Two stack PDA

A 2 stack - pushdown Automata is similar to a PDA, but it has two stacks instead of one. In each transition, we must denote the pop and push action on both stacks.

A machine using two stack pushdown accepts the recursively enumerable language.

We use 2 stack PDA to (find the solution of build) non-Context free language

eg. $L_1 = \{a^n b^n c^n \mid n \geq 1\}$

$L_2 = \{a^n b^m a^n b^m \mid m, n \geq 1\}$

$L_3 = \{w w^R \mid w \in \{a, b\}^+\}$

1) A 2 stack PDA is similar to PDA, but it has two stacks instead of one.

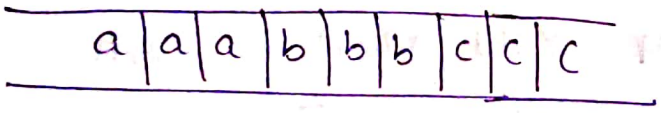
2) for each transition, we can (Push, Pop, No change) on both stacks.

3) Both stacks works independently.

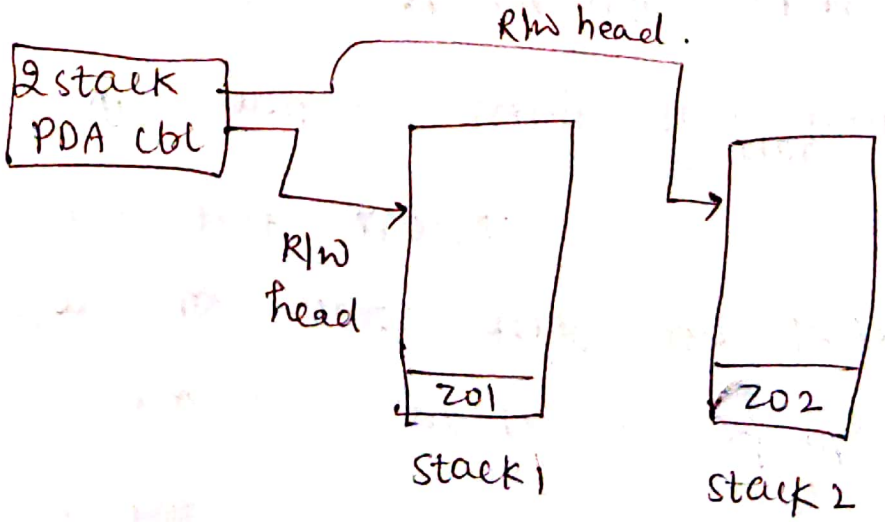
eg. Design a 2 stack PDA for

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

$L = \{abc, aabbcc, aaabbbccc \text{ ---} \}$



I/p tape.



δ is defined

$\delta(q_0, a, z_{01}, z_{02}) \rightarrow (q_1, a z_{01}, z_{02})$

$\delta(q_1, a, a, z_{02}) \rightarrow (q_1, aa, z_{02})$

$\delta(q_1, b, a, z_{02}) \rightarrow (q_2, \lambda, b z_{02})$

$\delta(q_2, b, a, b) \rightarrow (q_2, \lambda, bb)$

$\delta(q_2, c, z_{01}, b) \rightarrow (q_3, z_{01}, \lambda)$

$\delta(q_3, c, z_{01}, b) \rightarrow (q_3, z_{01}, \lambda)$

$\delta(q_3, \lambda, z_{01}, z_{02}) \rightarrow (q_f, \lambda, \lambda)$