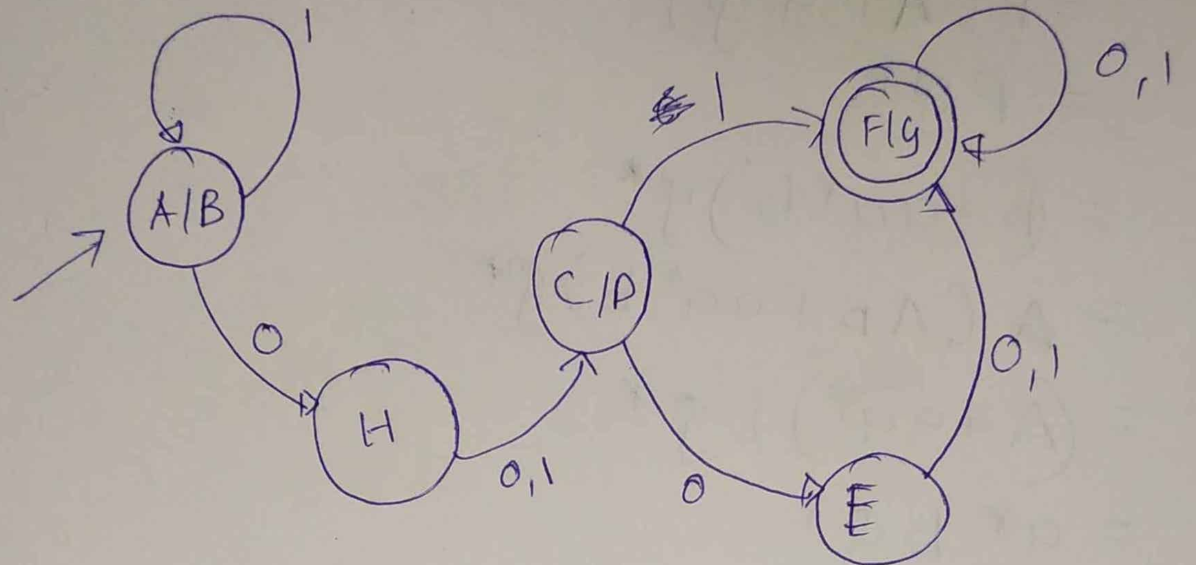
Name - Sudhanwa Bokade
Roll No-20BAI10302 Code- CSE2004
Slot E21+E22+E23
Subject - TOC (Theory of compiler design)

Q1



Q2a) i) $\varepsilon + 1^*(011)^*(1^*(011)^*)^* = (1+011)^*$

From identity rule,

$(0^*1^*)^* = (0+1)^*$ we have,

$R = \varepsilon + PP^*$, where $P = 1^*(011)^*$

$= P^*$ using I9

$= (Q^*S^*)^*$ where $Q=1$, $S=011$

$= (Q+S)^*$ using I11

$= (1+011)^*$

Hence Proved.

Q2(a)(ii)

$P + PQ^*Q = a^*bQ^*$ where $P = b + aa^*b$

and Q is any regular expression

Ans $LHS = P\Lambda + PQ^*Q$

$= P(\Lambda + Q^*Q)$

$= PQ^*$

$= (b + aa^*b)Q^*$

$= \Lambda(\Lambda b + aa^*b)Q^*$

$= (\Lambda + aa^*)bQ^*$

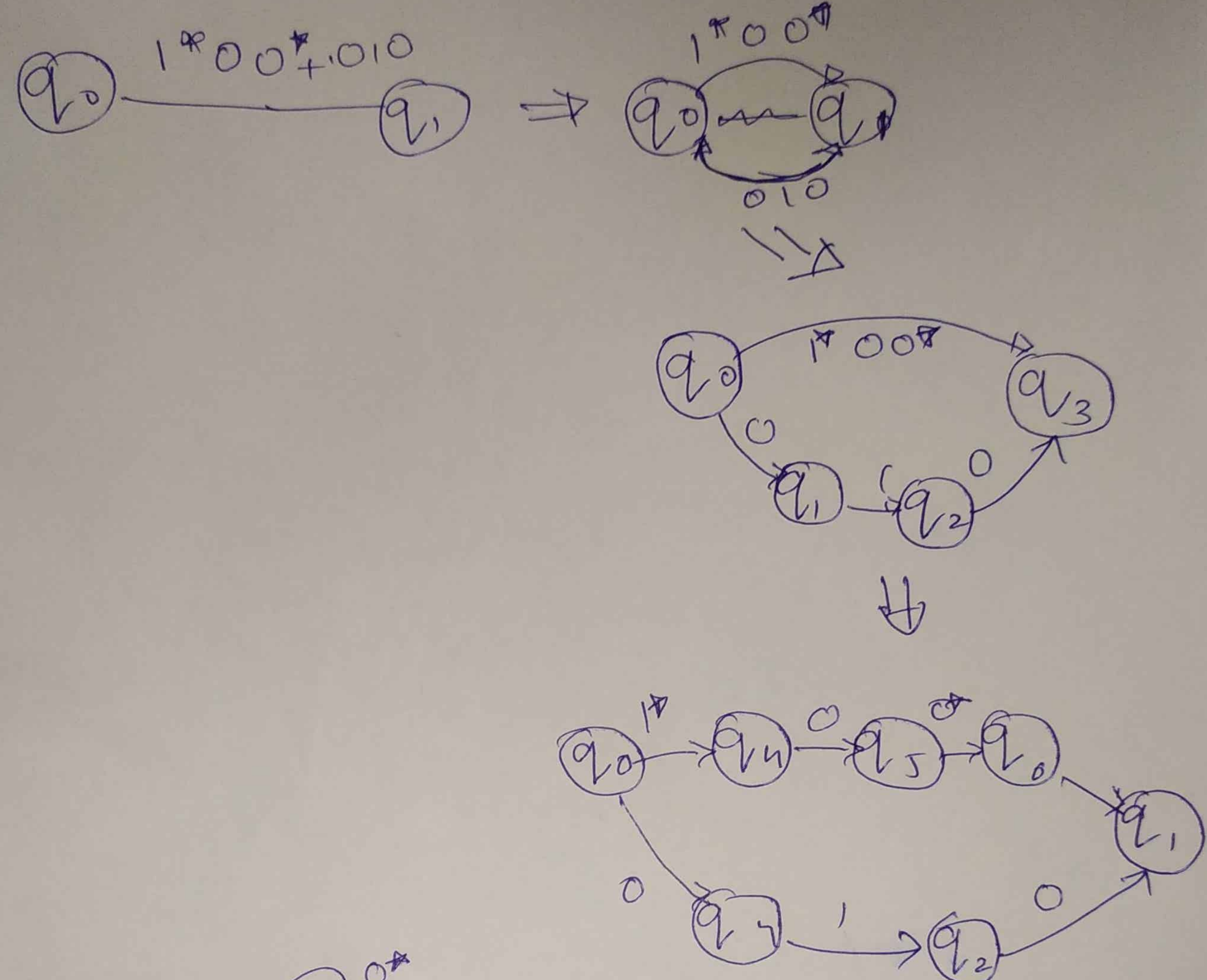
$= a^*bQ^*$

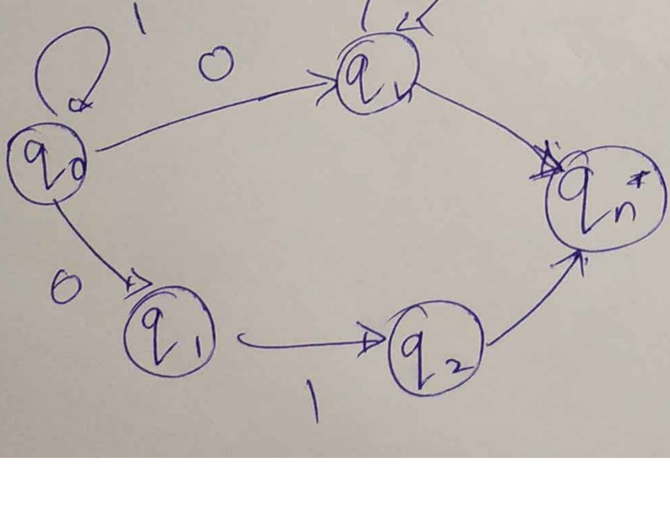
$= RHS$

Therefore hence proved.

q 2b

$1^* 0 0^* + 010$.





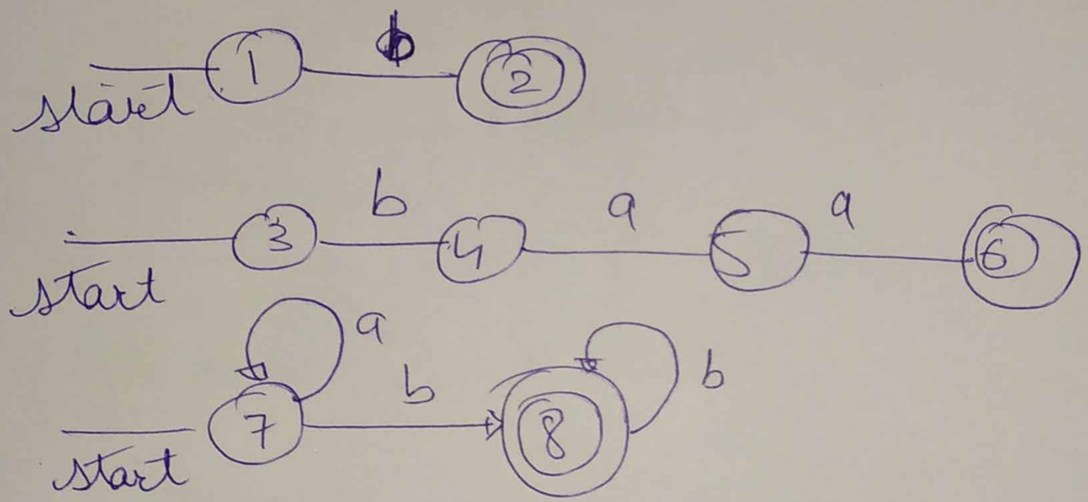
9 ~~7~~ (a)

F ~~B~~ Regular definitions
None.

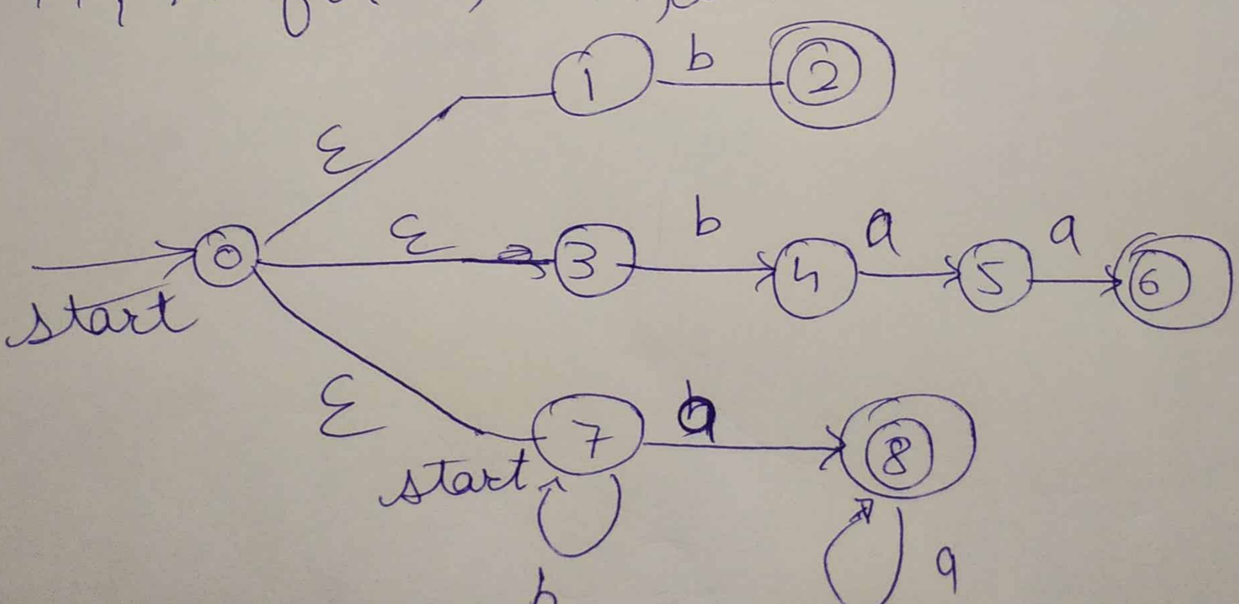
Req Transition Rules

b {} /* actions are omitted here */
baa{}

As The three tokens are recognized by the automata



a) NFA for b, baa, and $b^* a^+$.



Q3 b) $a^n b^n = \{ab, aabb, aaabbb\}$

aabb → Push A.

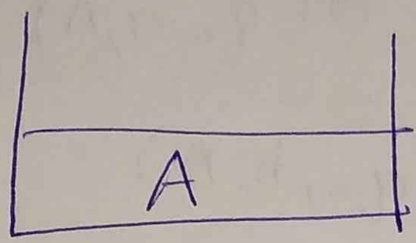
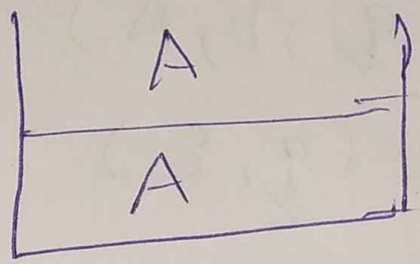


abb ⇒ Push A



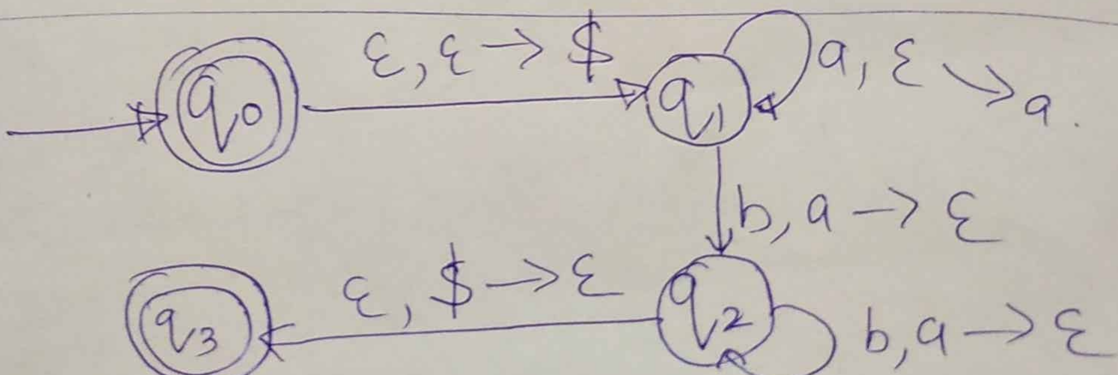
bb pop.

Push down Automata (PDA) accepting $\{a^n b^n \mid n \geq 0\}$

Initial state q_0 :-

string = {ab, aabb, aaabbb, aaaabbbb, ...}

State	Input	Transition function	Stack	Stack altering
			Z	
(q_0)	aabb	$\delta(q_0, a, Z) = (q_0, AZ)$	AZ	q_0
q_0	aabb	$\delta(q_0, a, Z) = \delta(q_0, a, A)$	AAZ	q_0
q_0	aabb	$\delta(q_0, b, A) \Rightarrow (q, \varepsilon)$ (POP)	AZ	q_1
q_1	aabb	$\delta(q_1, b, A) \Rightarrow (q, \varepsilon)$	Z	q_1
q_1	ε	$\delta(q, \varepsilon, Z)$	ε	q_1



Q4

$S \rightarrow a \mid abSb \mid aAb.$

$A \rightarrow bS \mid aAAb$

(I) String : ababab b

Left most derivation

$S \rightarrow abSb$

~~(II) Rigt~~ $S \rightarrow abaAbb$

$S \rightarrow ababSbb$

$S \rightarrow ababab b$

(II) Right most derivation

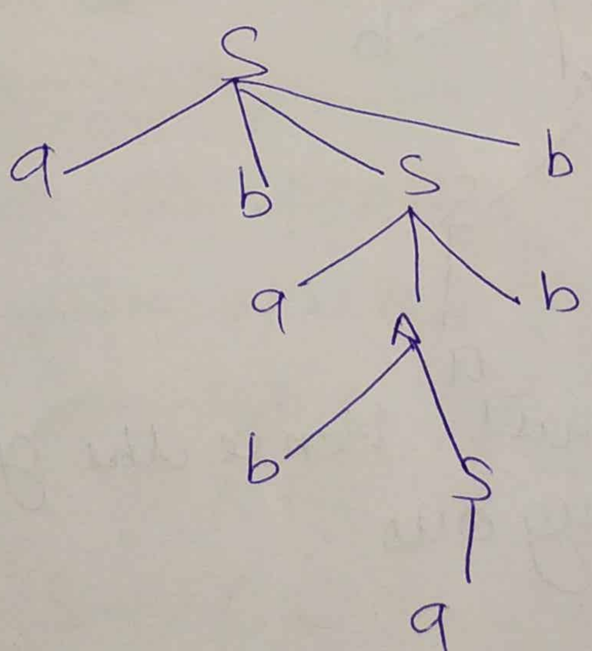
$S \rightarrow abSb.$

$S \rightarrow abaAbb$

$S \rightarrow ababSbb$

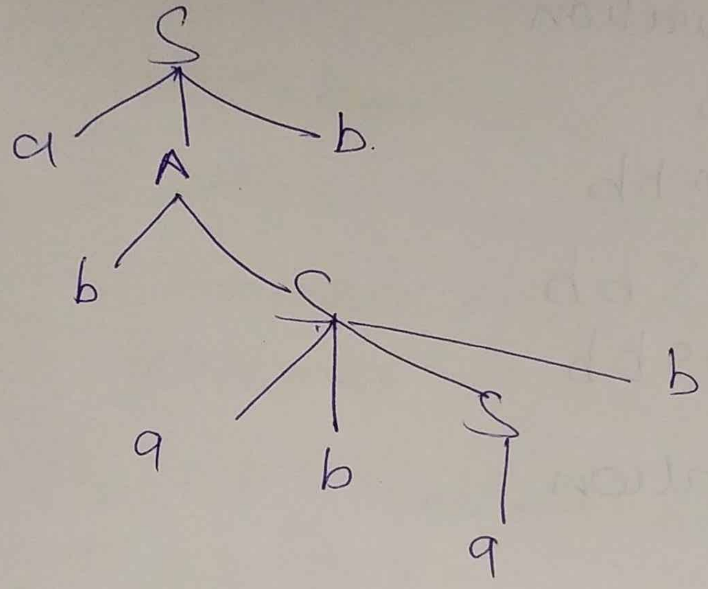
$S \rightarrow abababb$

(III) Derivation Tree

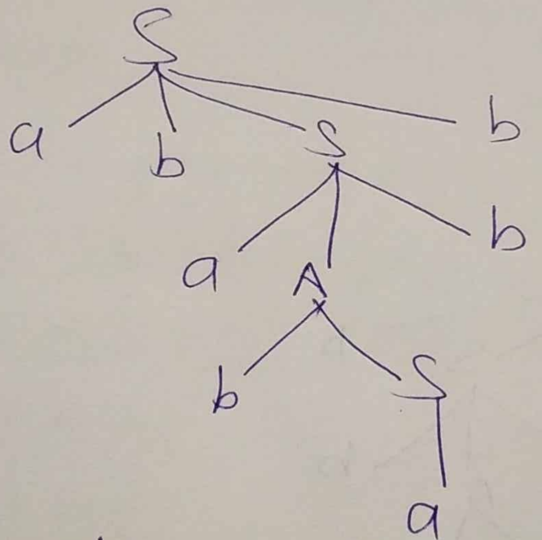


If we get 2 or more parse trees or left most OR rightmost derivation for a string in given grammar then the grammar is ambigous

Parse tree 1



Parse Tree 2



Two parse trees exist. Hence the given grammar is ambigous.

Q5a) Consider the CFG

$S \rightarrow ABAC$

$A \rightarrow aA \mid \varepsilon$

$B \rightarrow aB \mid \varepsilon$

$C \rightarrow c$

~~Consider the CFG.~~

& where S is the start symbol
Eliminate epsilon productions from this grammar.

This CFG contains the epsilon productions,
$A \rightarrow \varepsilon$, $B \rightarrow \varepsilon$.

To eliminate $A \rightarrow \varepsilon$, replace A with epsilon in the RHS of the productions, $S \rightarrow ABAC$
$A \rightarrow aA$,

For the production, $S \rightarrow ABAC$, replace A with epsilon one by one as, we get -

$S \rightarrow BAC$

$S \rightarrow ABC$

$S \rightarrow BC$

(ont 5 a)

For the production, $A \rightarrow aA$, we get,

$A \rightarrow a$.

Now the grammar becomes,

$S \rightarrow ABAC \mid ABC \mid BAC \mid BC$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid \varepsilon$

$C \rightarrow c$

To eliminate $B \rightarrow \varepsilon$, replace A with epsilon in the RHS of the production, $S \rightarrow ABAC$, $B \rightarrow bB$,

For the production, $S \rightarrow ABAC \mid ABC \mid BAC \mid BC$, replace B with epsilon as, we get,

$S \rightarrow AAC \mid AC \mid C$

For the production, $B \rightarrow bB$, we get, $B \rightarrow b$

Now the grammar becomes.

$S \rightarrow ABAC \mid ABC \mid BAC \mid BC \mid AAC \mid AC \mid C$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b$

$C \rightarrow c$

which does not contain any epsilon production

95b)

$S \rightarrow 0A \mid B.$

$A \rightarrow 0A \mid A$

$B \rightarrow 1B \mid 1$

There is no null production

Removing unit production

The unit productions are $A \rightarrow A.$

$A \rightarrow 0A \mid 0A$

$S \rightarrow 0A \mid B.$

$A \rightarrow 0A \mid 0A$

$B \rightarrow 1B \mid 1$

Now removing form $A \rightarrow aB$

Let $X \rightarrow 0,\ Y = 1$

$S \rightarrow XAYB$

$A \rightarrow XA \mid \cancel{X}A$

$B \rightarrow YB \mid 1$

A & B are now in CNF

Now making S in the form.

$A \rightarrow BC$

Let $P \rightarrow XA$

$Q \rightarrow YB$

$\therefore$ $S \rightarrow PQ$

$A \rightarrow XA \mid XA$

$B \rightarrow YB \mid 1$

$X \rightarrow 0$

$Y \rightarrow 1$ CNF Form

$P \rightarrow XA$

$Q \rightarrow YB$