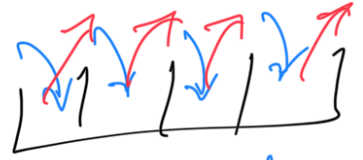


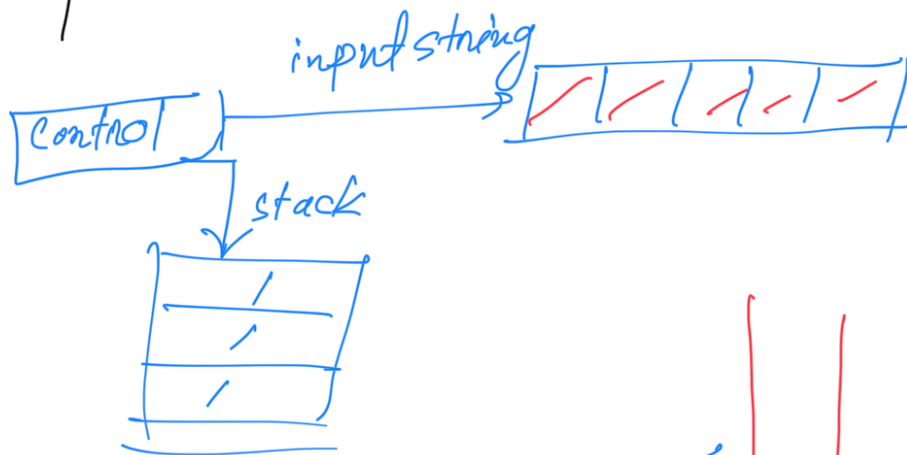
Pushdown Automata

* This is like NFA but has additional component \rightarrow Stack.

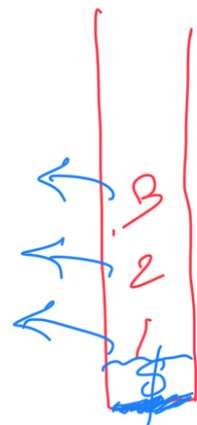


- \rightarrow provides additional memory beyond the finite memory of automata.
- \rightarrow Allows to recognize some non-regular languages.

* Pushdown automata is equivalent in power to CFG.



Indication of the bottom of the stack $\Rightarrow \text{\$}$

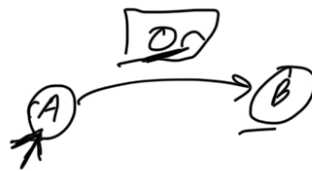


write symbol on the stack \Rightarrow push

Read symbol from the stack \rightarrow pop



0, 1



\rightarrow NFA/DFA

format

input symbol, pop symbol \rightarrow push symbol.

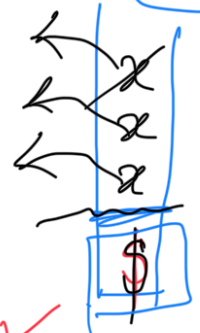


① $L = \{ \omega \in \{0,1\}^* : \omega = 0^n 1^n, n \geq 0 \}$



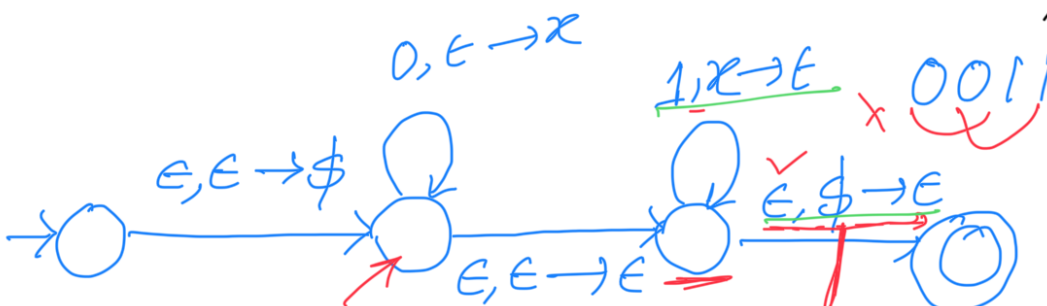
$\epsilon \sim$
01✓
0011✓
~~000111~~

0011



000111✓

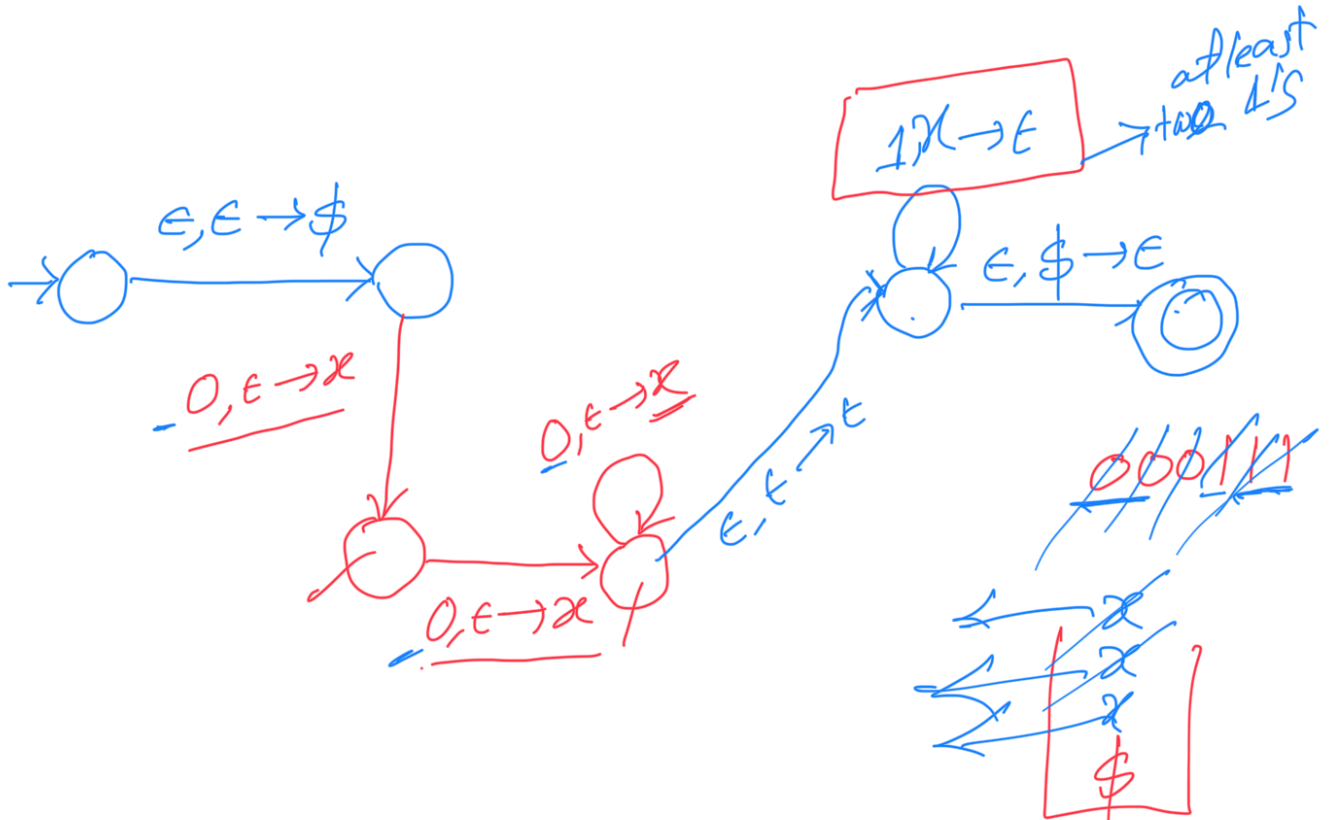
\rightarrow 001. X



~~1, 1~~
 $\left[\begin{array}{c} x \\ 1, \$ \end{array} \right]$

$\checkmark \epsilon \checkmark$
 $\checkmark 01 \checkmark$
 $\checkmark 0011 \checkmark$
 $\checkmark 000111 \checkmark$

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



~~000111~~
 $\left[\begin{array}{c} x \\ x \\ x \\ x \\ \$ \end{array} \right]$

\overleftarrow{xxx}
 \overrightarrow{xyz}

$\left[\begin{array}{c} 1 \\ 1 \end{array} \right]$

$\left[\begin{array}{c} z \\ y \\ x \end{array} \right]$

$\left[\begin{array}{c} x \\ y \\ z \end{array} \right]$

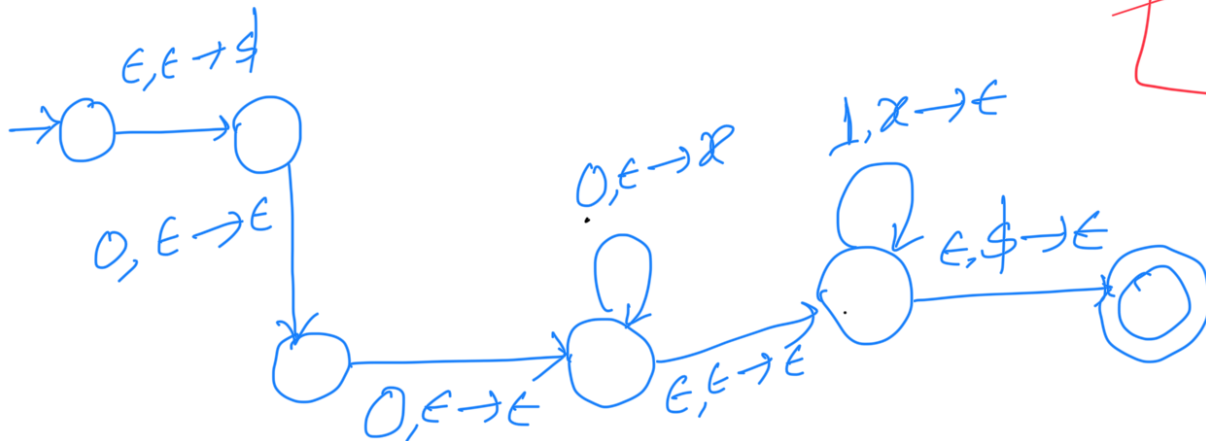
$\sim \dots \sim 0^{n+2} 1^n, n \geq 0$

(11) $L = \{ w \in \{0,1\}^* : w = 0^n 1^m \mid n, m \geq 0 \}$

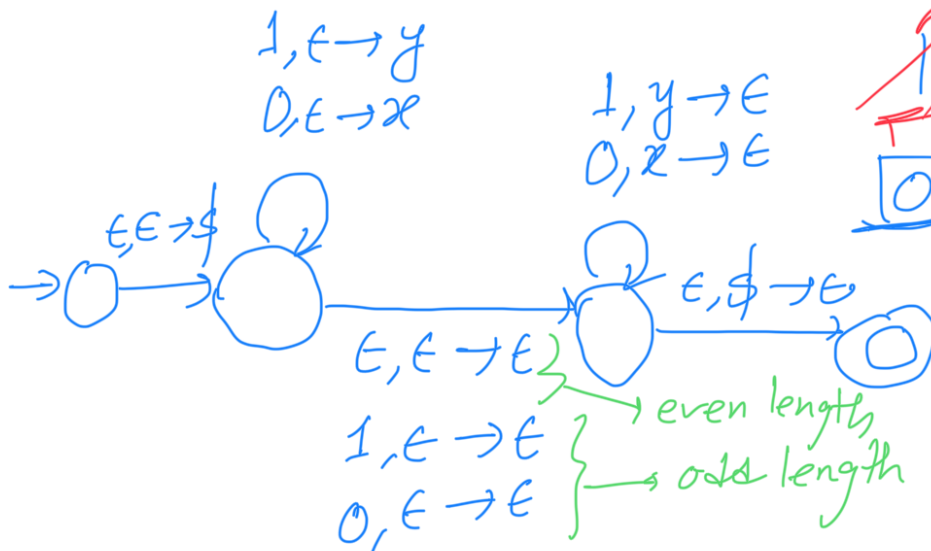
$$\begin{array}{r} \underline{00} \\ \underline{000} \downarrow \\ 0000 \downarrow \downarrow \end{array}$$

Handwritten diagram illustrating a stack data structure. It shows a box divided into two parts: the left part contains 00 and the right part contains $0^n 1^n$. Above the box, $0^{n+2} 1^n$ is written. An arrow labeled "stack??" points to the box. Another arrow labeled "stack" points from the box to the right. A red bracket is under the 00 part, and a green bracket is under the $0^n 1^n$ part.

A handwritten red box containing the symbols x , x , and $\$$ stacked vertically. A red line is drawn across the top of the box, and another red line is drawn across the middle, passing through the first x .



12 $L = \{w \in \{0,1\}^* : w \text{ is a palindrome}\}$

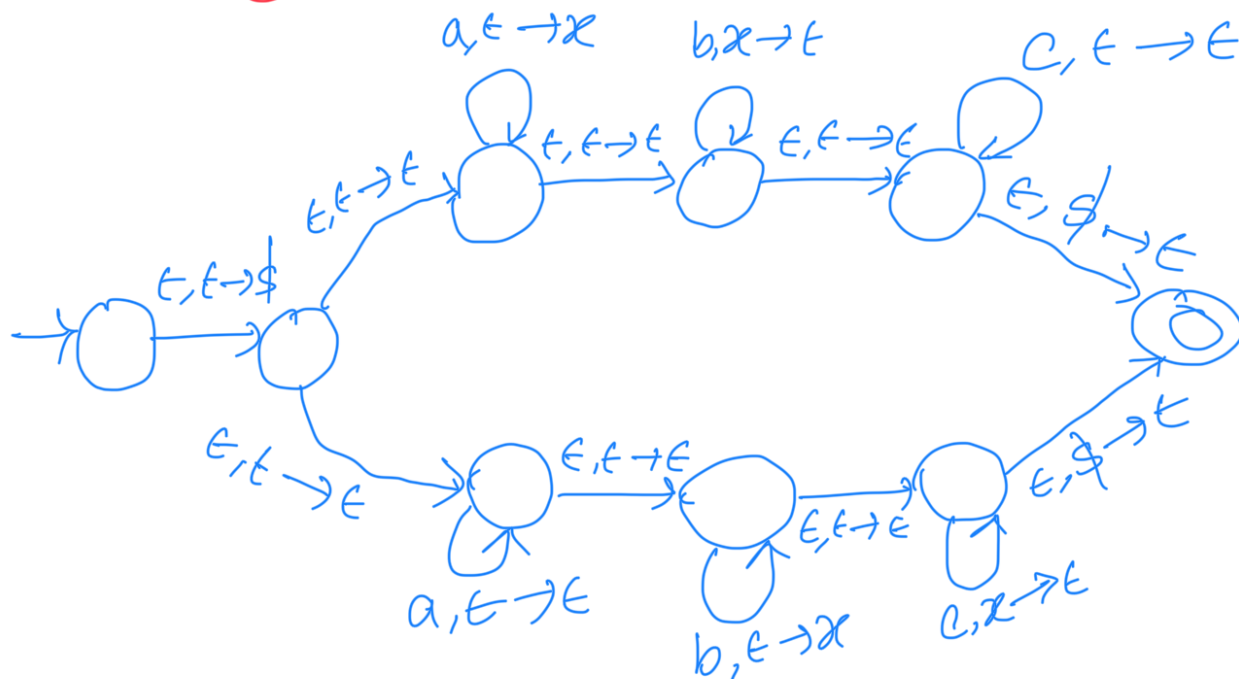


Handwritten diagrams illustrating the steps of a sorting algorithm, likely bubble sort, on a sequence of numbers. The diagrams show the sequence being rearranged through comparisons and swaps, with red and blue lines indicating the movement of elements. The sequence starts as 0, 1, 1, 0 and evolves through several steps, including a final state where the sequence is 0, 1, 1, 0 and the element 0 is highlighted in a green box.

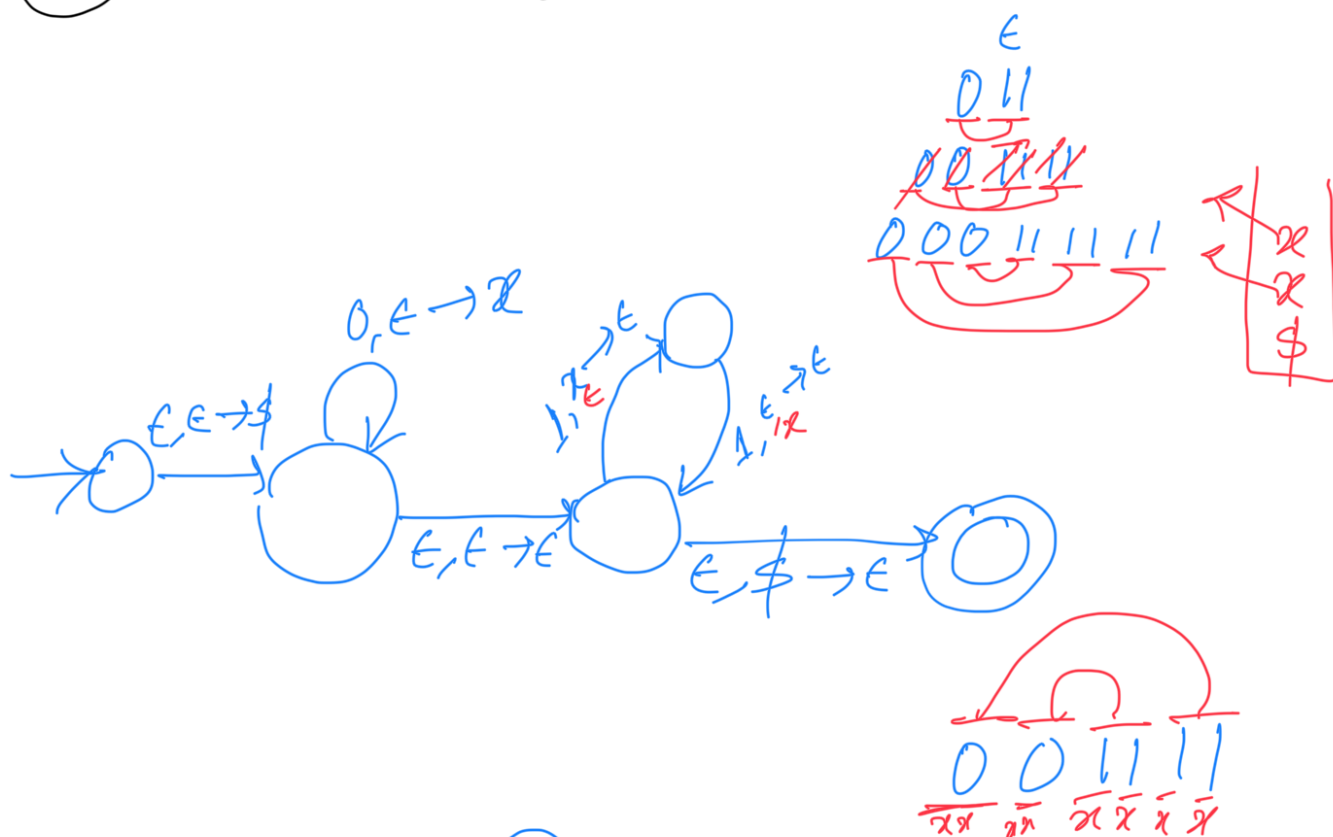
(v) $1 = \sum_{\omega \in \{a, b, c\}^k} \omega = a^i b^j c^k$, where $i+j+k = k$ or $j+k = k$

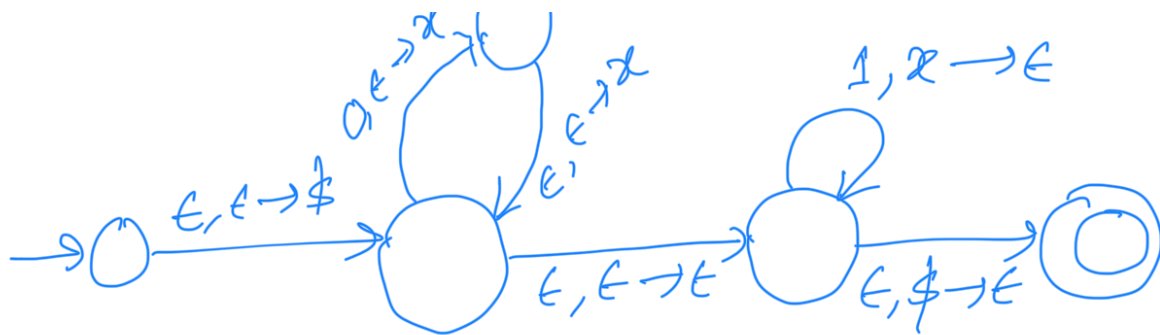
1, J, K, L

a b c



(vi) $L = \{ \omega \in \{0,1\}^* : \omega = 0^n 1^{2n}, n \geq 0 \}$

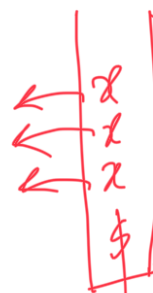
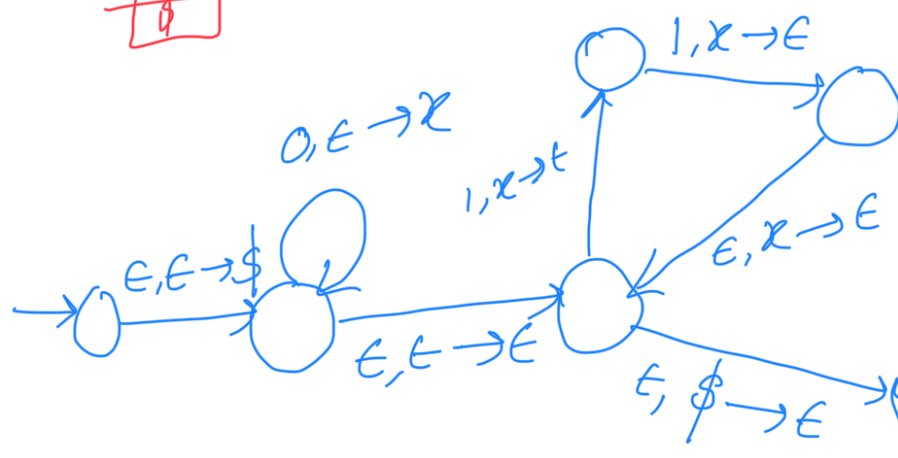




(VII) $L = \{ w \in \{0,1\}^* : w = 0^{3n} 1^{2n}, n \geq 0 \}$

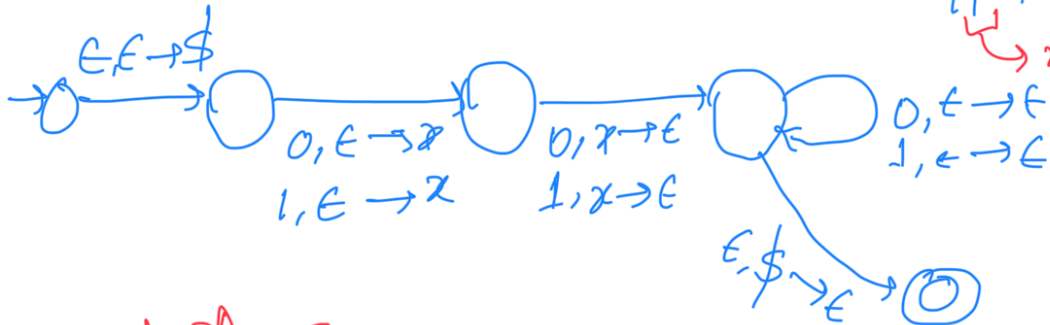


$000, 11$
 ~~000~~ 000 11 ~~11~~



000 11

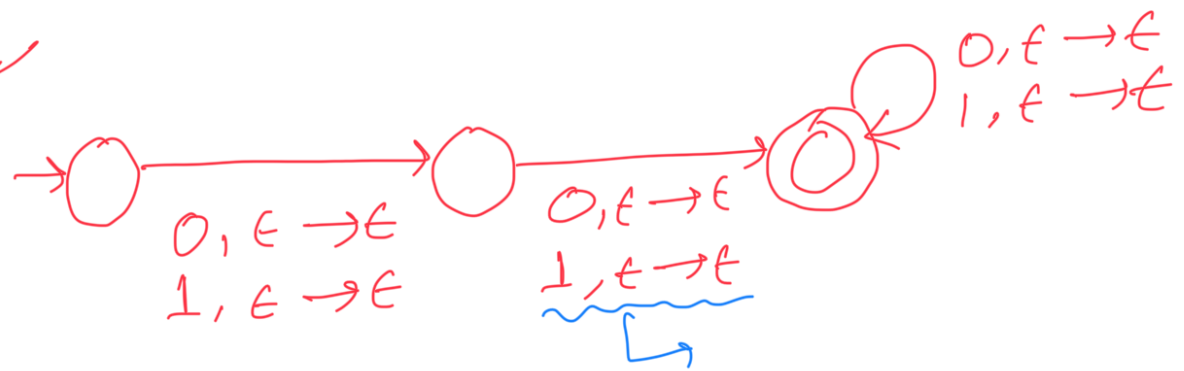
(VIII) $L = \{ w \in \{0,1\}^* : \text{The length of } w \text{ is at least two} \}$



NFA/DFA →



PDA

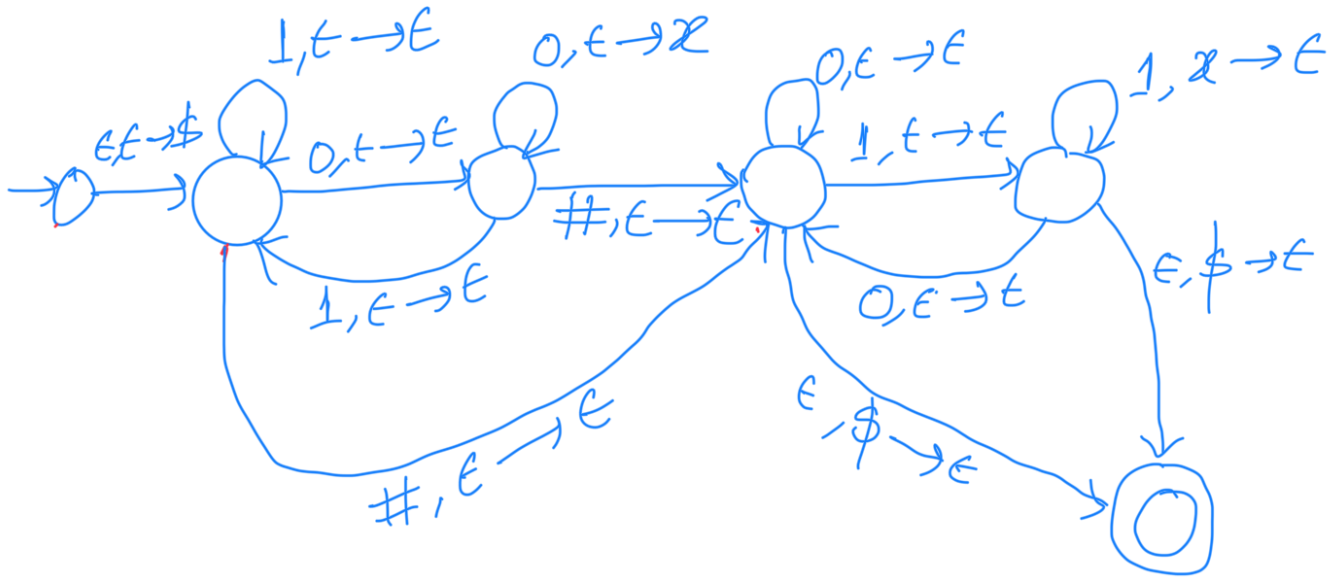


(IX) $L = \{ w_1 \# w_2 : \text{The number of '00' in } w_1 \text{ is the same as the number of '11' in } w_2 \}$

10# ✓
 #10 ✓
 # ✓
 111#000 ✓

001#001011
 000#11011
 011#000

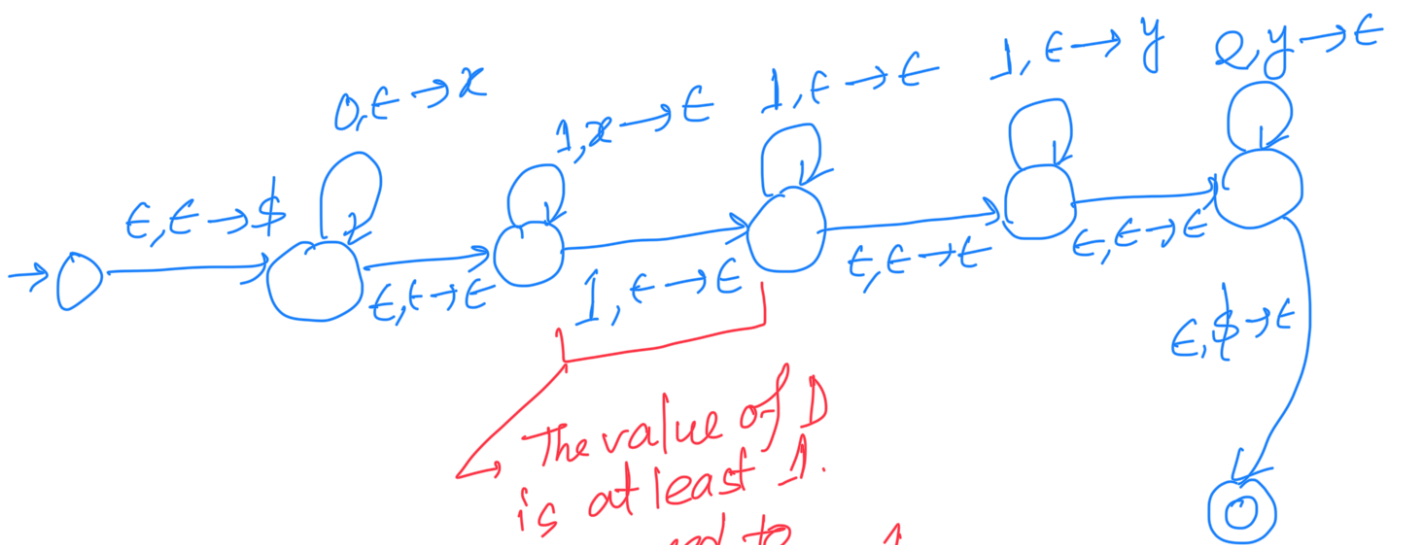
00
 01
 10
 11
 x pop



(X) $L = \{ w \in \{0,1,2\}^* : w = 0^i 1^j 2^k, j > i+k, i, k \geq 0 \}$

$0^i 1^i \boxed{1^D} 1^k 2^k$
 $\hookrightarrow 1 \text{ to } \infty$

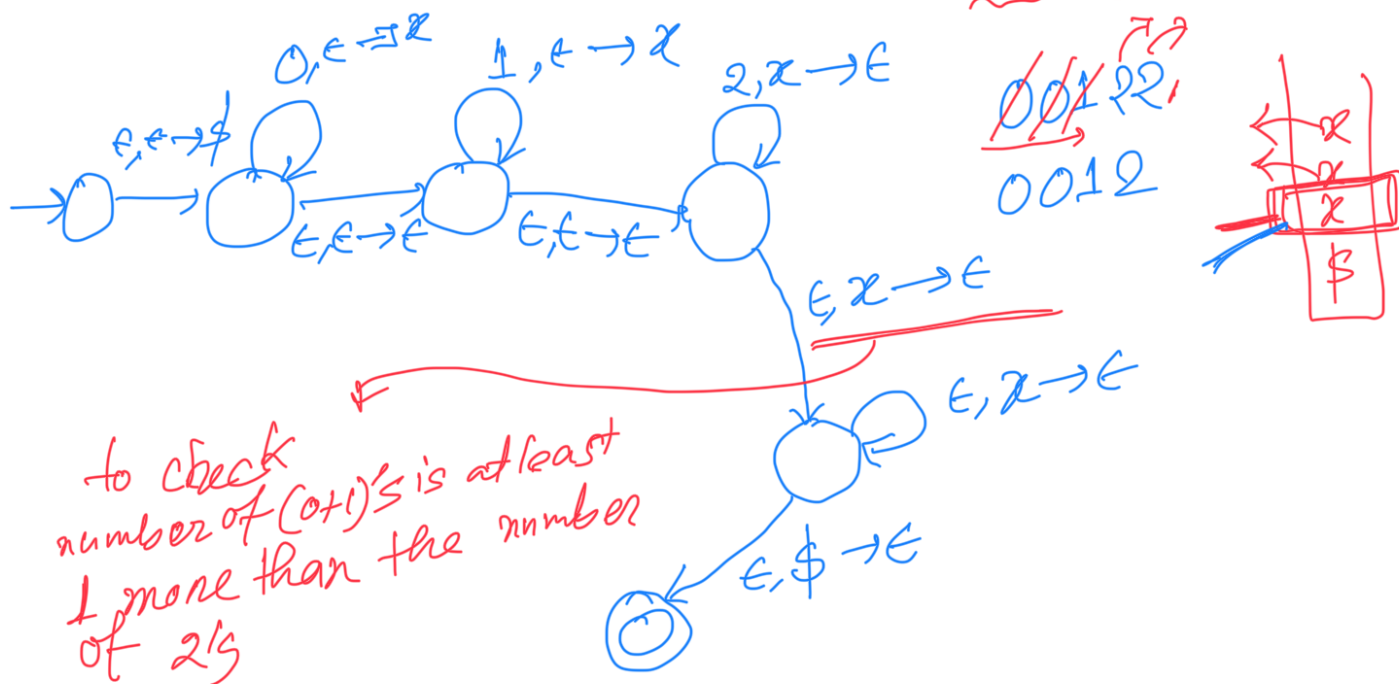
$j = i + k + D$
 $\hookrightarrow 1 \text{ to } \infty$



↖ The value of D is at least 1.
 So we need to read at least one 1 without pushing/popping anything to/from the stack.

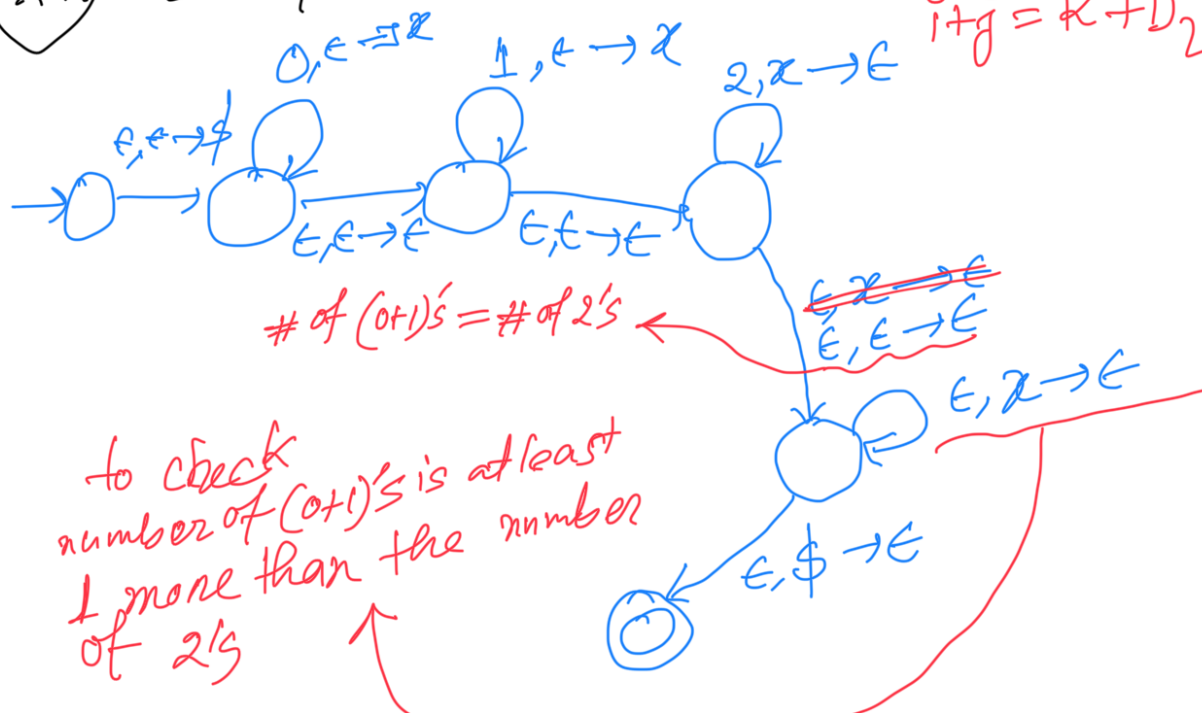
(XI) $L = \{ \omega \in \{0,1,2\}^*, \omega = 0^i 1^j 2^k, i+j > k \}$

$i+j = k + D \rightarrow (1+\infty)$
 $k = i+j - D$



(XII) $L = \{ \omega \in \{0,1,2\}^*, \omega = 0^i 1^j 2^k, i+j \geq k \}$

$i+j = k + D \rightarrow (0+\infty)$



Practice

- ① $L = \{ w \in \{a, b\}^* : w = a^i b^j, \text{ where } i > j, j \geq 0 \}$
- ② $L = \{ w \in \{0, 1, 2\}^* : w = 0^i 1^j 2^k, \text{ where } i + j = k, i, j \geq 0 \}$
- ③ $L = \{ w \in \{0, 1\}^* : 0 \text{ and } 1 \text{ alternates in } w \}$
- ④ $L = \{ w \in \{a, b\}^* : \text{the count of "a" in } w \text{ is a multiple of } 3 \}$