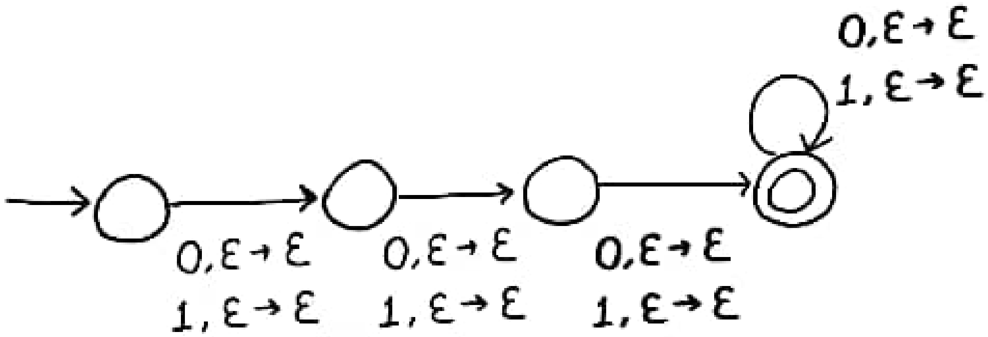
CSE331: Automata and Computability

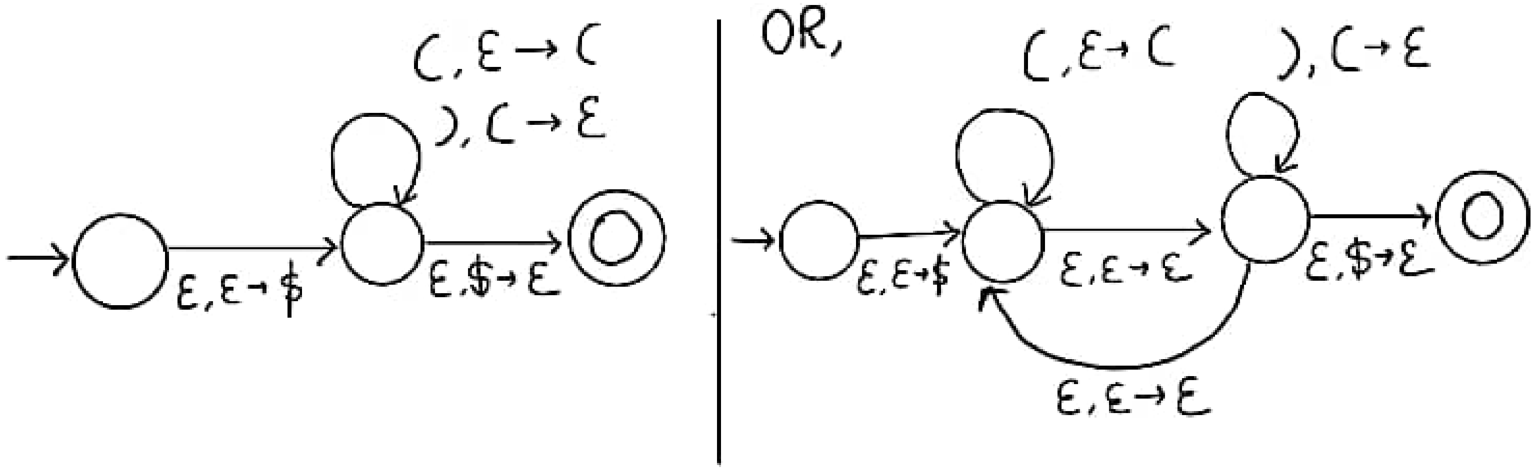
Quiz on: Pushdown Automata

Construct Pushdown Automata for the following languages.

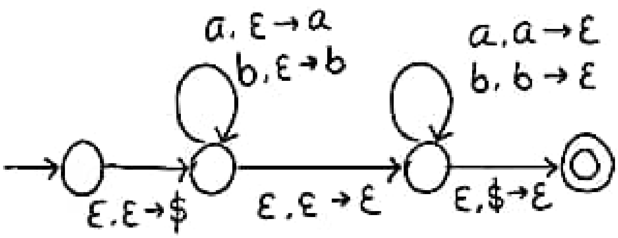
a) $L = \{w \in \{0,1\}^*$: length of w is at least three.$\}$ [Hint: Recall what kind of language L is.]



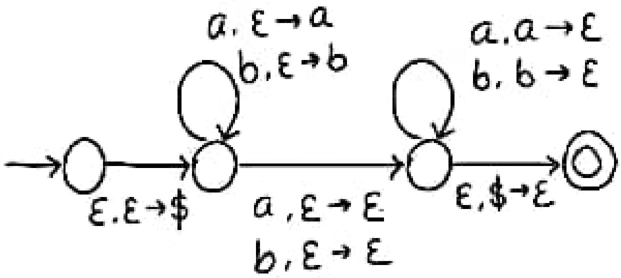
b) $L = \{w \in \{ (\,,\,) \}^*$: w is a valid parenthesis$\}$



c) L = {w ∈ {a , b}*: w is a even length palindrome} / L = {w ∈ {a , b}*: w is a odd length palindrome}



even Length Palindrome



odd Length Palindrome

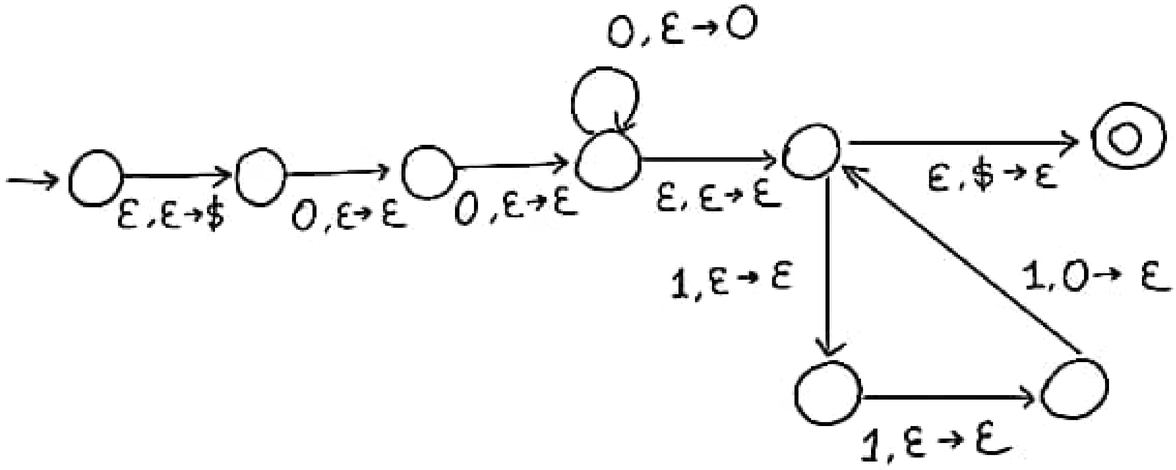
d) L = {w ∈ {0, 1}*: $0^{n+2}1^{3n}$, where n ≥ 0}

$0^{n+2}1^{3n} \Rightarrow 0^2 0^n 1^{3n}$

$n = 0$: 00

$n = 1$: 000 111

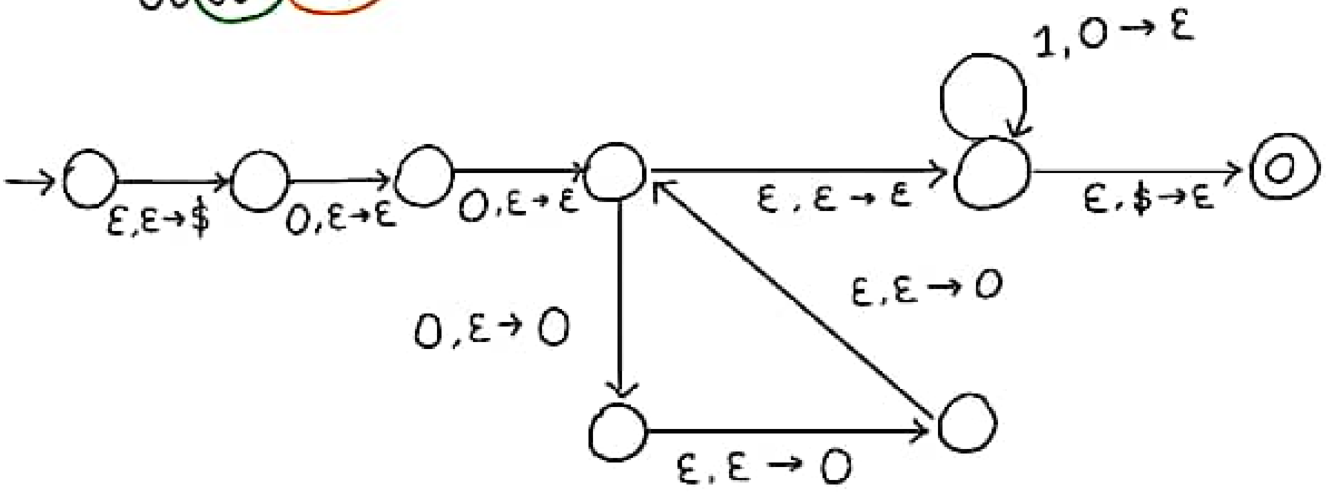
$n = 2$: 00 0 0 111 111



e) $L = \{w \in \{0, 1\}^*: 0^{n+2}1^{3n}$, where $n \geq 0\}$ **[Alternate Solution Idea]**

$n=2,\quad 0000\,111111$

$00\,000\,000\,111111$



f) $L = \{w \in \{a, b, c\}^*: a^i b^j c^k$, where i is even, $j = 2k$ and $i, j, k \geq 0\}$

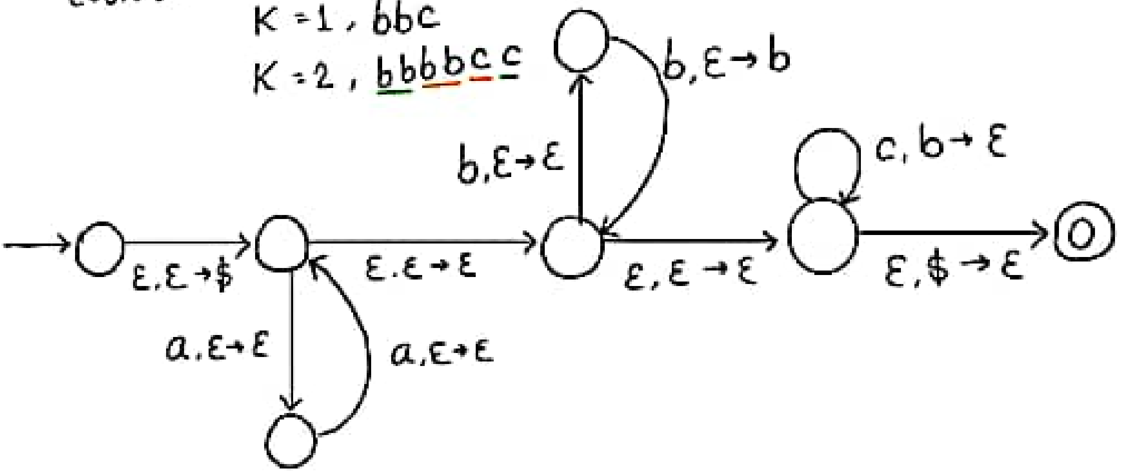
$a^i b^j c^k,\ j = 2k$

$\Rightarrow \underbrace{a^i}_{\text{even } a}\ \underbrace{b^{2k} c^k}$

$k=0,\ \varepsilon$

$k=1,\ bbc$

$k=2,\ bbbbcc$



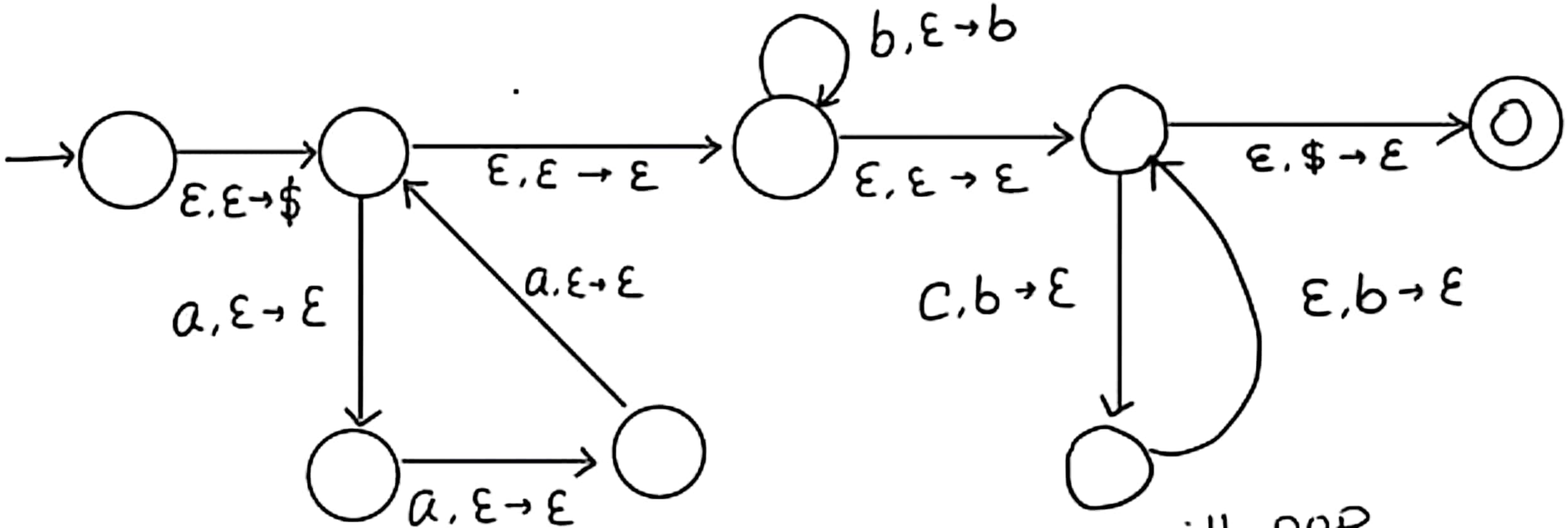
g) $L = \{w \in \{a, b, c\}^*: a^i b^j c^k$, where i is multiple of three, $j = 2k$ and $i, j, k \geq 0\}$.

Same as the previous question, an alternate way to handle j=2k

$a^i b^j c^k$

$\Rightarrow a^i b^{2k} c^k$

$K = 2,$ — bbbbcc



one c will pop
two bs

h) $L = \{w \in \{a, b, c\}^*: a^i b^j c^k$, where k is odd, $i = 2j$ and $i, j, k \geq 0\}$.

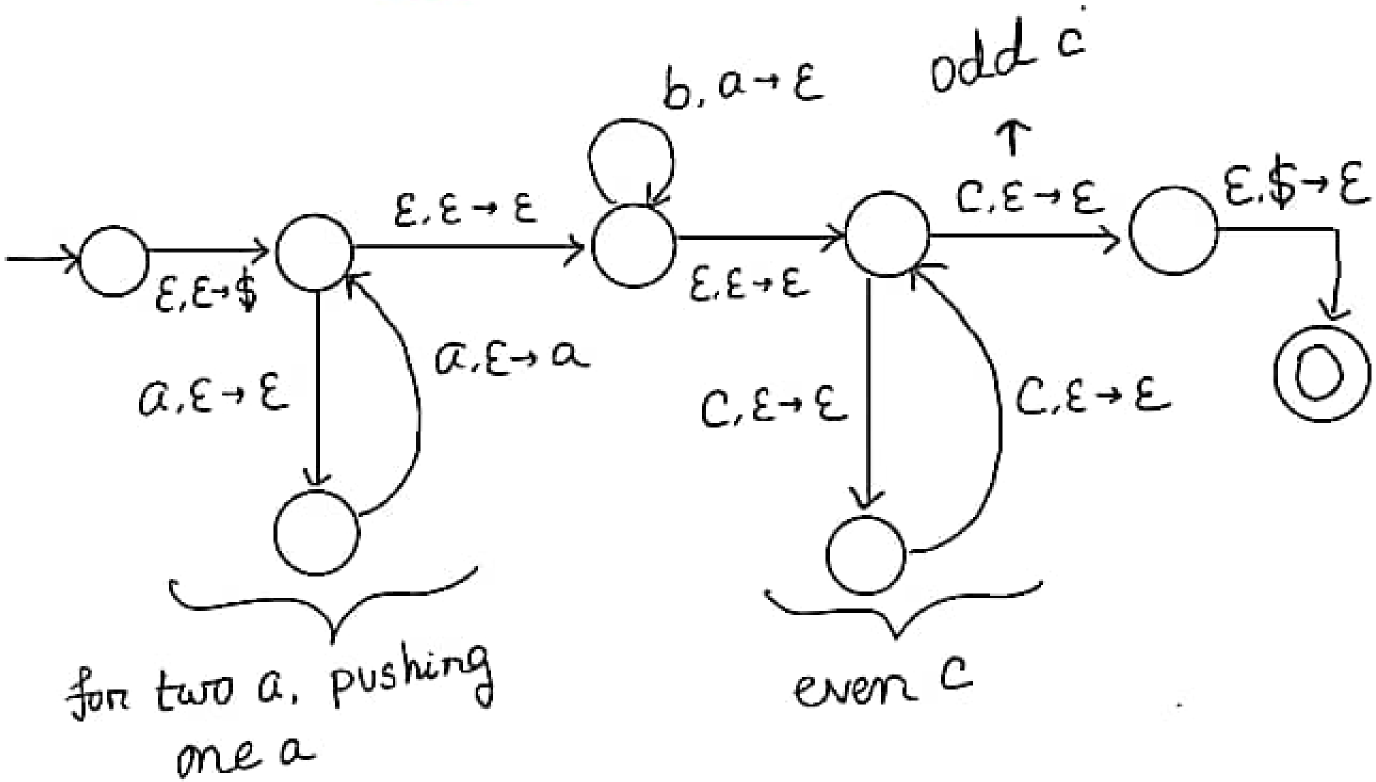
$a^i b^j c^k$

$\Rightarrow \underbrace{a^{2j} b^j}\ \underbrace{c^k} \rightarrow$ odd #K

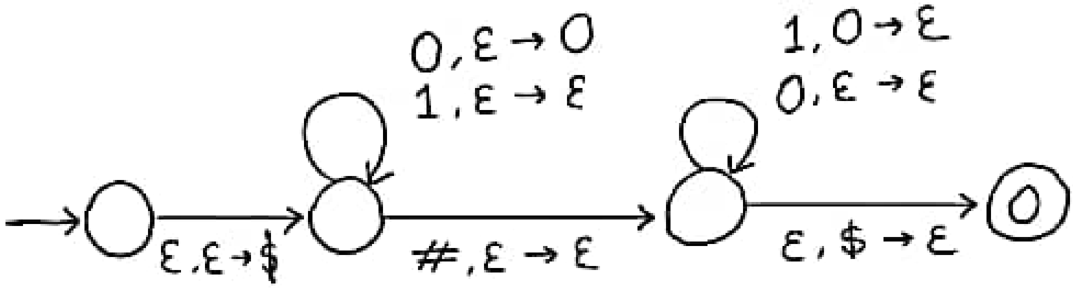
$\hookrightarrow j = 0,\ \varepsilon$

$j = 1,\ aab$

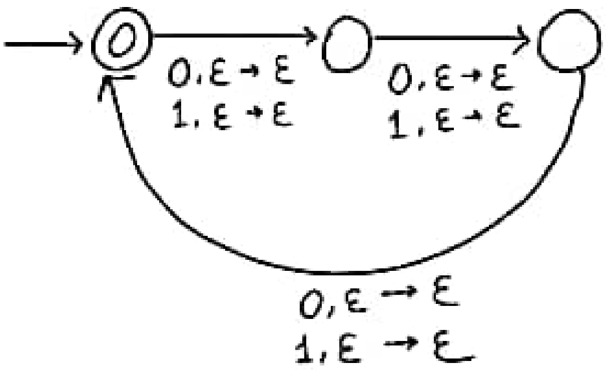
$j = 2,\ aaaabb$



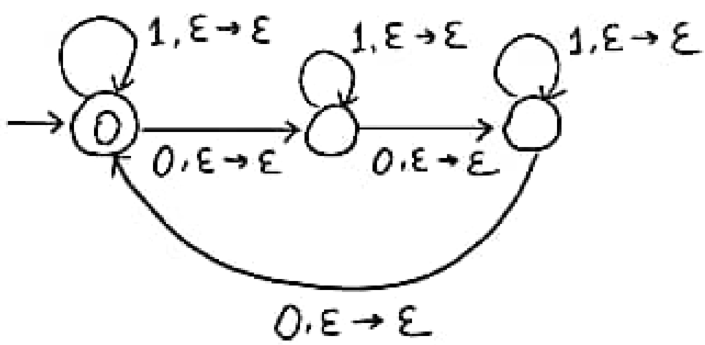
i) Let $\Sigma = \{0, 1, \#\}$. $L = \{ w_1\#w_2 \mid$ number of 0s in w_1 is equal to number of 1s in $w_2\}$



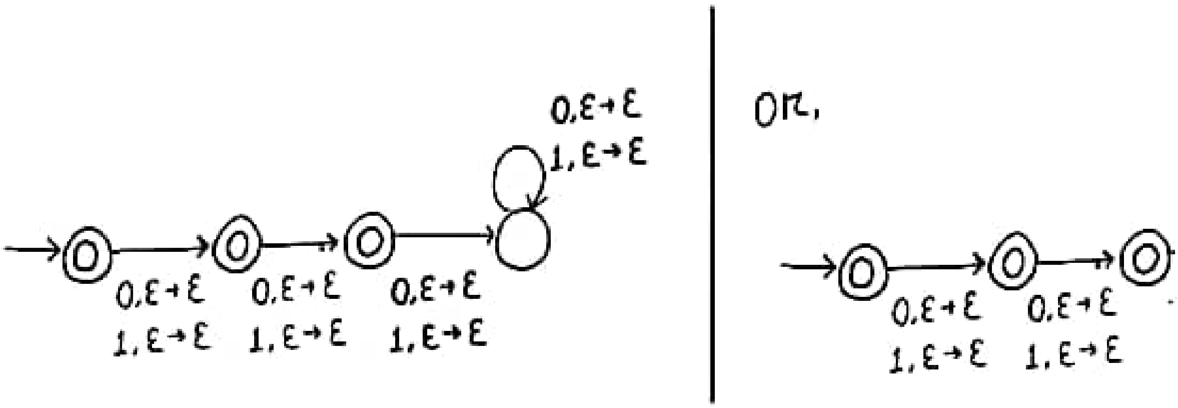
j) $L = \{w \in \{0,1\}^*$: length of w is a multiple of three$\}$ [Hint: Recall what kind of language L is.]



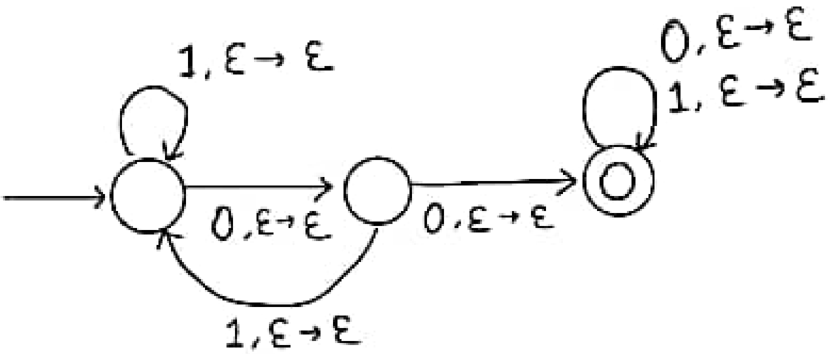
k) $L = \{w \in \{0,1\}^*$: number of 0s in w is a multiple of three$\}$ [Hint: Recall what kind of language L is.]



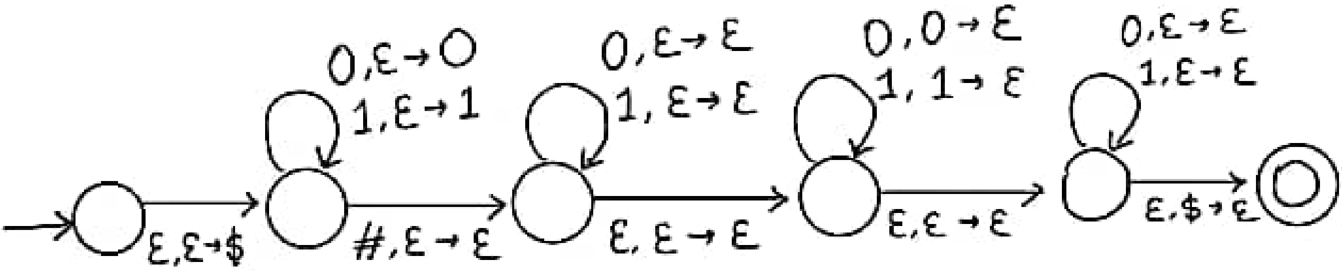
l) L = {w ∈ {0,1}*: length of w is at most two.} [Hint: Recall what kind of language L is.]



m) L = {w ∈ {0,1}*: w contains 00 as a substring}. Construct a PDA for L. [Hint: Recall what kind of language L is.]



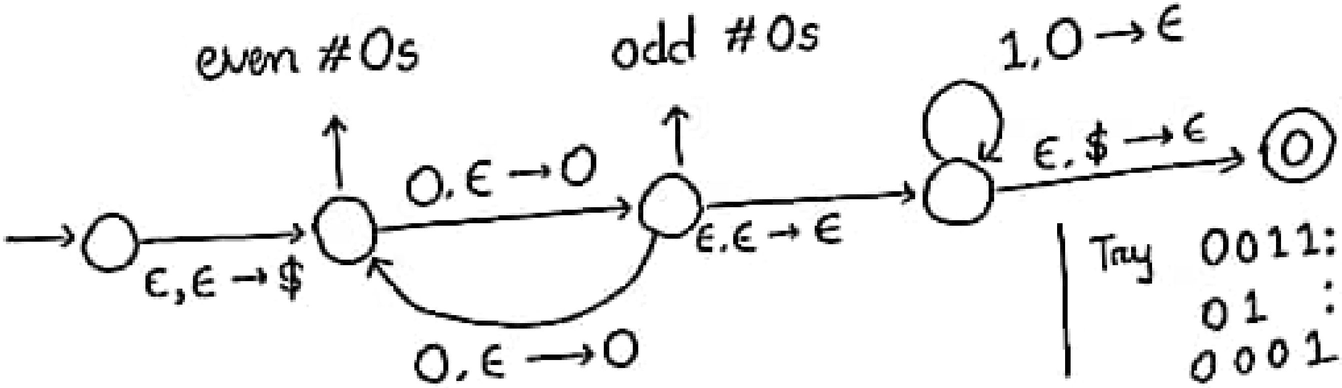
n) L = { w#x : w, x ∈ {a, b}* and x contains w^R as a substring}. [Recall: For a string w, w^R denotes w in reverse order.]



Pushdown Automata (PDA)

we will have same amount of 0s & 1s and n is odd

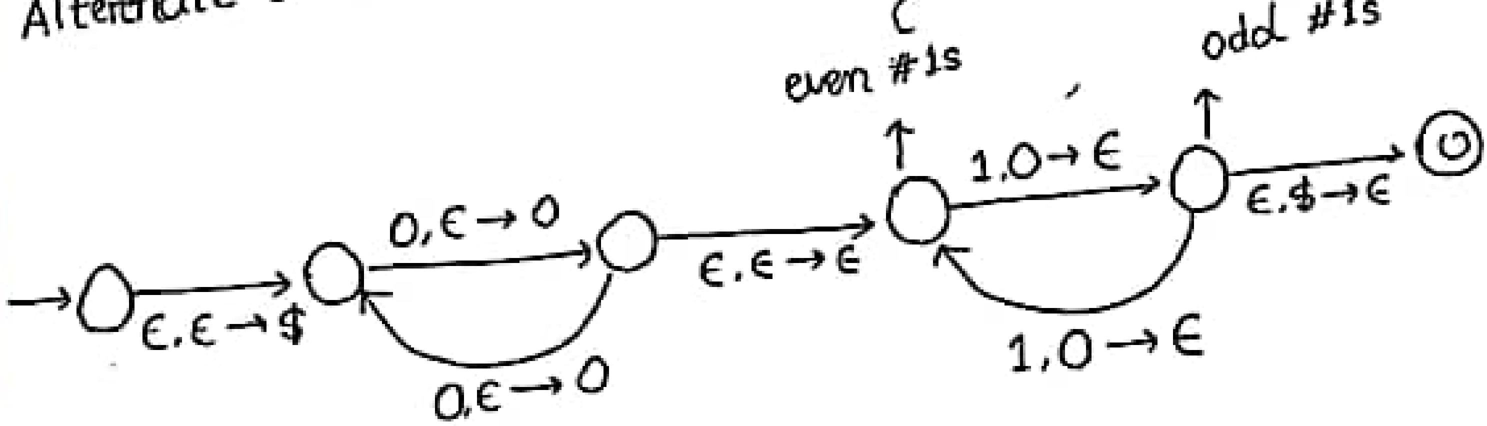
$L = \{\omega \in \{0,1\}^* : \omega = \underbrace{0^n 1^n}_{}, \text{ where } n \text{ is odd}, n \geq 0\}$



Try 0011: ✗
01 : ✓
0001: ✗

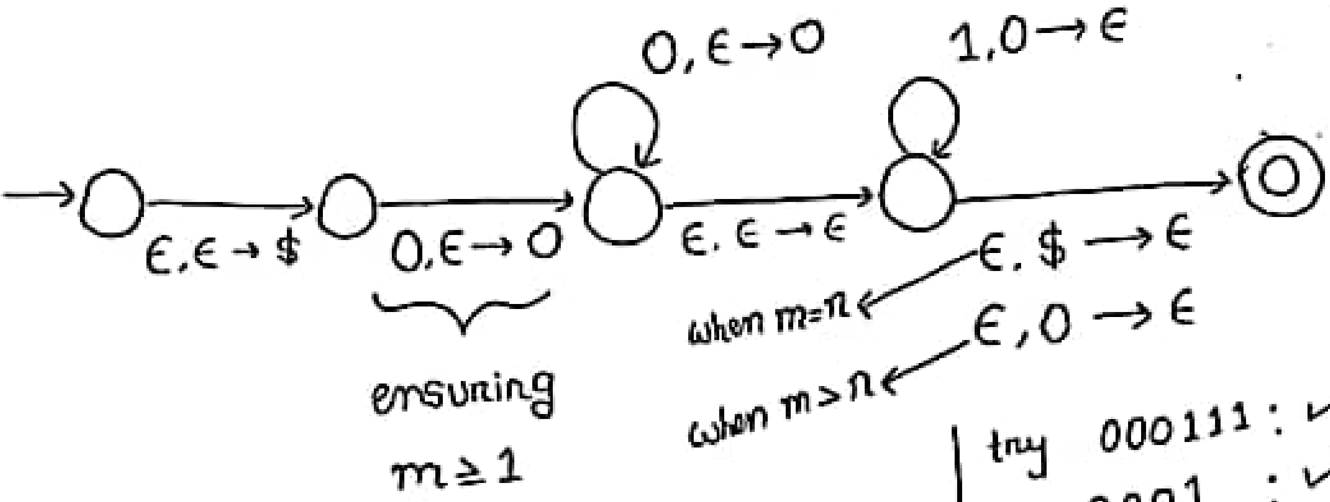
Alternate solution: #0s = #1s, then their parity will be same. However, you can do this as well.

Previously, we simply counted the # 1s, which was enough. Because, if



* what if $L = \{\omega \in \{0,1\}^* : \omega = 0^m 1^n$, where m and n are odd.$\}$

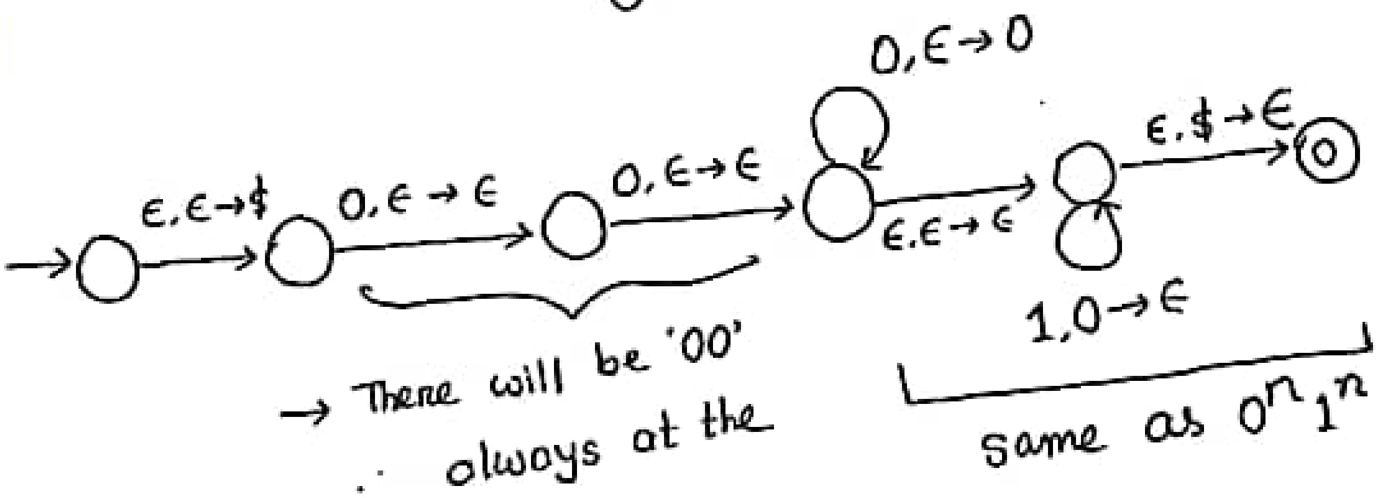
$L = \{ \omega \in \{0,1\}^* : \omega = 0^m 1^n, \text{ where } \boxed{m \geq 1, n \geq 0,} \text{ and } m \geq n \}$



try		
000111	:	✓
0001	:	✓
011	:	×
0	:	✓

$\boxed{\text{* what if } m \leq n\,?}$

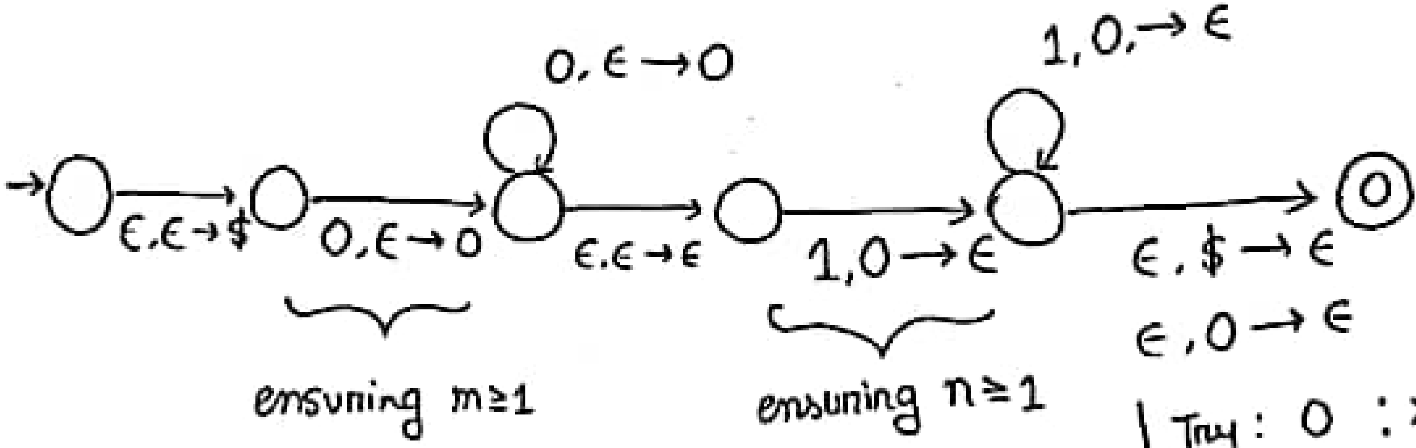
$L = \{ \omega \in \{0,1\}^* : \omega = \underbrace{0^{2+n} 1^n}_{0^2 0^n 1^n}, \text{ where } n \geq 0 \}$



→ There will be '00' always at the beginning.

→ No need to push them in the stack.

$L = \{\omega \in \{0,1\}^* : \omega = 0^m 1^n, \text{ where } \boxed{m, n \geq 1} \text{ and } m \geq n\}$

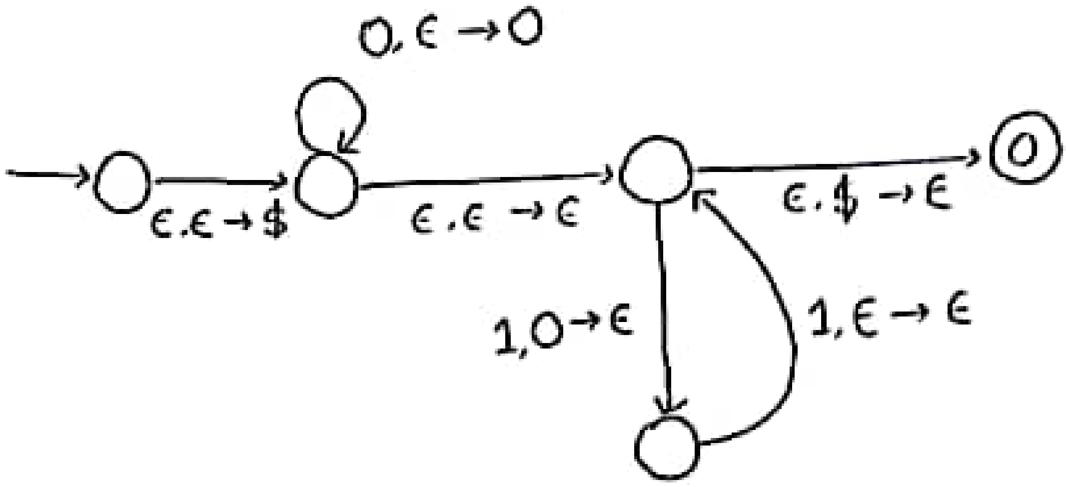


Try: 0 : ✗
01 : ✓
011 : ✗
001 : ✓

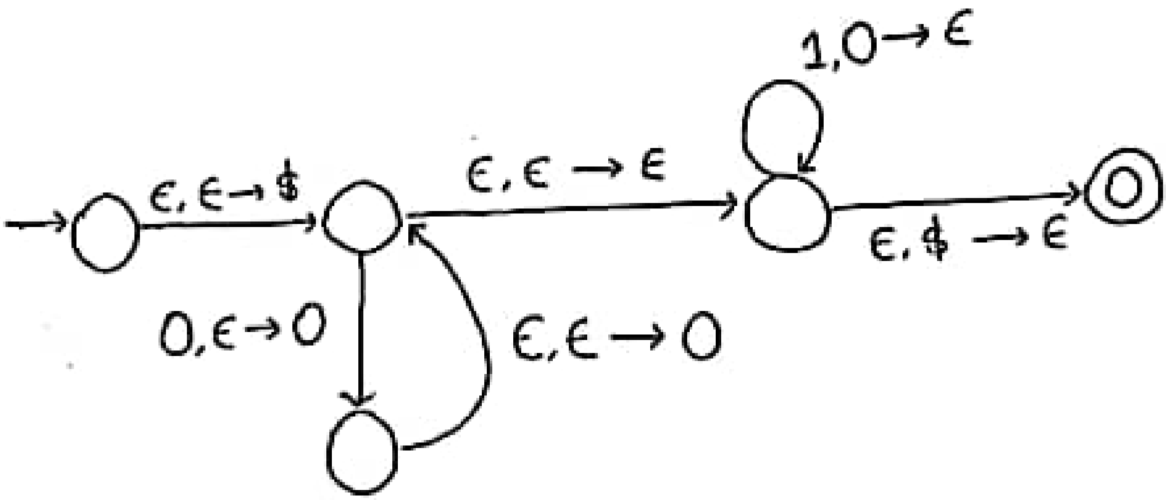
compare the problem with

$L = \{\omega = 0^m 1^n, \text{ where } \boxed{m \geq 1, n \geq 0} \text{ and } m \geq n\}$

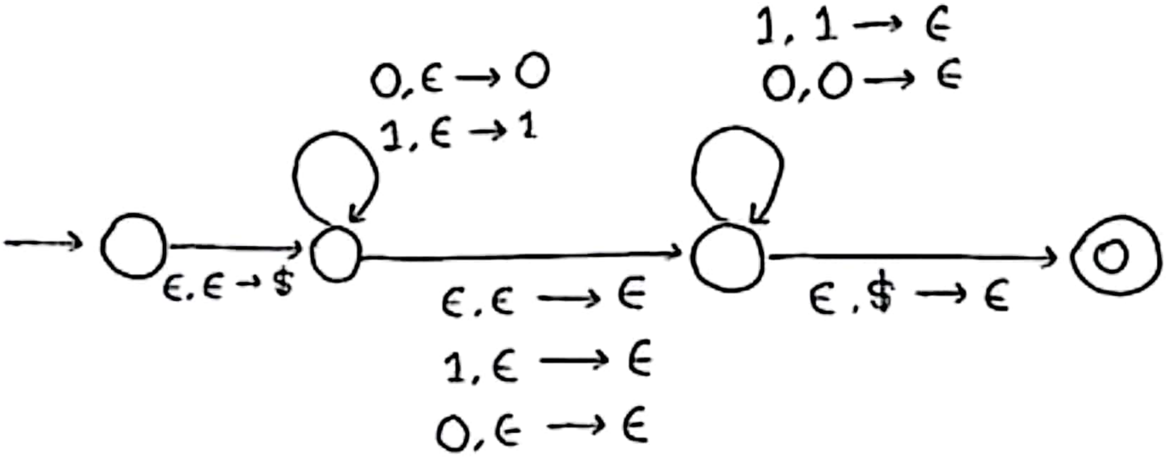
$L = \{\omega \in \{0,1\}^*: \omega = 0^n 1^{2n}, \text{ where } n \geq 0\}$



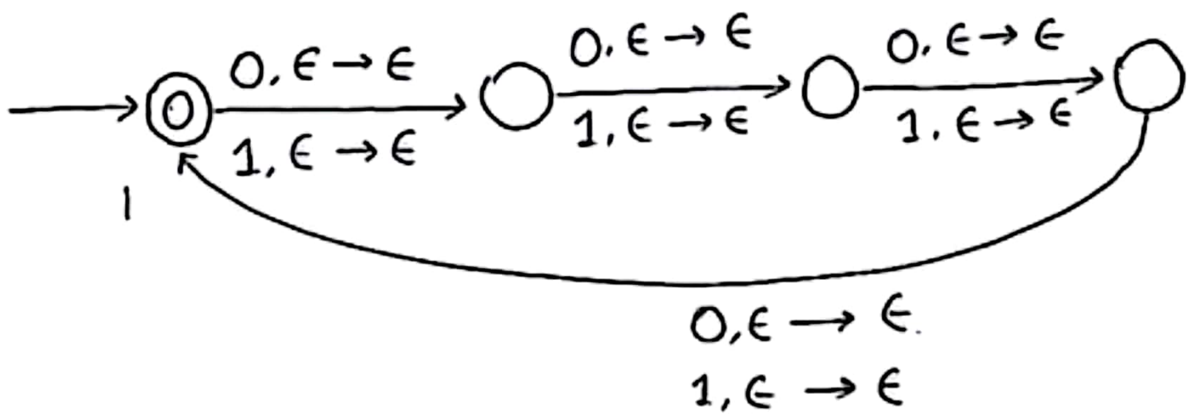
Alternate solution:



$L = \{ w \in \{0,1\}^* : w \text{ is a palindrome.} \}$



$L = \{ w \in \{0,1\}^* : w \mid \text{the length of } w \text{ is multiple of four} \}.$



Since the language is regular, we don't have to use the stack. If you use stack, it is also fine. 'Use the stack' means, pushing and popping element from the stack.

$L1$ = #1s in ω is multiple of 3.

$L2$ = ω contains even #0s

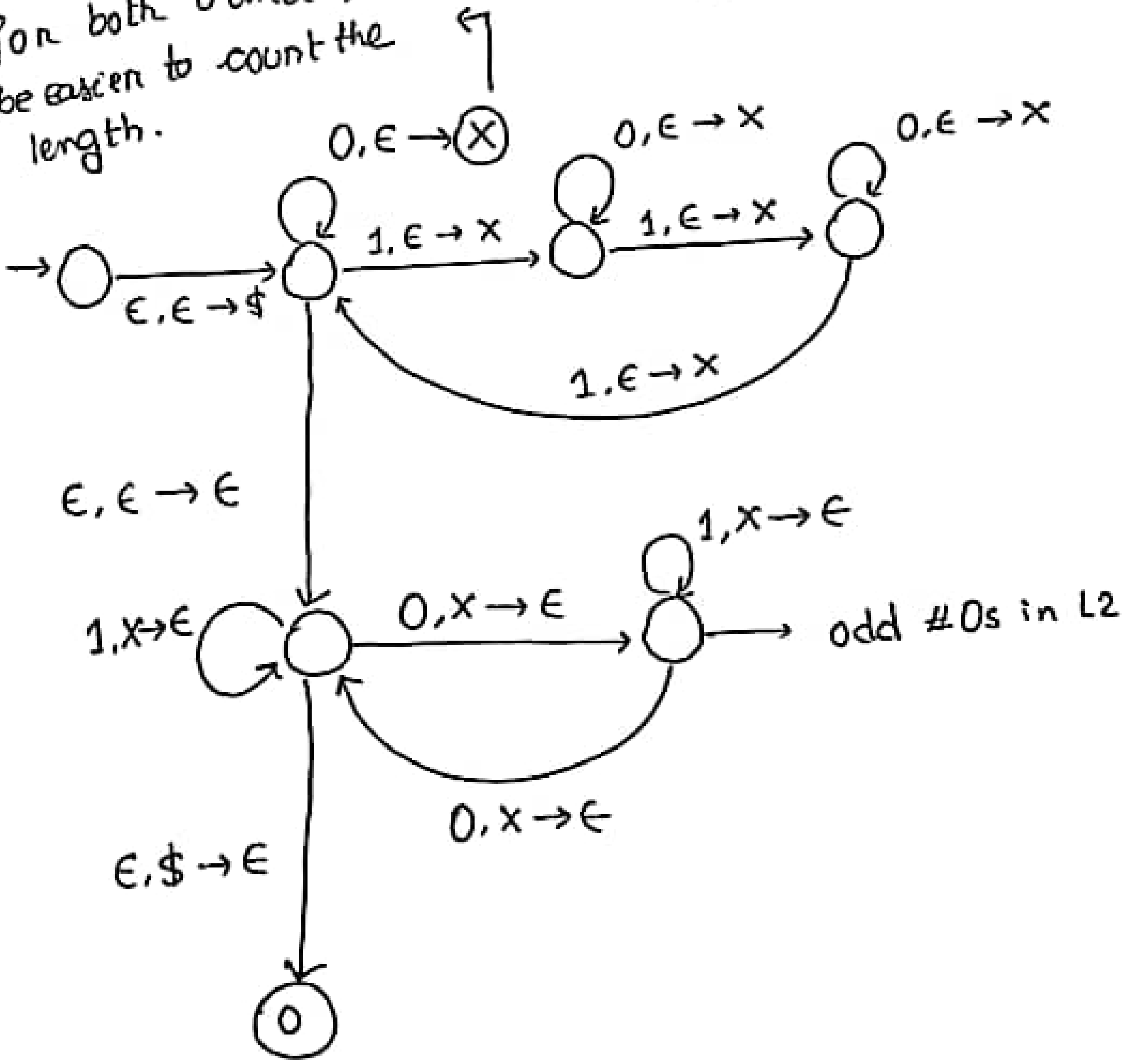
$L = w : \omega = uv$, where $u \in L1$, $v \in L2 \rightarrow$ regular

$|u| = |v| \rightarrow$ non regular

↓

we will use stack to count the lengths are equal

Inserting a common element for both 0 and 1, it will be easier to count the length.

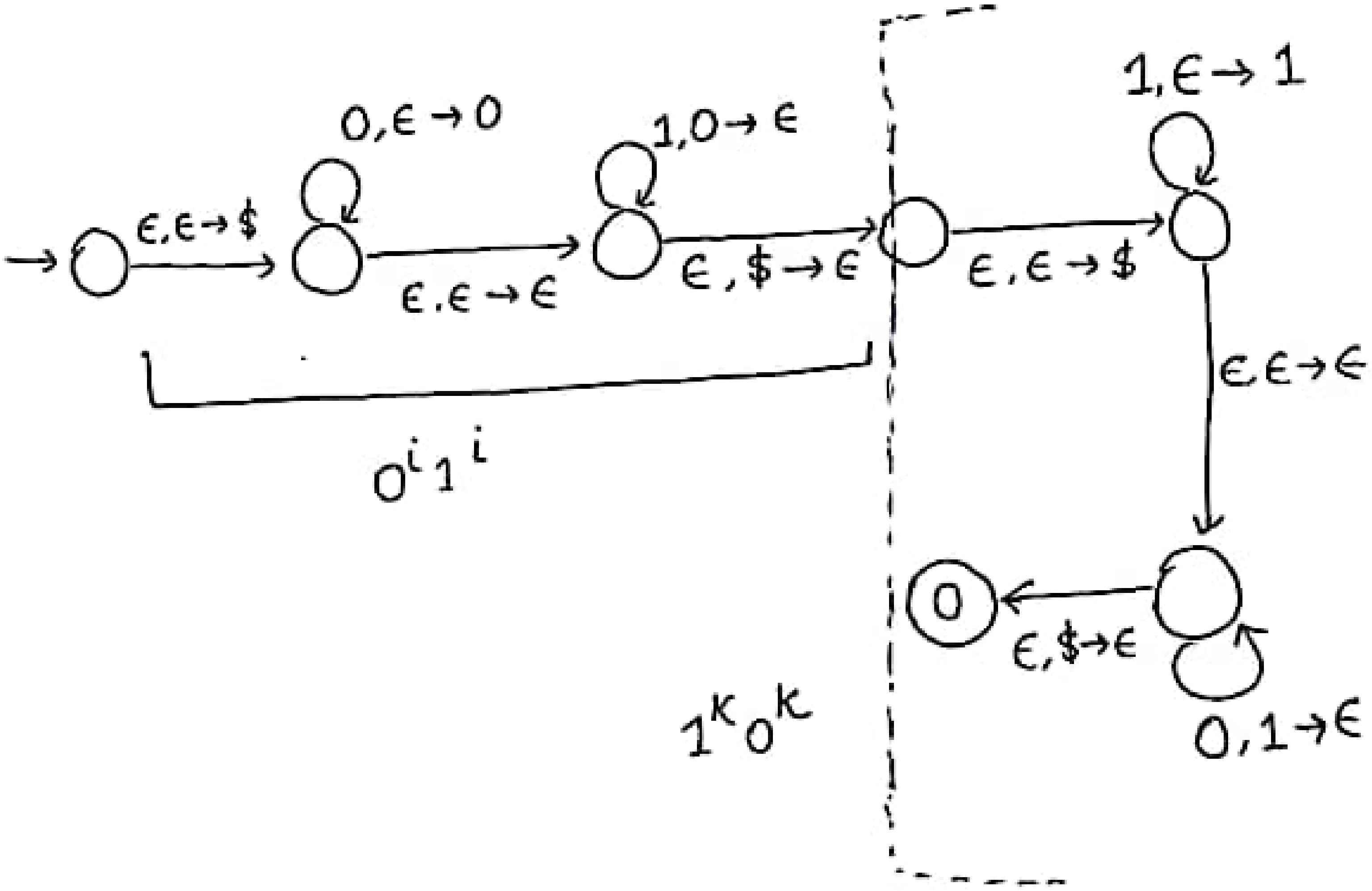


$L = \{ \omega \in \{0,1\}^* : \omega = 0^i 1^j 0^k \mid j = i+k \text{ and } i,k \geq 0 \}$

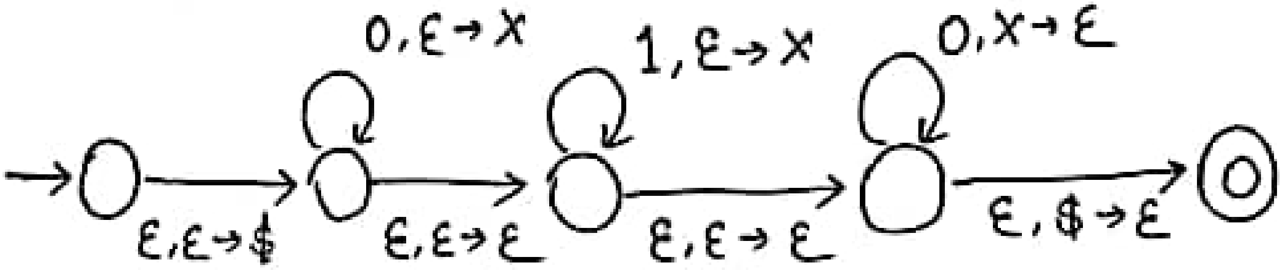
$= 0^i 1^j 0^k$

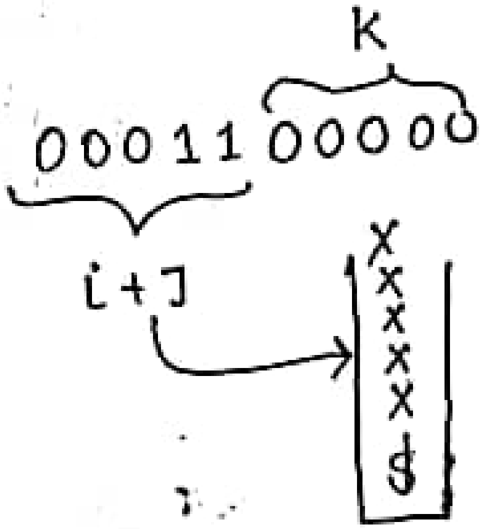
$= 0^i 1^{i+k} 0^k$

$= \underbrace{0^i 1^i} \underbrace{1^k 0^k}$



$L = \{ w \in \{0,1\}^* : w = 0^i 1^j 0^k$, where $i + j = k$ and $i, j \geq 0 \}$

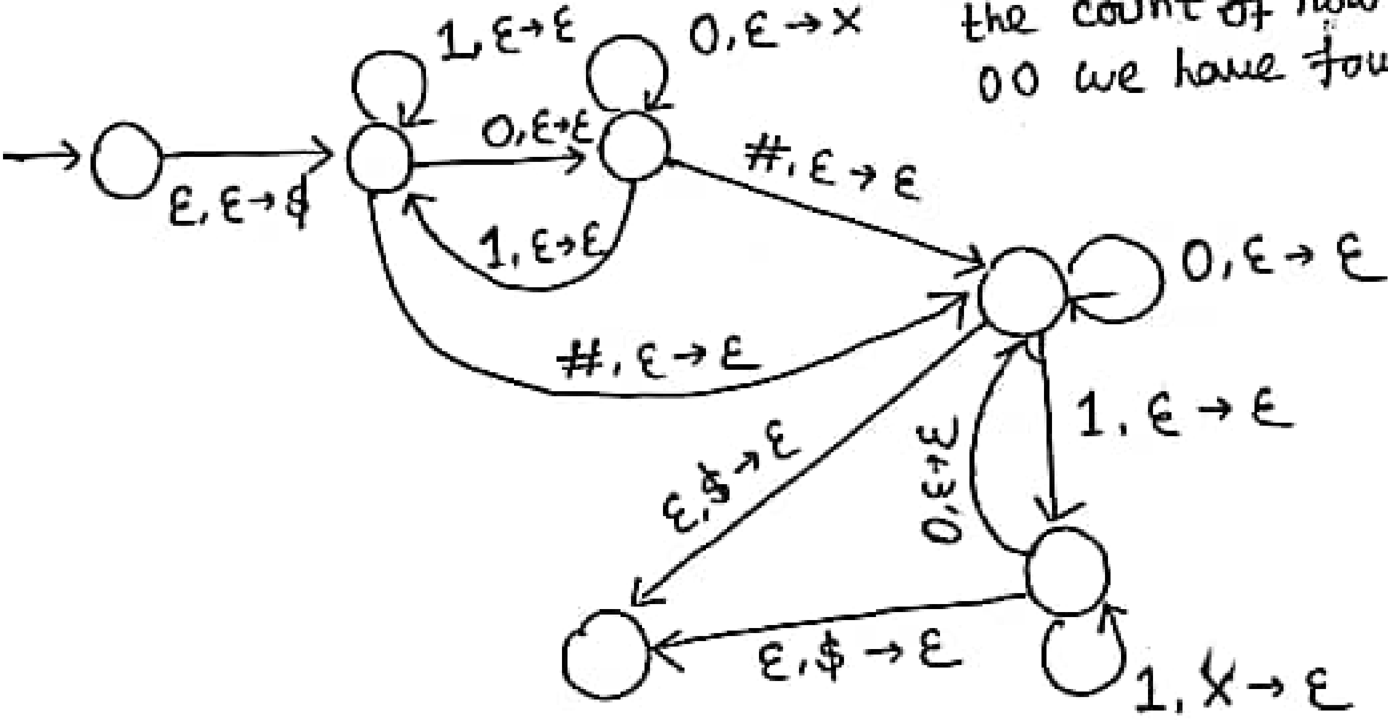




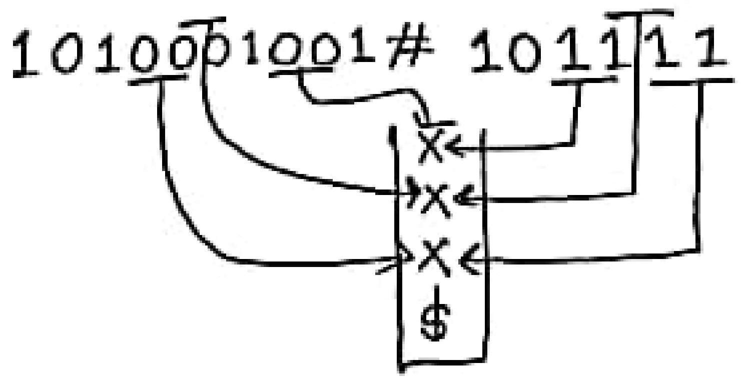
$L = \{ w_1, w_2 \in \{0,1\}^* : w_1 \# w_2$: number of 00 substrings in w_1 is equal to number of 11 substrings in w_2.$\}$

* note, 000 has two 00 as a substring.

if we ~~see~~ find 00, then insert X to keep the count of how many 00 we have found.



if we find 11 then pop X from the stack.



$L = \{ \omega \overline{\omega}^R : \omega \in \{0,1\}^* \}$

for example, $\omega = 0100$

complement of $\omega = \overline{\omega} = 1011$

reverse of $\overline{\omega} = \overline{\omega}^R = 1101$

Doing $\omega \to \omega^R \to \overline{\omega}^R$ will be same thing

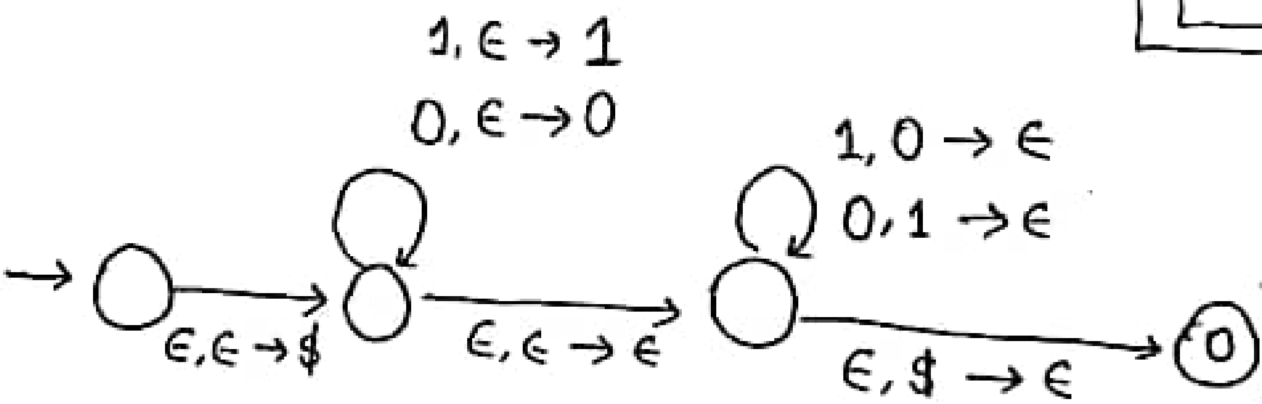
$\omega \overline{\omega}^R = 01001101$

this is even length palindrome

first think how to solve : $\omega\omega^R = 01000010$

so if we complement the part

$\omega \overline{\omega}^R = 01001101$



Alternate solution.