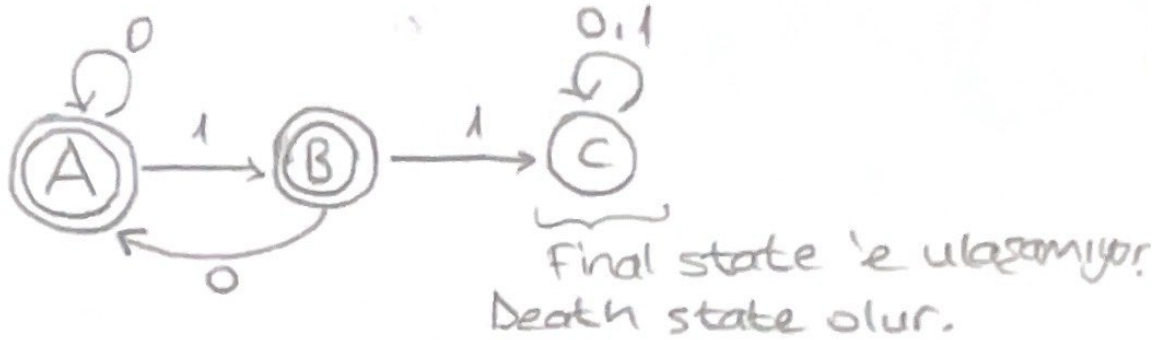


## DFA örnek 1:

Ardışık olmayan tüm stringleri kabul eden DFA

$$A = (Q, \Sigma, \delta, q_0, F)$$



$$Q = \{A, B, C\} \quad \Sigma = \{0, 1\} \quad q_0 = A \quad F = \{A, B\}$$

0100. girdisini okuyalım:

$$\delta(A, 0) = A$$

$$\delta(A, 1) = B$$

$$\delta(B, 0) = A$$

$$\delta(A, 0) = A$$

$$\delta(A, 0) = A$$

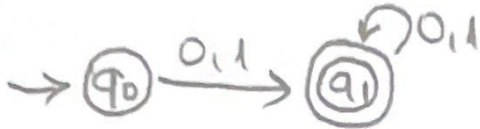
### DFA örnek 2:

0 ve 1'lerden oluşan s uzunluğundaki string-  
leri temsileden dil  $s \geq 0$



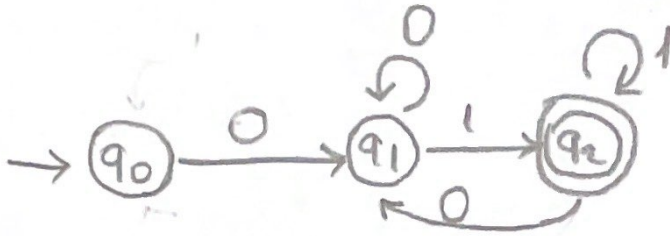
### DFA örnek 3:

Yukarıdaki örnekte  $s > 0$  ise;



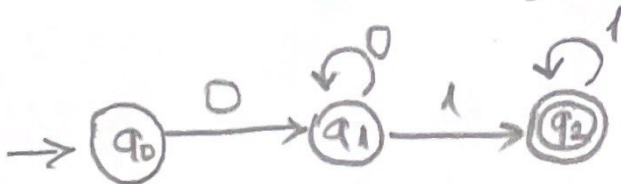
### DFA örnek 4:

0 ve 1'lerden oluşan ve 0 ile başlayıp 1 ile  
biten dil



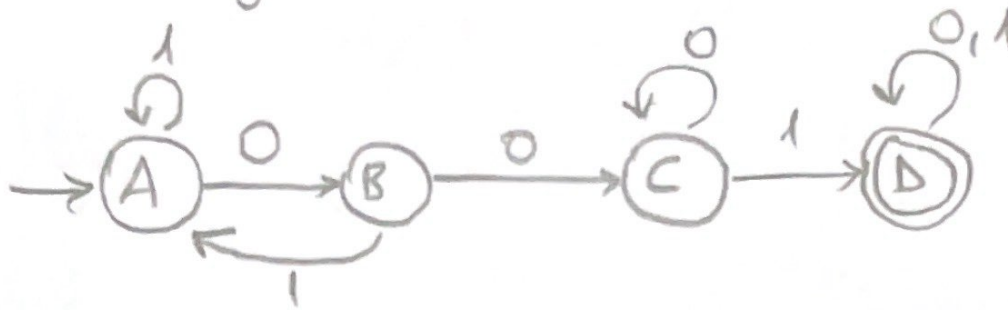
### DFA örnek 5:

$\{0^n 1^k \mid n \geq 1 \text{ ve } k \geq 1\}$



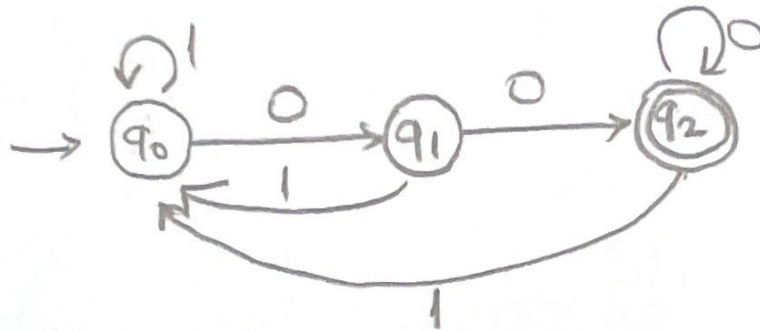
### DFA örnek 6:

0 ve 1'lerden oluşan 001 alt stringini içeren tüm stringler



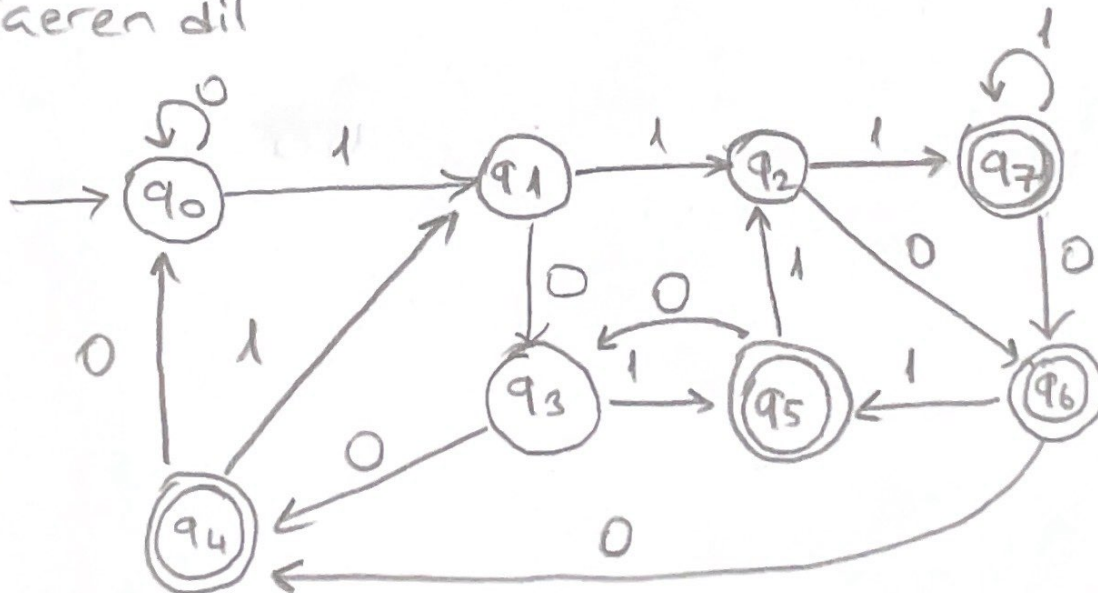
### DFA örnek 7:

0 ve 1'lerden oluşan 00 ile biten dil

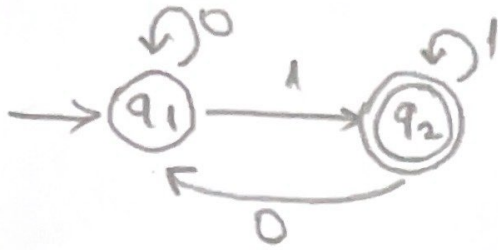


### DFA örnek 8: ★★

Sonunda 3. karakteri 1 olan tüm stringleri içeren dil



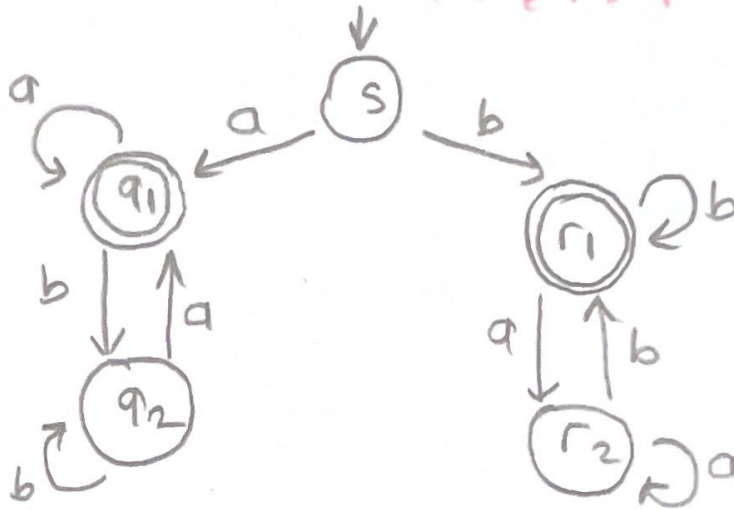
## DFA örnek 9:



1 ile biten stringlerden oluşan dil.

	0	1
q <sub>1</sub>	q <sub>1</sub>	q <sub>2</sub>
q <sub>2</sub>	q <sub>1</sub>	q <sub>2</sub>

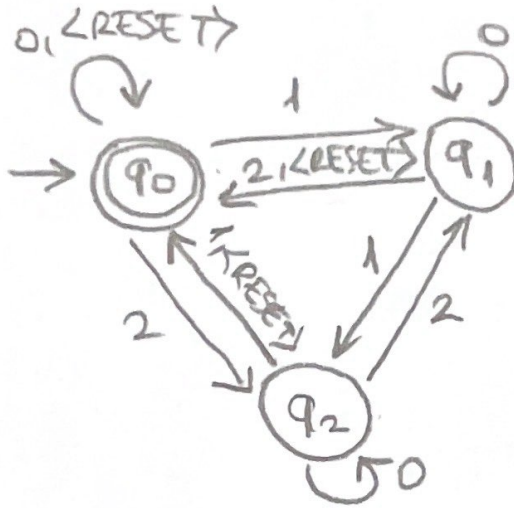
## DFA örnek 10: ★ ★ ★



a ile başlayıp a ile biten veya b ile başlayıp b ile biten stringlerden oluşan dil.

## DFA örneği 11: ★★

$$\Sigma = \{ \langle \text{RESET} \rangle, 0, 1, 2 \}$$



Her geleni sayıyı  
öncekine ekleyen ve  
3'ün katı olduğun-  
da final duruma  
geçen makine.  
(Reset toplamı 0'a  
döndürür.)

$$\delta_i(q_j, 0) = q_j$$

$$\delta_i(q_j, 1) = q_k \quad k = j+1 \text{ mod } 3 \text{ olan } k$$

$$\delta_i(q_j, 2) = q_k \quad k = j+2 \text{ mod } 3 \text{ olan } k$$

$$\delta_i(q_j, \langle \text{RESET} \rangle) = q_0$$

Bu şekilde yukarıdaki makineyi  
genelleştirip formüle dökmüş oluyoruz.



## DFA operations / Örnek 1:

Alfabe  $\Sigma$  26 harften oluşsun  $\{a, b, c, d, \dots, z\}$ .

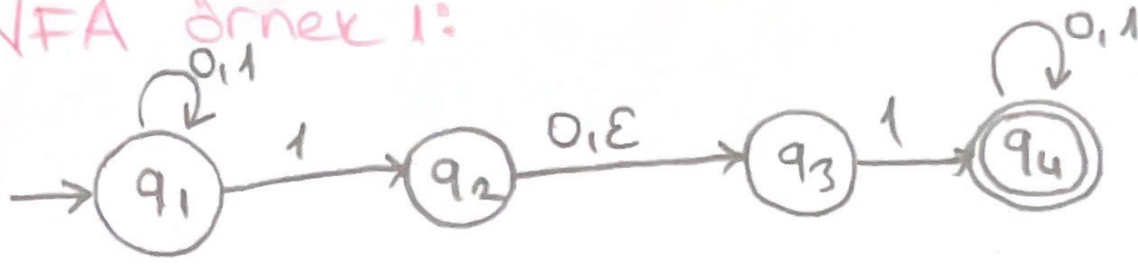
$A = \{\text{good, bad}\}$  ve  $B = \{\text{boy, girl}\}$  olmak üzere;

$$A \cup B = \{\text{good, bad, boy, girl}\}$$

$$A \circ B = \{\text{goodboy, goodgirl, badboy, badgirl}\}$$

$$A^* = \{\epsilon, \text{good, bad, goodgood, goodbad, badgood, badbad, goodgoodgood, goodgoodbad,} \dots\}$$

## NFA örnek 1:

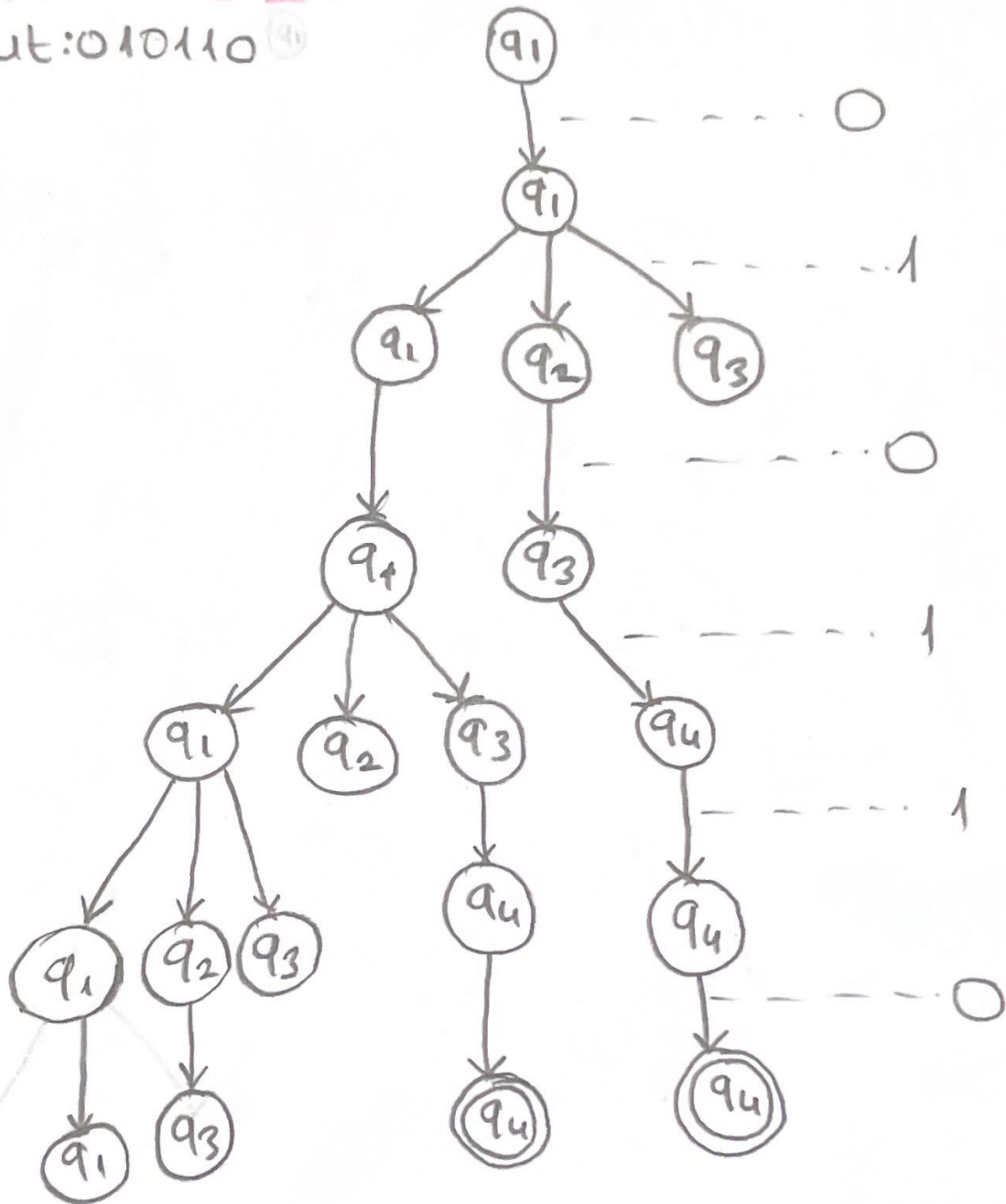


Bu örnekte ;

- $q_1$ 'de iken 0 gelirse DFA gibi işlenir.
- 1 gelirse 1 ile ilişkili durum sayısı kadar makinenin kopyası üretilir ve bu kopyaların her biri bir durum üzerinden yolu takip eder.
- Eğer sonrasında yine bir sayı için birden fazla durum olsaydı aynı şekilde bölünerek devam ederdi.
- Eğer kopya makineye gelen input okların hiçbirisiyle temsil edilmiyorsa o makine ve o makineye ait alt dallar ölür.
- En sonunda herhangi bir makine input'un sonunda bir duruma uyuyorsa NFA o input'lu string olarak kabul eder.
- $\epsilon$  sembolü için de aynıısı geçerlidir.

# NFA örnek 2:

input: 010110



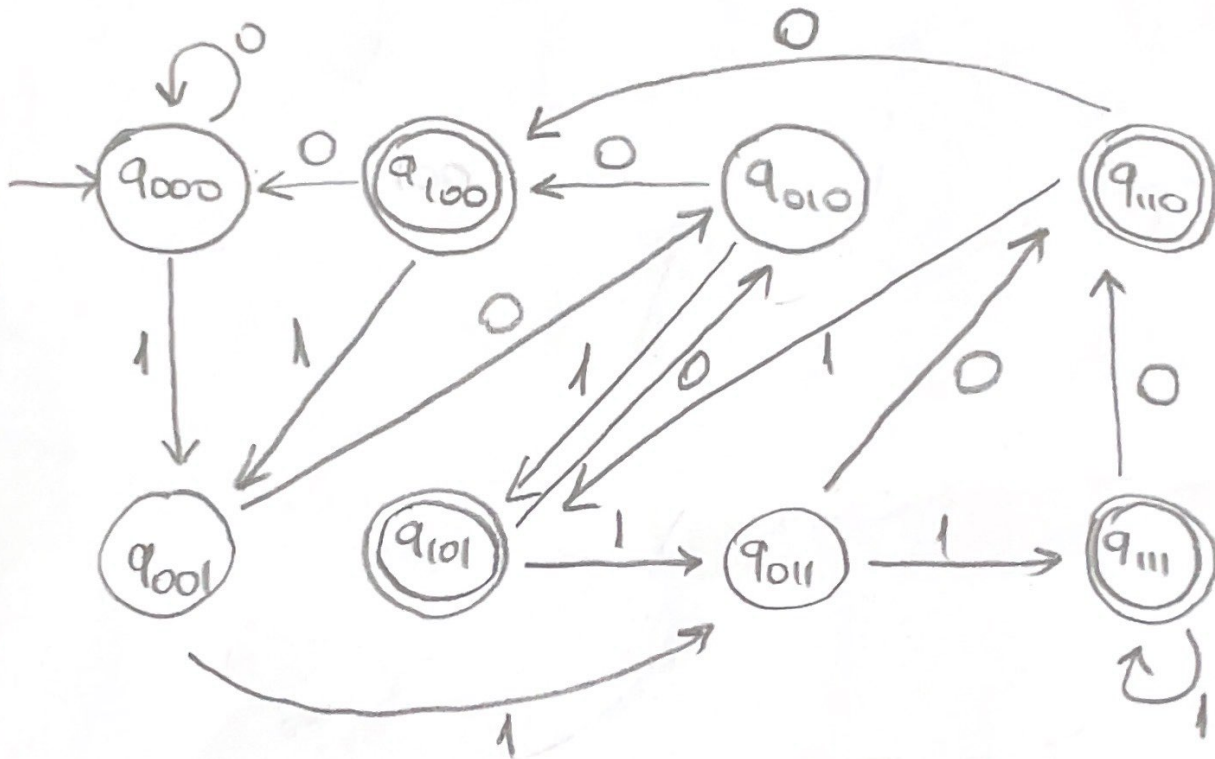
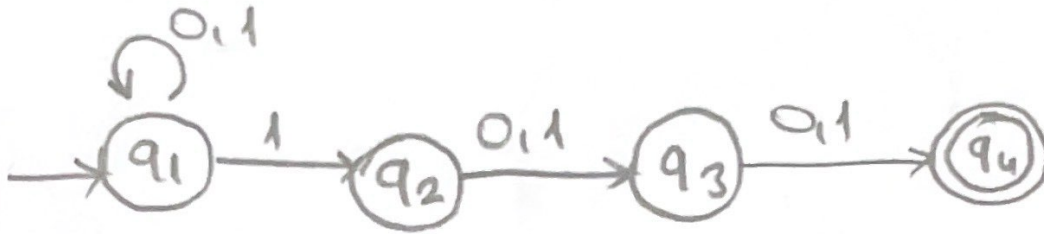
Original

54271



### NFA örnek 3:

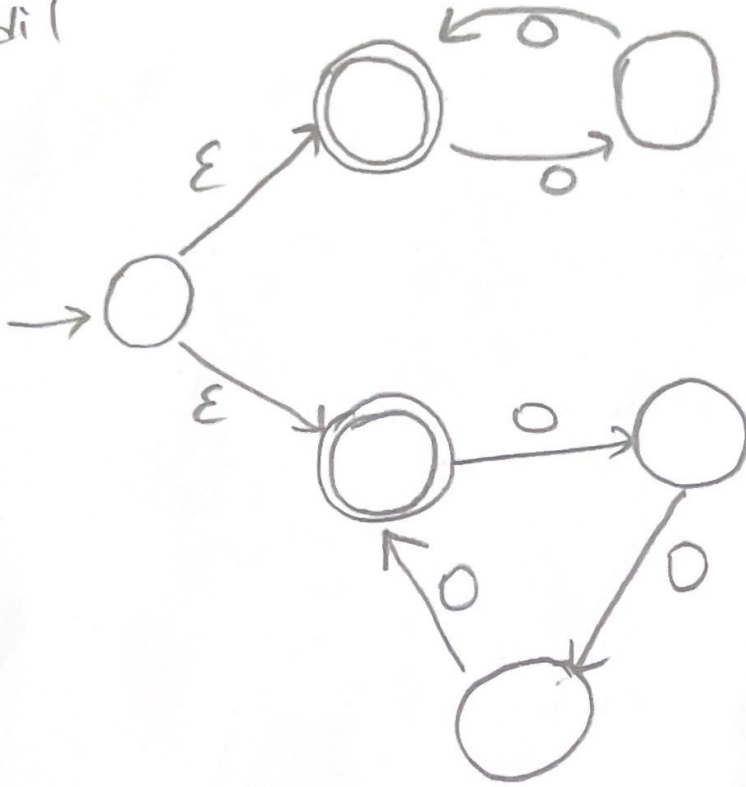
$\Sigma = \{0,1\}$  iken sondan 3. karakteri 1 olan stringleri kabul eden NFA



Aynı makine DFA ile yapılırsa

# NFA Örnek 4:

$\Sigma = \{0\}$  (unary alphabet)  $0^k, k$  2'nin  
ya da 3'ün katı olmak üzere kabul eden  
dil



## NFA Örnek 5:

$\epsilon$ , a, baba, baa stringlerini kabul eden  
ama b, bb, babba stringlerini kabul etme-  
yen dil

