Chapter-14

Pushdown Automata

(Pushdown Stack + Finite Automata)

Memory + FA

Regular language (R.L.) $\xrightarrow{R.E.}$ finite Automata (F.A.)

machine that accepts RL

CFL (context free language) $\xrightarrow{CFG}$ PDA (Pushdown Automata)

machine that accepts CFL

CFL → Regular lg. → FA → CFG

CFL → Non-Regular lg.

eg. Palindrome, equal a and equal b, $a^n b^n$, etc.

for $a^n b^n$	Palindrome	equal a/b
$S \to aSb$	$S \to aSb$	$S \to aSb$
$S \to \lambda$	$S \to bSb$	$S \to bSa$
	$S \to a \mid b \mid \lambda$	$S \to \lambda$

* Write down the CFG for non-regular lg.

a) a^p → where p is a prime no.

no CFG

b) $a^n b^n c^n$

no CFG.

I/P tape → It is tape having inputs and infinite in one dirn.

Date : / /

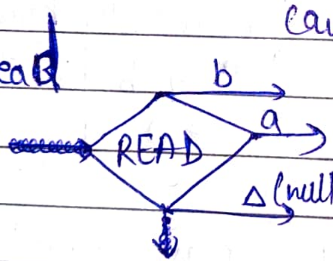
Page No.

CFG = all regular lg + some non-regular lgs.

↓

(RE → FA → CFG)

* PDA (Pushdown Automata)

	FA	PDA (states)	[Flow Chart]
Initial state	⊖	(Start) ↓	no incoming edge, exactly one outgoing edge.
final state	⊕	↓ →(Accept)	one or more incoming edges and no outgoing edge. (Halt state)
Non-final state		↓ →(Reject)	one or more incoming edges and no outgoing edge. → can be single reject state or can be multiple reject state
for every state of FA, you read a symbol and branch to another state		Read 	Accept / Reject.

eg. ~~L = all words ending in a~~

→ In PDA, I/P string is stored in a I/P tape.

↳ infinite in 1 direction

↳ divided into cells

↳ initialize with Δ → blank

I/P tape

aba

Δ	Δ	Δ	Δ	Δ	Δ

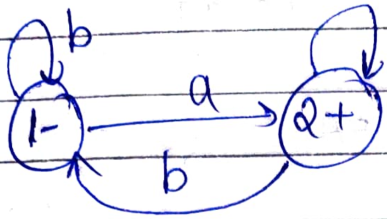
↳

a	b	a	Δ	Δ	..

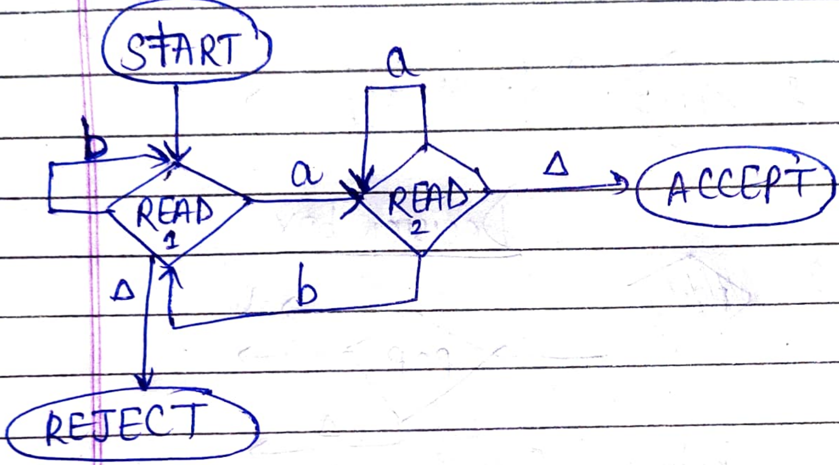
→ for every state in FA, there will be a corresponding read state in PDA.

eg. L = all words ending in a.

$r = (a+b)^* a$

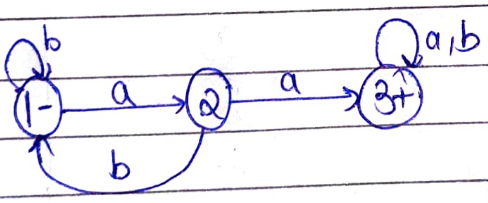


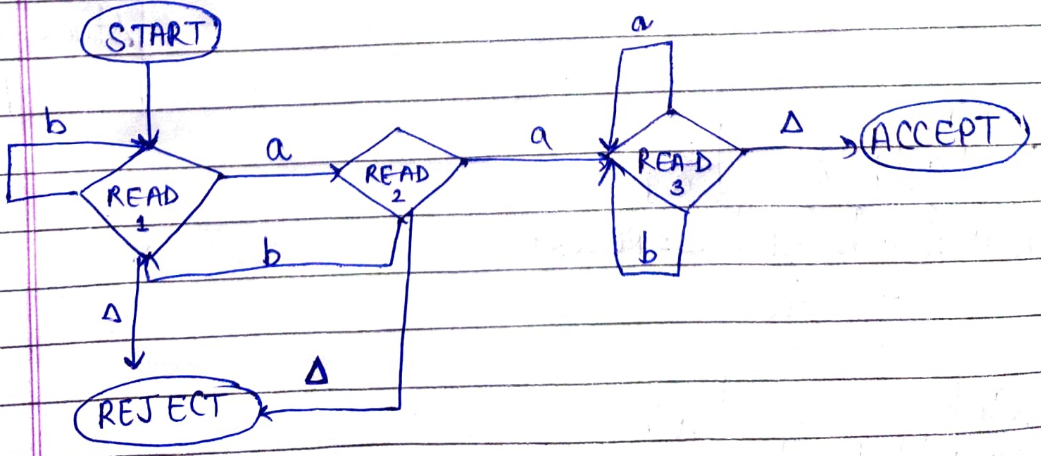
for PDA,
$\Sigma = \{a, b, \Delta\}$.



eg. L = all words having aa somewhere.

$r = (a+b)^* aa (a+b)^*$



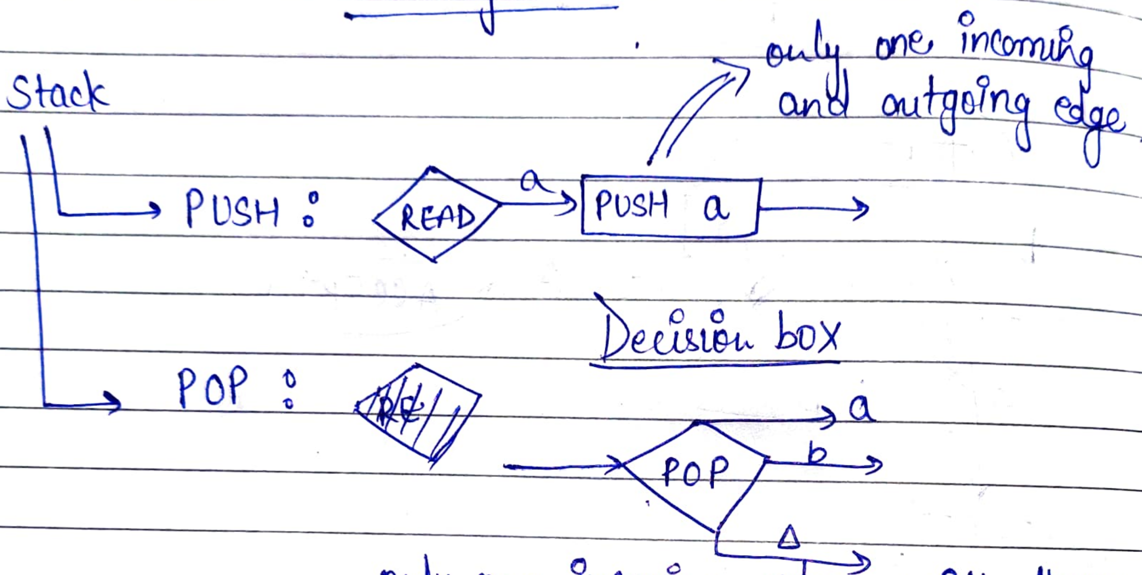


for every pushed a, there will be a poped b, to get a PDA accepted the lg.

Pushdown Automate (PDA) (for non-reg lg)

Pushdaon + finite Automata
Stack

memory + FA



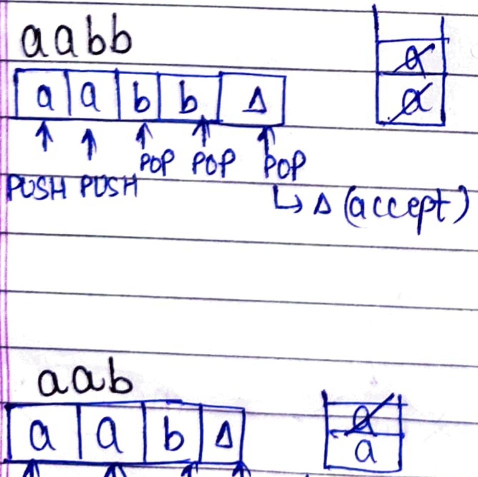
Decision box

only one incoming edge with three outgoing edge of decision

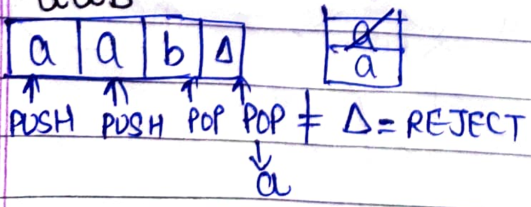
→ $a^n b^n$

I/P Symbol of Read	Action
a	PUSH
b	POP→(match a
Δ	POP=Δ (Accept) stack should be empty

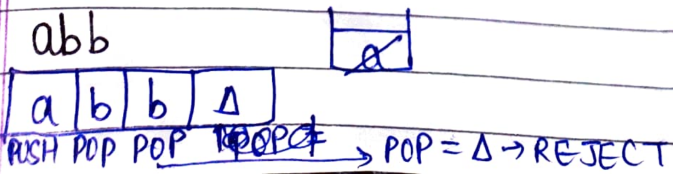
① aabb



② aab



③ abb

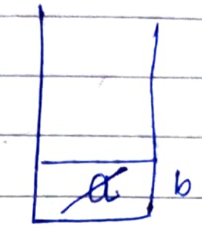


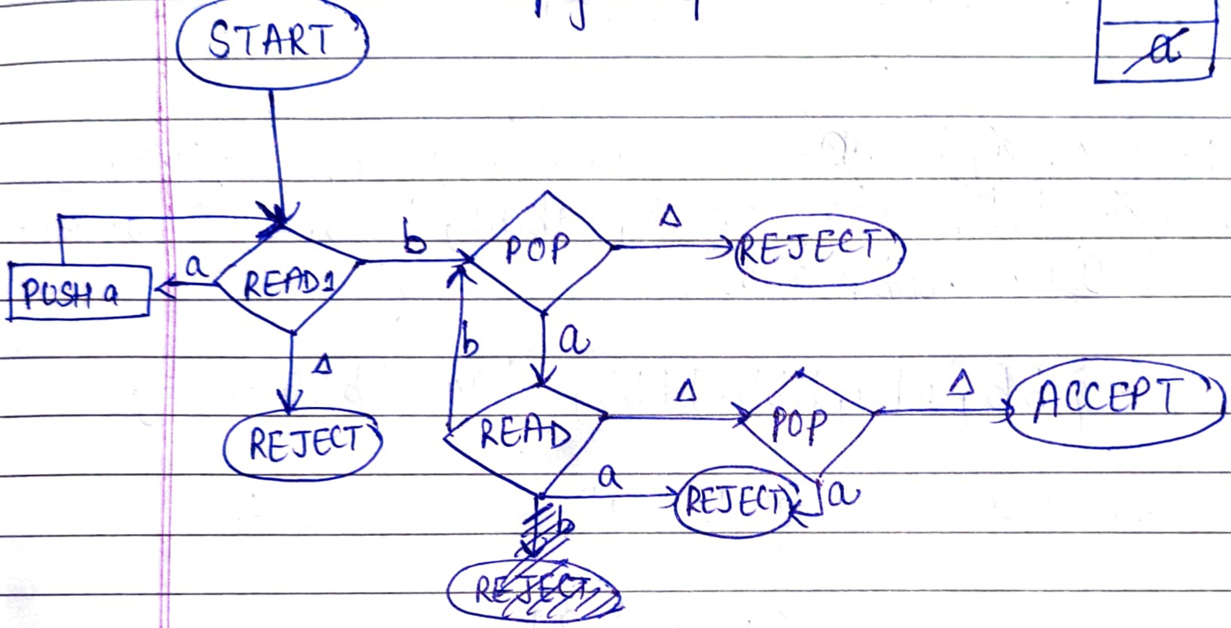
Ques. Draw the PDA for the following languages.

a) $a^n b^n$ $n \geq 1$ (assumption)

| a | b | Δ | → smallest word 'ab'

PUSH POP POP = Δ = empty = accept



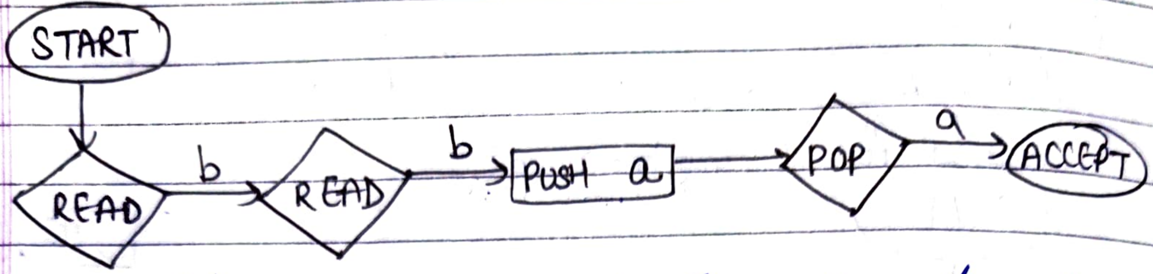


(here words rejected where length of b is greater than a will be rejected such as $a^k b^l$, $l > k$).

READ : I/p Tape , $\Sigma = \{a, b, \Delta\}$
POP : STACK , $\Gamma = \{\Delta, a\}$

* for READ state, there will be three transition i.e. $\{a, b, \Delta\}$ and for POP state, there will be two transition i.e. $\{\Delta, a\}$.

Ques. Determine the lg accepted by PDA.



✓ L = { Δ, a, b, aa, ab, ba, bb, aaa, aab, aba, }
bba, bab, baa, bbb, aaaa, aaba, abaa, bbbb, baaa, bbba,
bbaa, abbb, ~~abab~~, - - - }.

L = { bb, bba, bbb, bbbb, bbaa, bbba - - }

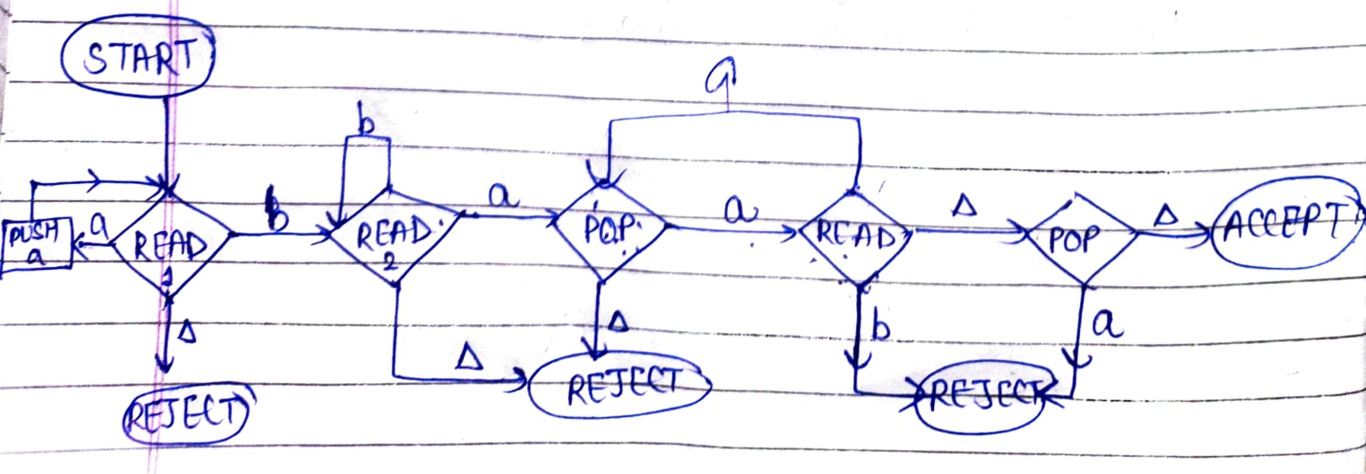
Ques Draw PDA for $a^n b^m a^n$, n, m ≥ 1

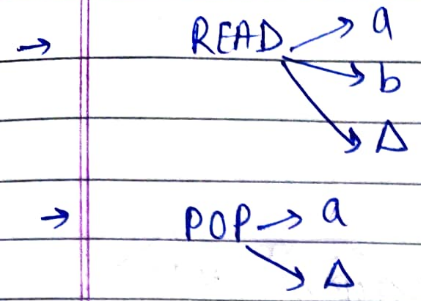
$\underbrace{aa \quad bb^* \quad a}a$

Logic:

I/P Symbol Read	Action.
① a (before b's)	PUSH
② b?	————
③ a (after b's)	POP → (match a)
④ Δ	POP = Δ (accept)

aaba





Ques Draw PDA for $a^l b^m$, $l > m$

⇒ $a^l b^m$

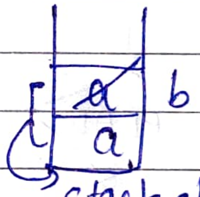
⇒ $a^{k+m} b^m$ $k \geq 1$

⇒ $a^k a^m b^m$

$l, m \geq 1$.

smallest word

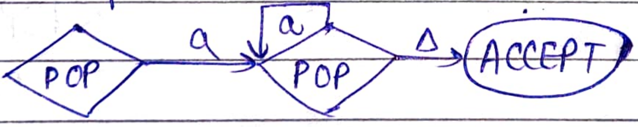
L = {aab}

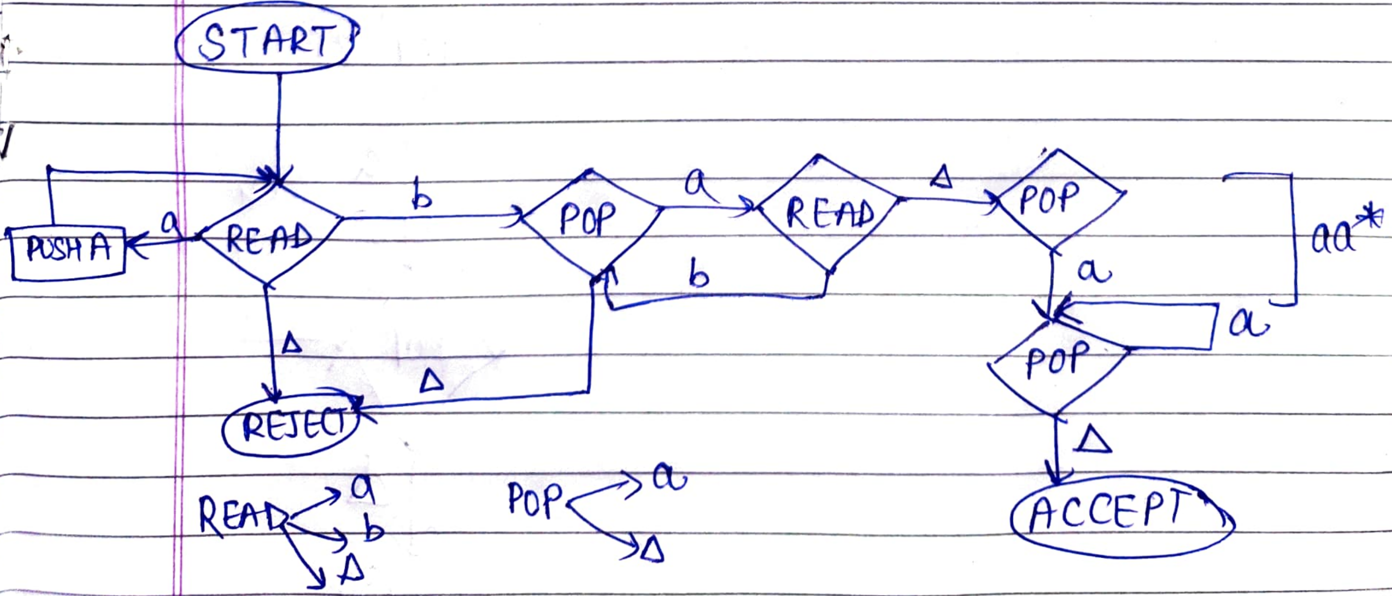


stack should not be empty is atleast 'a'.

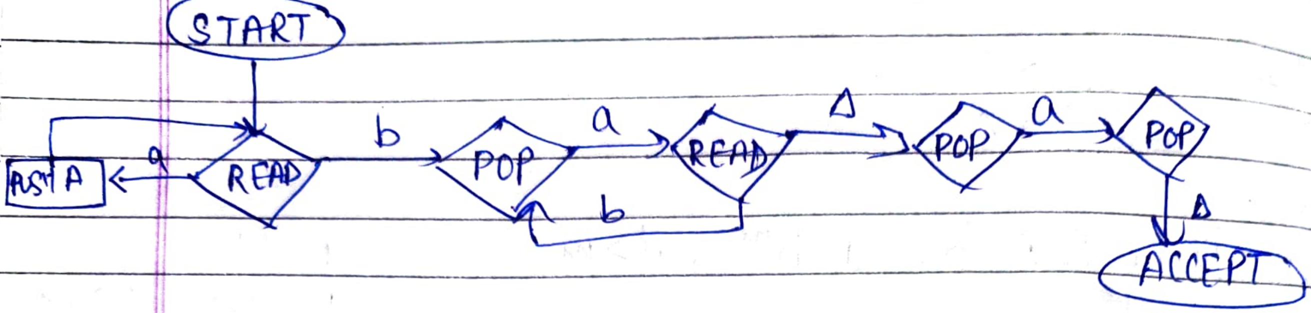
logic:

	I/P Symbol Read	Action
①	a	PUSH
②	b	POP → (match a)
③	Δ	(stack should be empty atleast one a). POP = aa^* (at least one a)

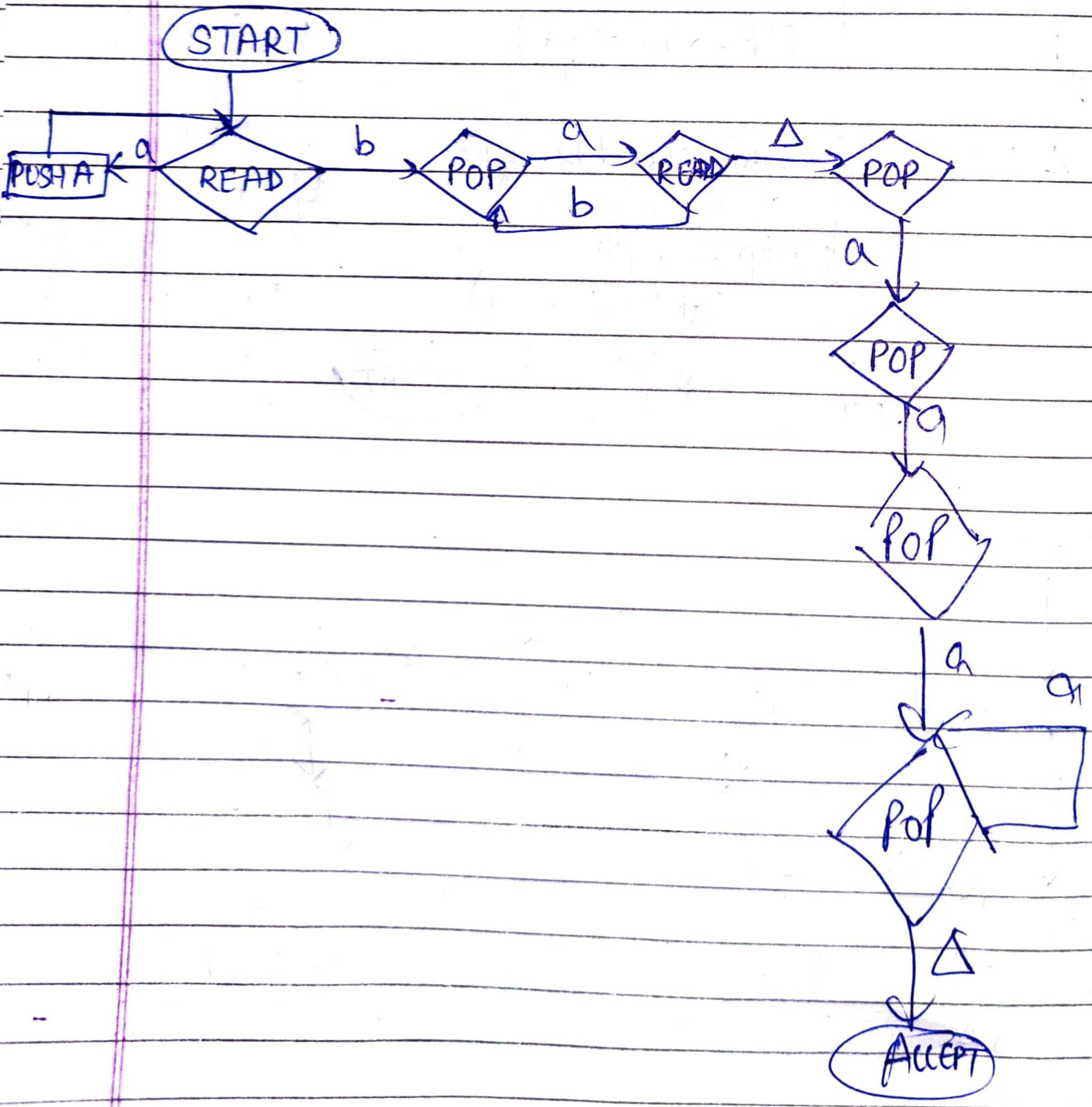




Q $a^{m+1} b^m$



Q $a^{m+l} b^m$, $l \geq 3$ or $l > 2$


START
PUSH A
a
READ
b
POP
a
b
READ
Δ
POP
a
POP
a
POP
a
a
POP
Δ
ACCEPT