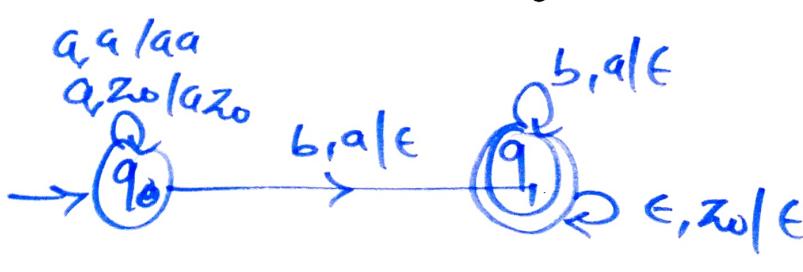
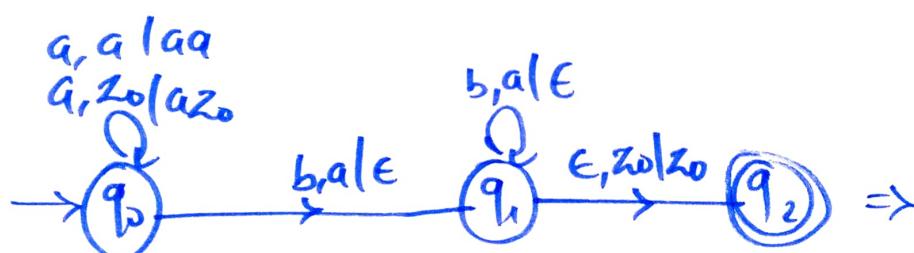
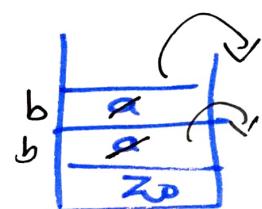
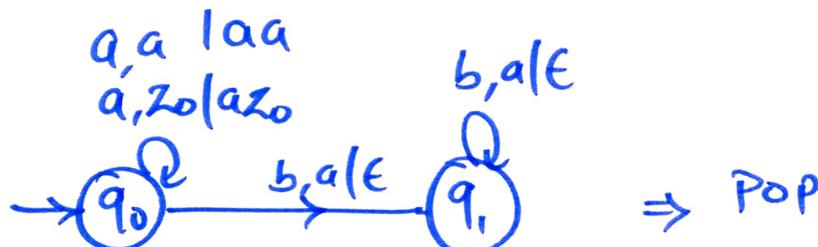
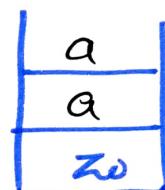


Q) Construct PDA for following language -

i) $L = \{a^n b^n / n \geq 1\}$

↓
then count b
↓
First count no. of a's

$$L = \frac{\underline{aabbb}}{\underline{\text{Push Pop}}} \epsilon$$



Acceptance by empty stack

$$② L = \{ w \mid n_a(w) = n_b(w) \}$$

\Rightarrow Equal no. of a's & b's i.e. count no. of a's equal to no. of b's count.

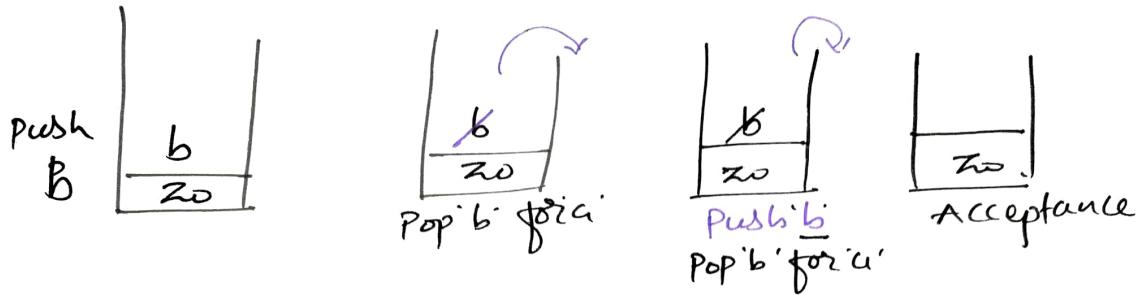
This problem is diff' from $a^n b^n \mid n \geq 1$ because in this you have to read first a & then b but in this given problem string can be like this $w = ab, ba$.

$$\therefore L = \{ \epsilon, \underline{ab}, \underline{ba}, \underline{aab}, \underline{abb}, \underline{bab}, \underline{bab}, \dots \}$$

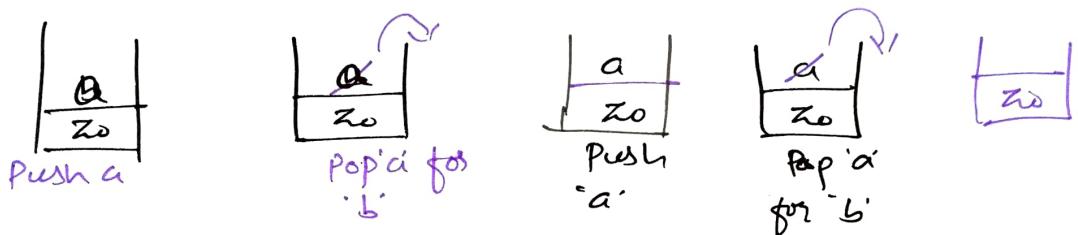
So, we can't say that for all 'a' push 'a' & for all 'b' pop 'a' (or) vice versa there should be other approach

Note: Push the symbols if you don't have chance of popping & if you have chance of popping then pop the symbols.

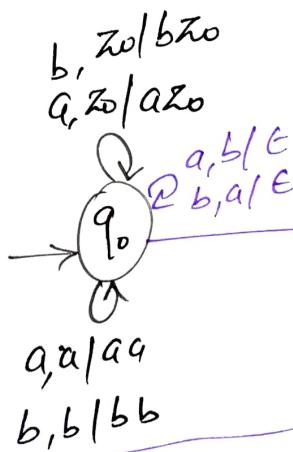
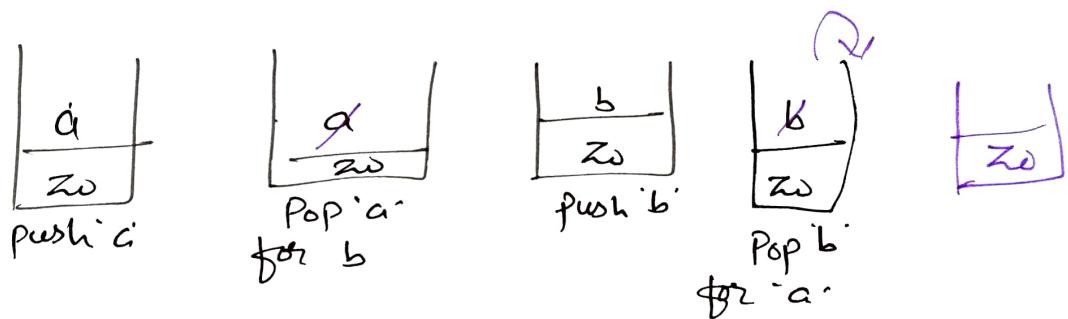
i) $w = baba$



ii) $w = abab$



iii) $w = abbba$



Acceptance by final state,

$q_0 \xrightarrow{a, a/E; b, b/E} q_f$

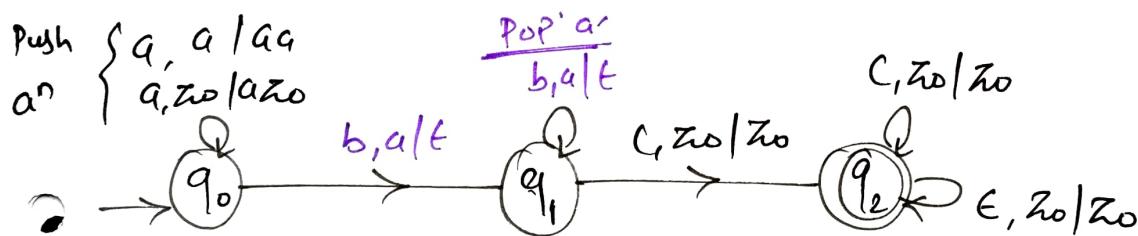
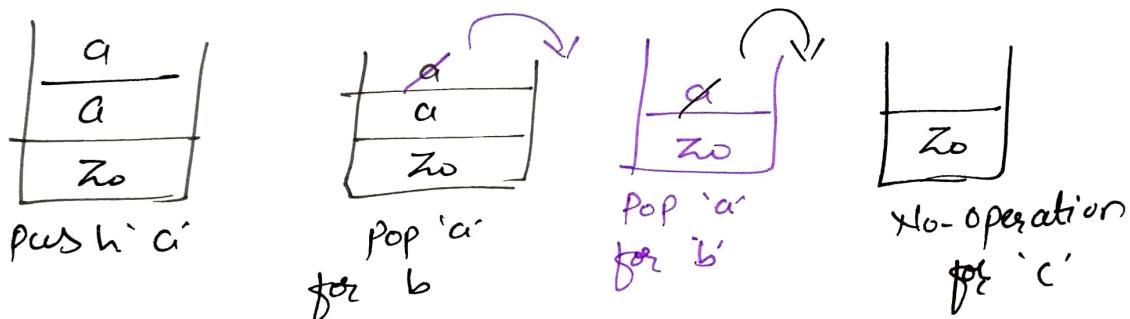
Acceptance by empty stack

$q_0 \xrightarrow{\epsilon, z_0/\epsilon} q_f$

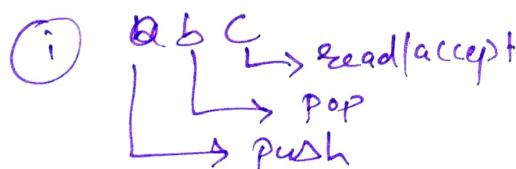
$$③ L = \{ a^n b^n c^m \mid n, m \geq 1 \}$$

⇒ Here, first we compare a & b first & that should be equal. Here for every a, we are pushing 'a' & for every 'b' pop 'a' And finally simply read 'c' (minimum one 'c')

$$w = \frac{aa}{\text{Push}} \frac{bb}{\text{Pop}} c \xrightarrow{\epsilon} \text{No-op}$$



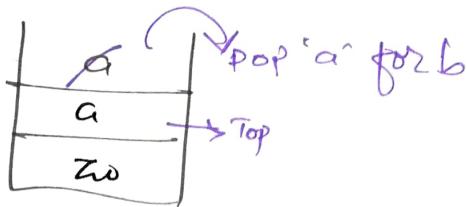
Note:- In DFA for every state there will be transitions but here for every state there is no transition.



⇒ This is dead configuration so for this PDA will halt & it is not accepted.

1 iii

aabcc
push pop read
 (c, z0)

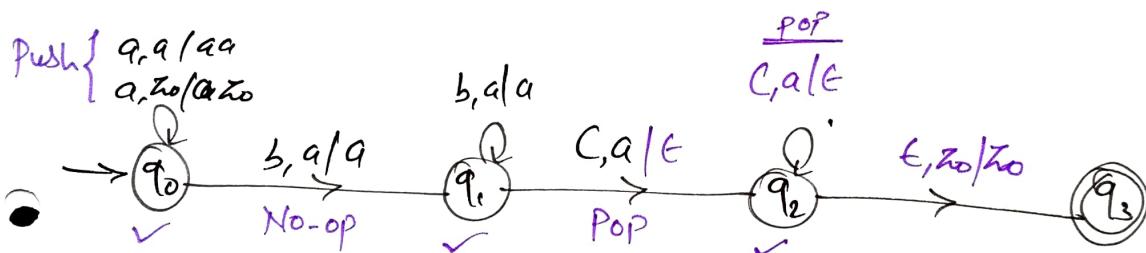


but here (c, a) so it is dead configuration.
 . Halt there.

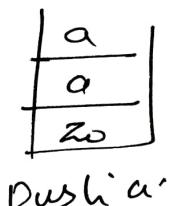
* Halting is at final (or) non-final state; no infinite loop here.

• 4) $L = \{ a^n b^m c^n \mid n, m \geq 1 \}$

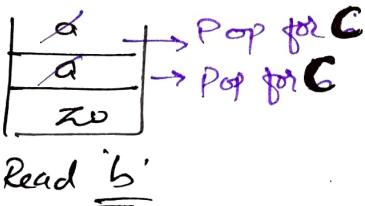
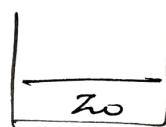
\Rightarrow Here 'a' should match with 'C' & 'b' should be left alone for accepting/reading.



aa bb cc Pop
Push Read ~~bb~~



Push a

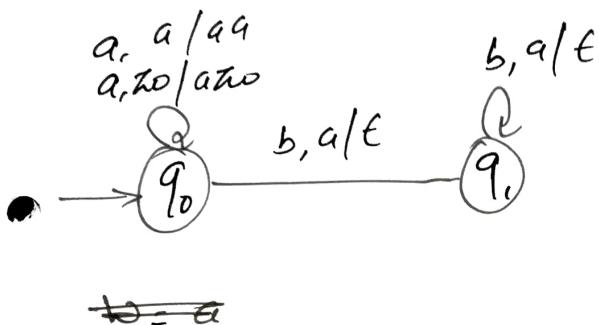
Read b

Note:- Three states are compulsory (q_0, q_1, q_2) because after seeing 'b' you should go to next state & after 'b' no 'a's are allowed & After seeing 'c' go to next state & remember that you have seen 'c' so no 'b' is allowed.

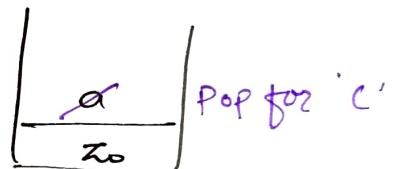
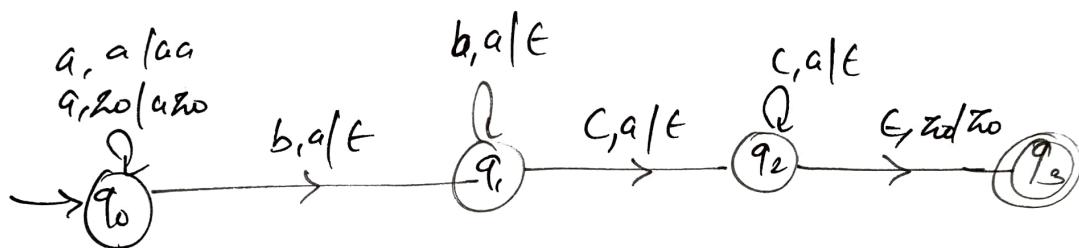
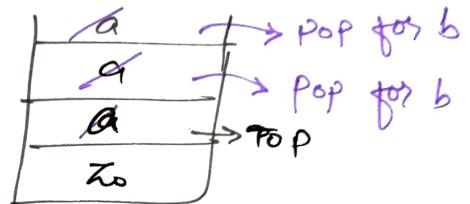
$$5) L = \{a^m b^n c^n \mid m, n \geq 1\}$$

$$\Rightarrow \frac{a^m a^n}{\text{Push 'a'}} \frac{b^n c^n}{\text{Pop for 'a'}}$$

Here count all 'a' & match with b and c



$$w = \frac{a^m a^n}{\text{Push}} \frac{b^n c^n}{\text{Pop}}$$



So, here for 'a', push it on the stack & after this 'b' is there then pop 'a' for all 'b' & change the state i.e. q_1 . For 'C' we can't do the Pop operation at q_1 state itself because after b, C will come and after C b will come, so order will change For this we have to change the state for 'C'.

$$\textcircled{6} \quad L = \{a^n b^m c^n \mid m, n \geq 1\}$$

$$\Rightarrow a^n b^m b^n c^m \Rightarrow \underline{a^n b^n} \underline{b^m c^m}$$

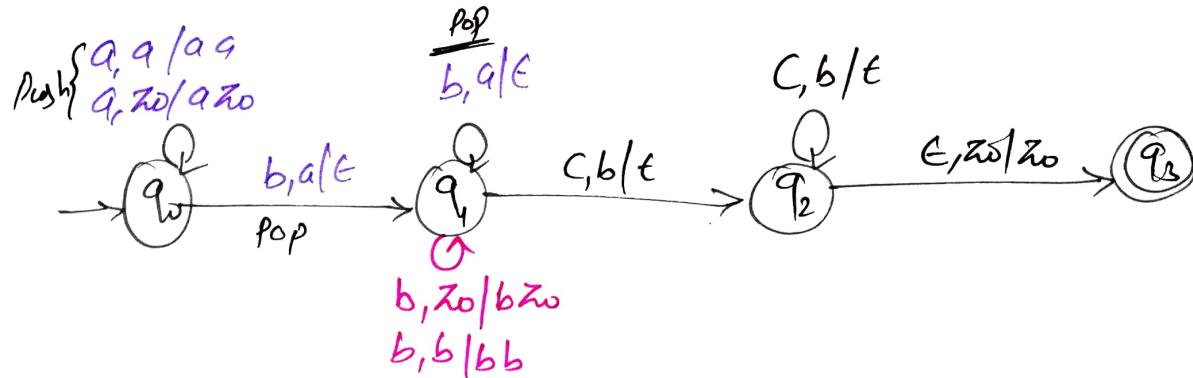
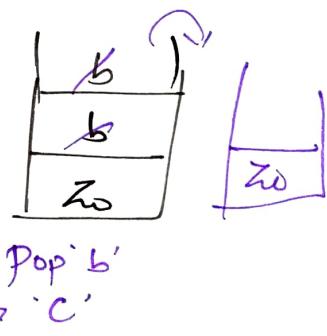
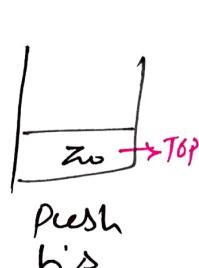
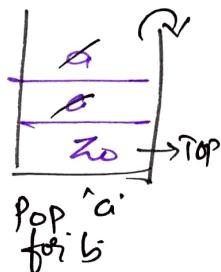
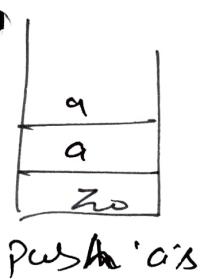
Here, a^n push on stack, whenever you see b 's you have to match them with c 's & Remaining b 's should pushed on the stack & match them against c .

Rewrite as $\{a^n b^n / b^m c^m \mid n, m \geq 1\}$

∴ For all a push 'a' & for all b pop 'a' & for all remaining b 's keep on pushing & match them against

'c's

$$w = \overbrace{aa}^{\text{push}} \overbrace{bb}^{\text{pop}} \overbrace{bb}^{\text{push}} \overbrace{cc}^{\text{pop}}$$



⑦ $L = \{a^n b^m c^{n+m} | n, m \geq 1\}$

$$a^n b^m c^n c^m$$

\Rightarrow

Here, we have to push 'a' then push 'b'

After this for every 'c' we have to pop a & b