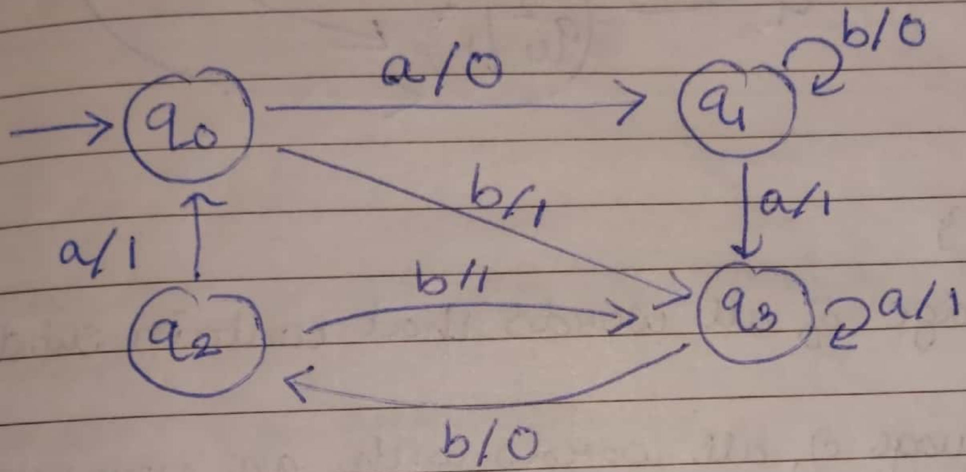


thus, giving output = $\boxed{00001101}$

Q3 Convert Moore to Mealy machine

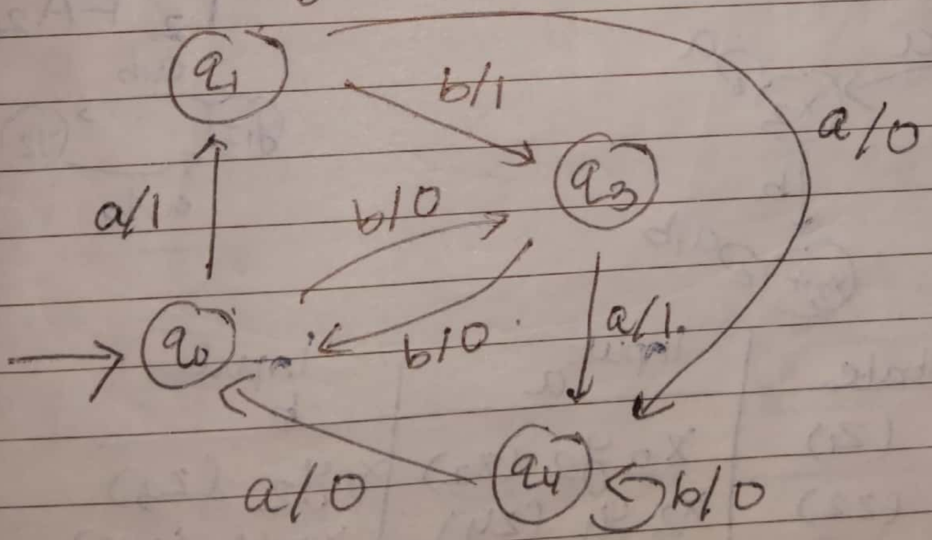


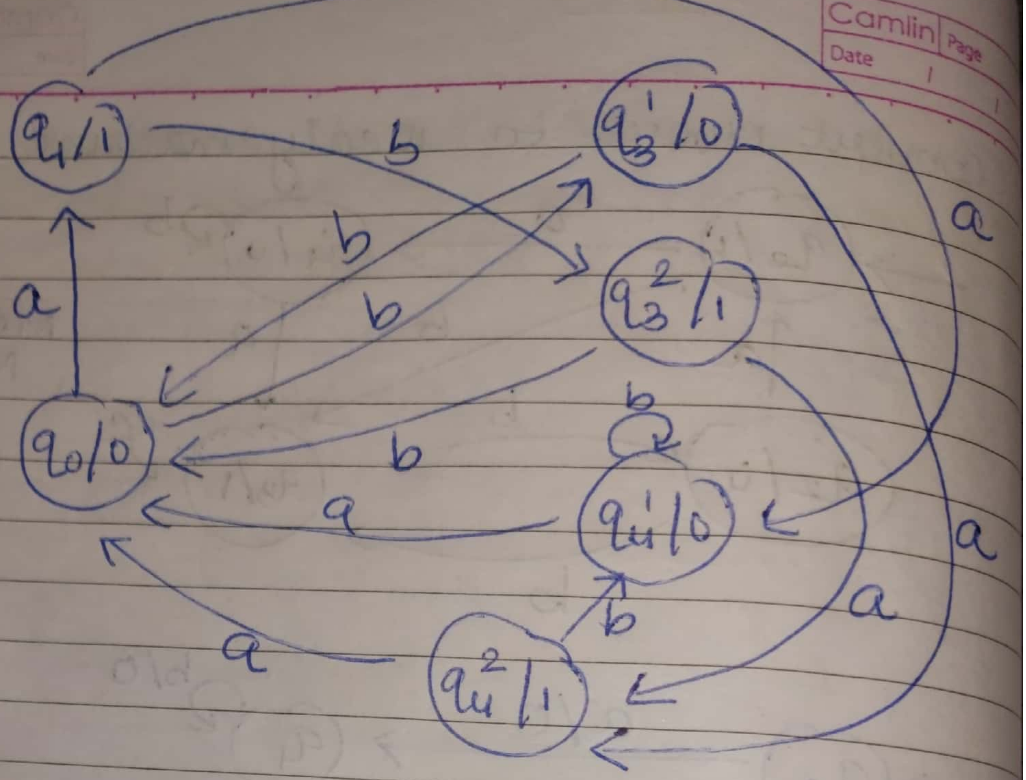
Moore Machine



Mealy Machine

Q4 Convert Mealy into Moore machine



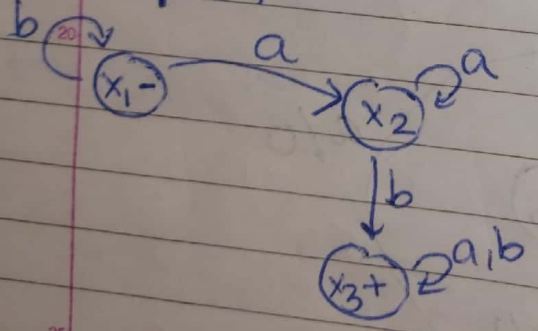


Q5. $\Sigma = \{a, b\}$

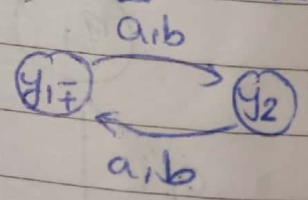
L_1 : language of all words that contain substri[ng] ab.

L_2 ; language of all words with an even numb[er] of letters.

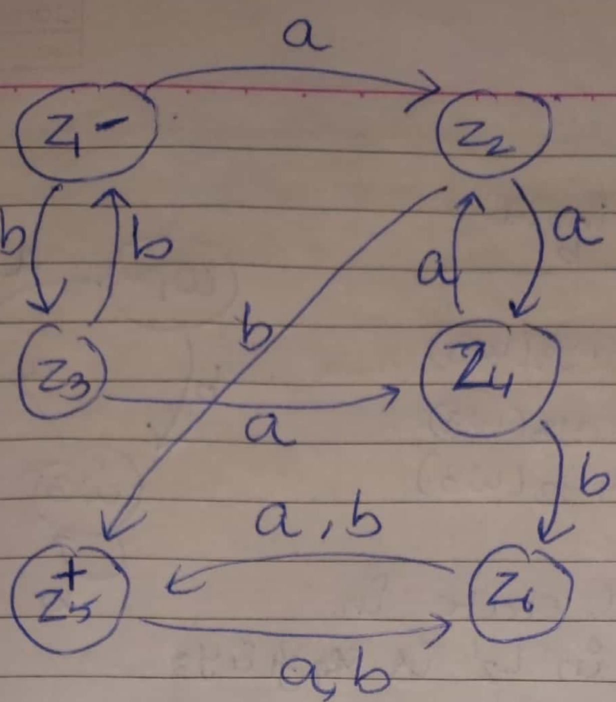
L_1 FA_1



L_2 FA_2



	new state	Input a	Input b
+(-)	x_1y_1 (z_1)	x_2y_2 (z_2)	x_1y_2 (z_3)
	x_2y_2 (z_2)	x_2y_1 (z_4)	x_3y_1 (z_5)
-	x_1y_2 (z_3)	x_2y_1 (z_4)	x_1y_1 (z_1)
+-	x_2y_1 (z_4)	x_2y_2 (z_2)	x_3y_2 (z_6)
(+)-	x_3y_1 (z_5)	x_3y_2 (z_6)	x_3y_2 (z_6)
+	x_3y_2 (z_6)	x_3y_1 (z_5)	x_3y_1 (z_5)

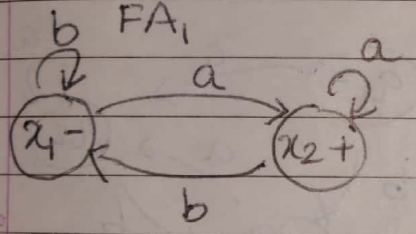


$\boxed{L_1 \cap L_2}$

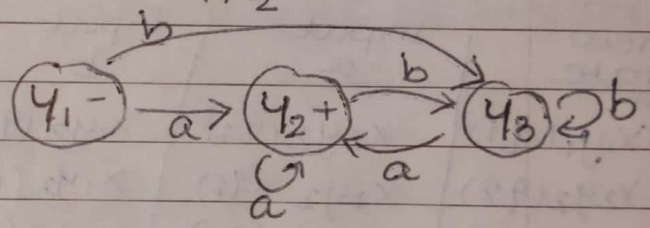
Q6. are the languages accepted by following FAs equivalent?

Decidability Problems.

FA1

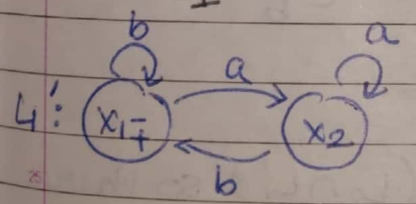


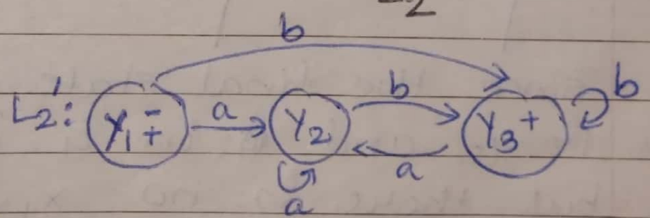
FA2



L1

L2





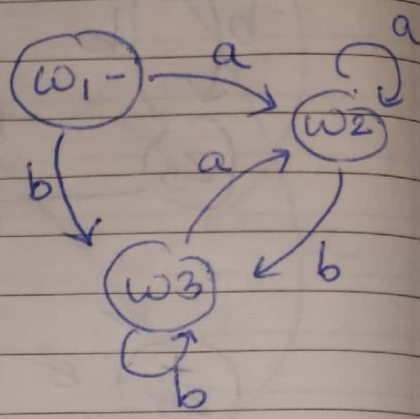
now building finite automata for

$(L_1' + L_2)' + (L_2' + L_1)'$ decidability th^m

$= (L_1 \cap L_2') + (L_2 \cap L_1')$

$(L_1 \cap L_2')$

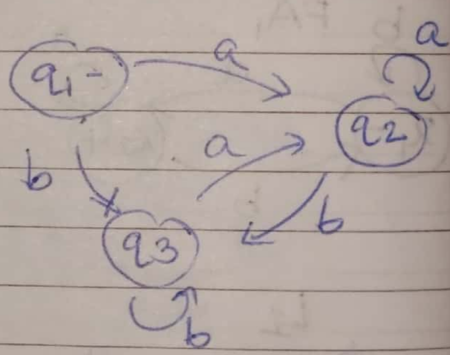
new state	input a	input b
$\pm x_1y_1(w_1)$	$x_2y_2(w_2)$	$x_1y_3(w_3)$
$+x_2y_2(w_2)$	$x_2y_2(w_2)$	$x_1y_3(w_3)$
$+x_1y_3(w_3)$	$x_2y_2(w_2)$	$x_1y_3(w_3)$



Since the final state in L_1 is x_2 and in L_2' is y_1 & y_3 but there is no x_2y_1 or x_2y_3 as a new state so there is no final state. That means no words are accepted by finite automata of $L_1 \cap L_2' \Rightarrow \phi$.

$(L_2 \cap L_1')$

new state	input a	input b
$\pm x_1y_1(q_1)$	$x_2y_2(q_2)$	$x_1y_3(q_3)$
$+x_2y_2(q_2)$	$x_2y_2(q_2)$	$x_1y_3(q_3)$
$+x_1y_3(q_3)$	$x_2y_2(q_2)$	$x_1y_3(q_3)$



Since the final state in L_2 is y_2 and in L_1' is x_1 but there is no x_1y_2 in $(L_2 \cap L_1')$ so there is no final state. That means no words are accepted by finite automata of $L_2 \cap L_1' \Rightarrow \phi$

$$(L_1 \cap L_2') + (L_2 \cap L_1')$$
$$= \phi + \phi$$
$$= \phi$$

$\therefore$ No words are accepted by following FAs that means the finite automata are equivalent

Q7. Pumping Lemma :
Prove $L = \{a^n b^{2n}\} = \{abb, aabbbb, aaabbb$
is non-regular

Proof: $L = \{a^n b^{2n}\}$
$= \{abb, aabbbb, aaabbbbbb, \ldots\}$

Assume L is regular.
By pumping lemma, there exists x, y,
such that xy^nz are all words of L.

Case 1: If y is made up of all a's

$\Rightarrow$ $x = \Lambda$, $y = aa$, $z = b$ } aaaaab~~b~~

In xyyyyz,
$\Rightarrow$ there are more a_s than b_s which
not true since number of a_s should be half
number of bs

Case 2: If y is made of all b's

$\Rightarrow$ $x = a$, $y = b$, $z = \Lambda$ } ~~a~~abbbbbbbb

$\Rightarrow$ In xyyyyyyyz, there are more bs tha
when number of bs should be only
the number of as.

Case 3: If y contains substring ab.

$\Rightarrow$ $x = a$, $y = ab$, $z = b$ } ~~aa~~ abababb~~b~~

$\Rightarrow$ In xyyyz, there are ~~mo~~ two copies of
(there is substring ab more than once)
which is wrong. Since there should be
one substring ab.

Hence, y does not exist and L is not regular.

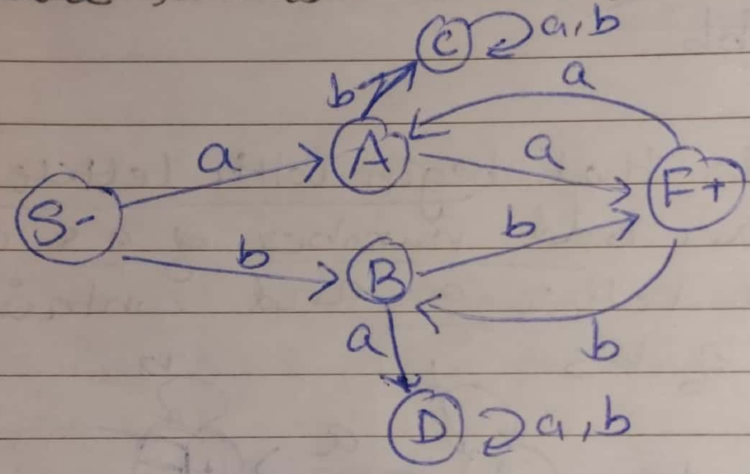
Hence Proved using pumping lemma that L is nonregular.

Scanned with CamScanner

Q8 Find CFGs $\Sigma\{a,b\}$

(a) language of all words that consist only of double letters (aa or bb)

FA:



CFG:

$S \rightarrow aA \mid bB$

$A \rightarrow aF \mid bC$

$B \rightarrow bF \mid aD$

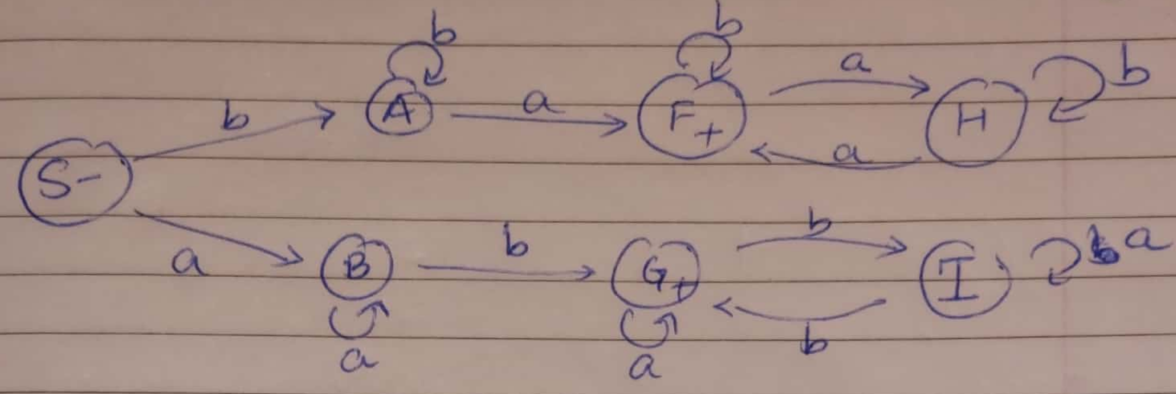
$F \rightarrow aA \mid bB \mid \Lambda$

$C \rightarrow aC \mid bC$

$D \rightarrow aD \mid bD$

(b) all words that begin with the letter b and contains an odd number of a's or begin with letter a and contains an even number of b's.

FA:



CFG:

$S \rightarrow bA \mid aB$

$A \rightarrow bA \mid aF$

$B \rightarrow aB \mid bG$

$F \rightarrow aH \mid bF \mid \Lambda$

$G \rightarrow bI \mid aG \mid \Lambda$

$H \rightarrow bH \mid aF$

$I \rightarrow bG \mid aI$