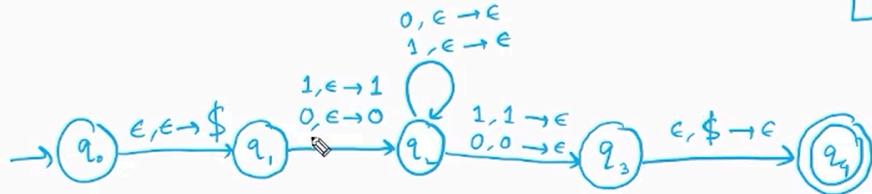


$$\Sigma = \{0, 1\}$$

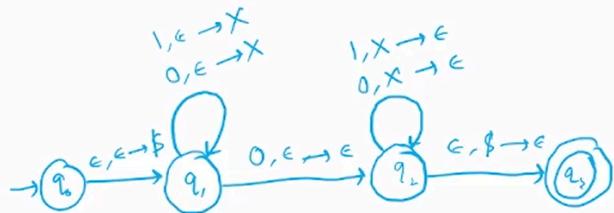
$\{w \in \{0,1\}^* \mid w \text{ starts and ends with the same symbol}\}$

00
/ /
0(01001)
(0)1111000100



$\{w \in \{0,1\}^* \mid w \text{ is of odd length and has 0 as its middle symbol}\}$

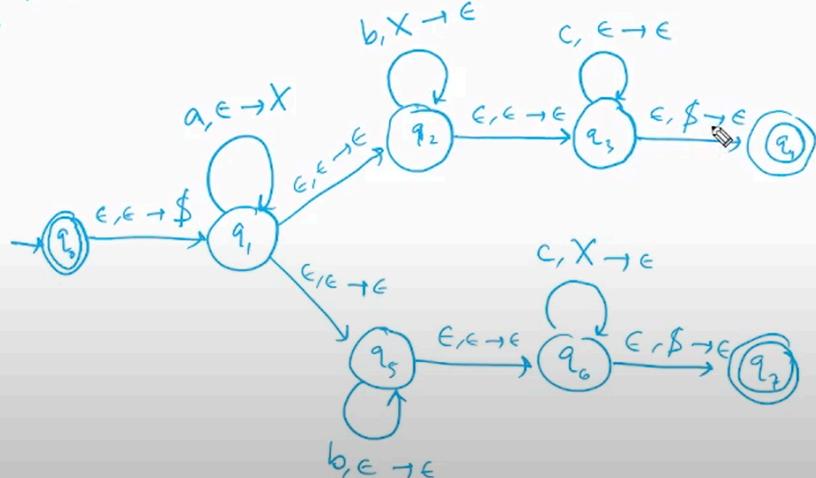
0
100
101000101111111
X X X X X X X X X X X X
\$



$$\Sigma = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k\}$$

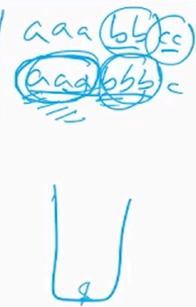
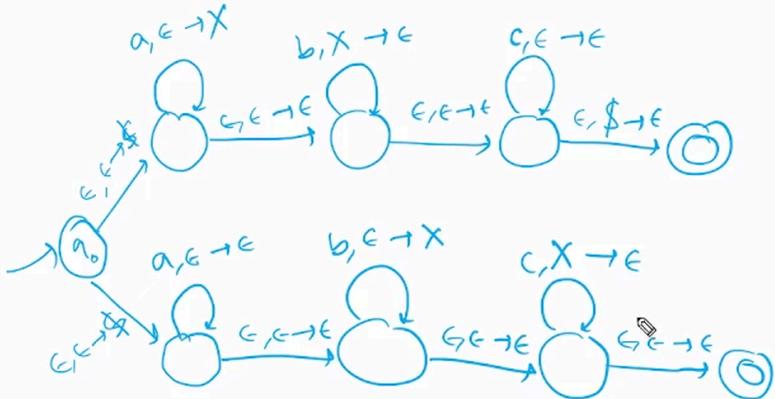
$A = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k\}$ .

aabbcc
<u>aabbcc</u>
aaabbbcc



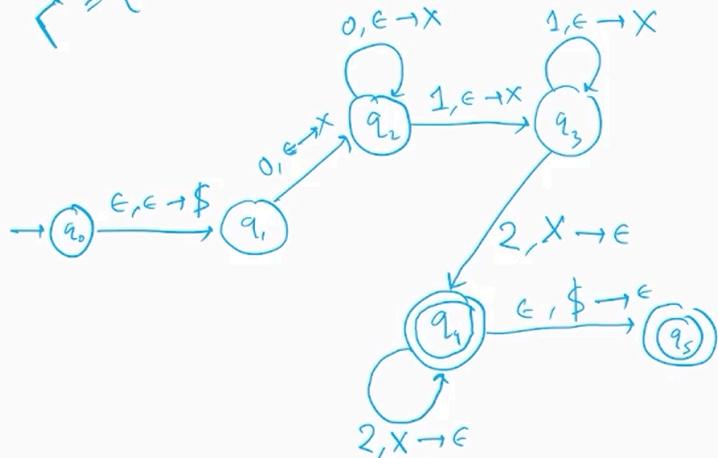
Activa

$$A = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ and either } i = j \text{ or } j = k\}.$$



$$\begin{cases} \Sigma = \{0, 1, 2\} \\ \Gamma = \{\$, X\} \end{cases}$$

$$L = \{0^i 1^j 2^k \mid i+j \geq k \text{ and } i, j, k > 0\}$$



$$0011122222 \quad \textcircled{2}$$



Activate Windows

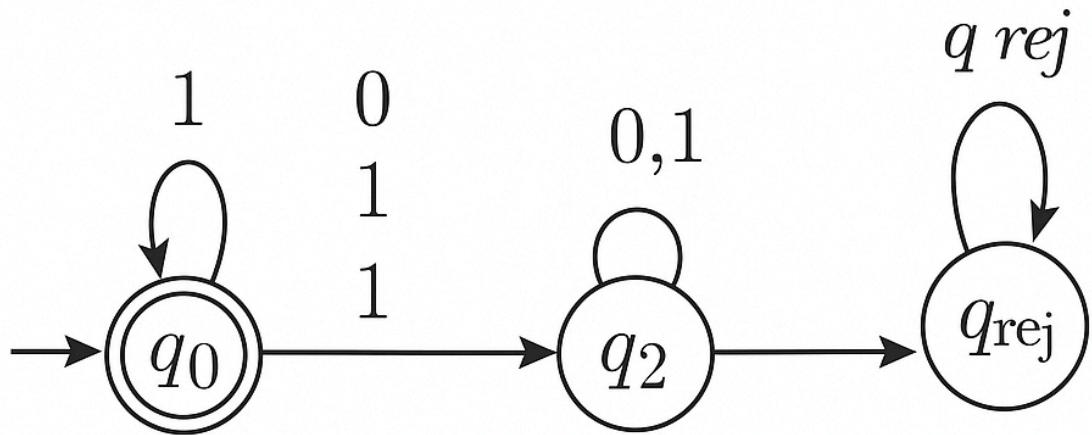
### Problem 5 (CO3): Designing Pushdown Automata (10 points)

$$L_1 = \{w \in \{0, 1\}^* : w \text{ contains at most two } 0\text{s}\}$$

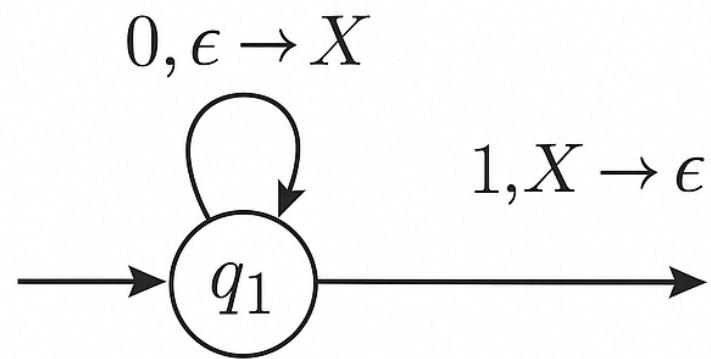
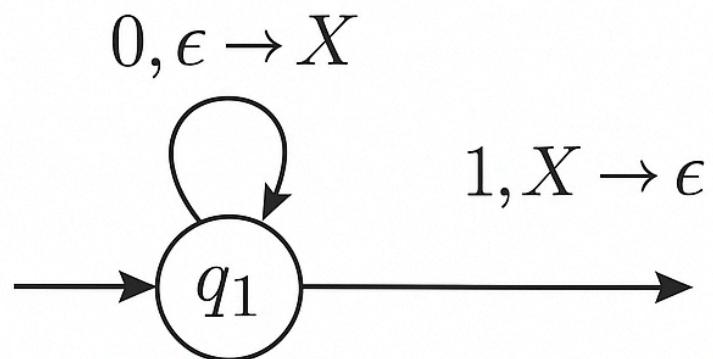
$$L_2 = \{w \in \{0, 1\}^* : w = 0^{2n}1^{3n} \text{ where } n \geq 0\}$$

$$L_3 = \{w\#x : w, x \in \{0, 1\}^* \text{ and } w^R \text{ is a substring of } x\}$$

- (a) Give the state diagram of a pushdown automaton that recognizes  $L_1$ . (3 points)
- (b) Give the state diagram of a pushdown automaton that recognizes  $L_2$ . (3 points)
- (c) Give the state diagram of a pushdown automaton that recognizes  $L_3$ . [Recall: For a string  $w$ ,  $w^R$  denotes  $w$  in reverse order.] (4 points)



(a)



(b)

**(a) PDA for  $L_1 = \{w \in \{0, 1\}^*: w \text{ contains at most two } 0s\}$**

**ধারণা:**

এই PDA-তে stack-এর দ্রবকার নেই, কারণ এটা finite automaton দিয়েই করা যায়। শুধুমাত্র তিনটি 0 পড়লে আমরা reject করব।

**States:**

- $q_0$ : শুরু অবস্থা, এখনো কোনো 0 পড়িনি
- $q_1$ : একটি 0 পড়েছি
- $q_2$ : দুইটি 0 পড়েছি
- $q_{rej}$ : তিন বা তার বেশি 0 পড়লে reject

**Transitions:**

- $q_0 \xrightarrow{0} q_1, q_0 \xrightarrow{1} q_0$
- $q_1 \xrightarrow{0} q_2, q_1 \xrightarrow{1} q_1$
- $q_2 \xrightarrow{0} q_{rej}, q_2 \xrightarrow{1} q_2$
- $q_{rej} \xrightarrow{0,1} q_{rej}$

Accepting states:  $q_0, q_1, q_2$

---

### (b) PDA for

$$L_2 = \{w \in \{0, 1\}^*: w = 0^{2n}1^{3n} \text{ where } n \geq 0\}$$

ধারণা:

আমরা PDA দিয়ে 0-এর সংখ্যা  $2n$  বার এবং পরে 1-এর সংখ্যা  $3n$  বার মেলাবো।

আমরা প্রতি 2টি 0 পড়লে stack-এ একটি  $x$  রাখবো, এবং পরে প্রতি 3টি 1 পড়লে একটি  $x$  pop করবো।

**States:**

- $q_0$ : শুরু ও 0 পড়ার সময়
- $q_1$ : 1 পড়া শুরু হয়েছে
- $q_{acc}$ : গ্রহণযোগ্য অবস্থা

**Transitions:**

- $q_0 \xrightarrow{0, \epsilon \rightarrow X} q_0$  (first 0)
- $q_0 \xrightarrow{0, \epsilon \rightarrow X} q_0$  (second 0 — after each pair one  $x$ )
- $q_0 \xrightarrow{1, X \rightarrow \epsilon} q_1$
- $q_1 \xrightarrow{1, X \rightarrow \epsilon} q_1$
- $q_1 \xrightarrow{1, X \rightarrow \epsilon} q_1$  (after 3 pops one pair consumed)
- Accept if stack is empty

### (c) PDA for

$L_3 = \{w\#x : w, x \in \{0, 1\}^* \text{ and } w^R \text{ is a substring of } x\}$

#### ধারণা:

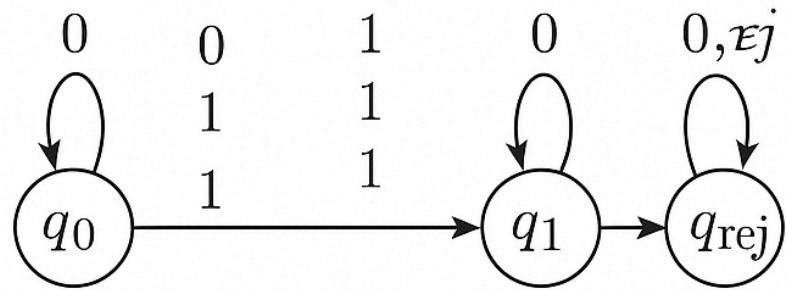
এই PDA-তে আমরা  $w$ -এর রিভার্স  $w^R$  stack-এ রাখব এবং  $\#$ -এর পরে  $x$ -এ  $w^R$  match খুঁজবো sliding window ব্যবহার করে।

#### States:

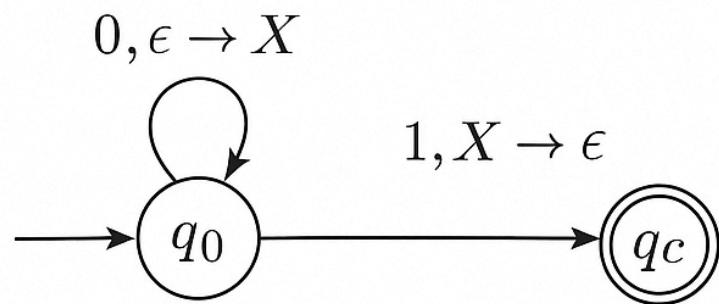
- $q_0$ :  $w$  পড়া এবং stack-এ push করা
- $q_1$ :  $\#$  পেয়েছি
- $q_2$ :  $x$  থেকে substring match খোঁজা
- $q_{acc}$ : সফল match

#### Transitions (ধারণা):

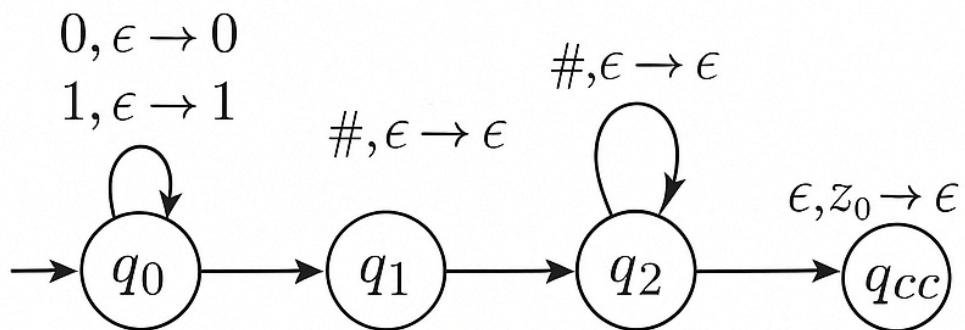
1.  $q_0 \xrightarrow{a, \epsilon \rightarrow a} q_0$ , where  $a \in \{0, 1\}$
2.  $q_0 \xrightarrow{\#, \epsilon \rightarrow \epsilon} q_1$
3. From  $q_1$ , repeatedly try to match top of stack with next input:
  - Start new trial from current position in  $x$ , push all input chars seen so far in a buffer (not formal in PDA but simulated)
  - If match fails, restart
4. If full  $w^R$  matched, stack empty  $\rightarrow$  accept



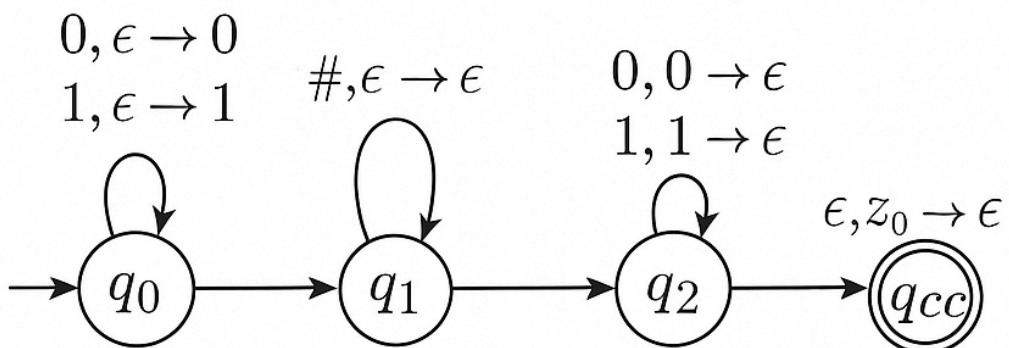
(a)



(b)



(c)



(c)