

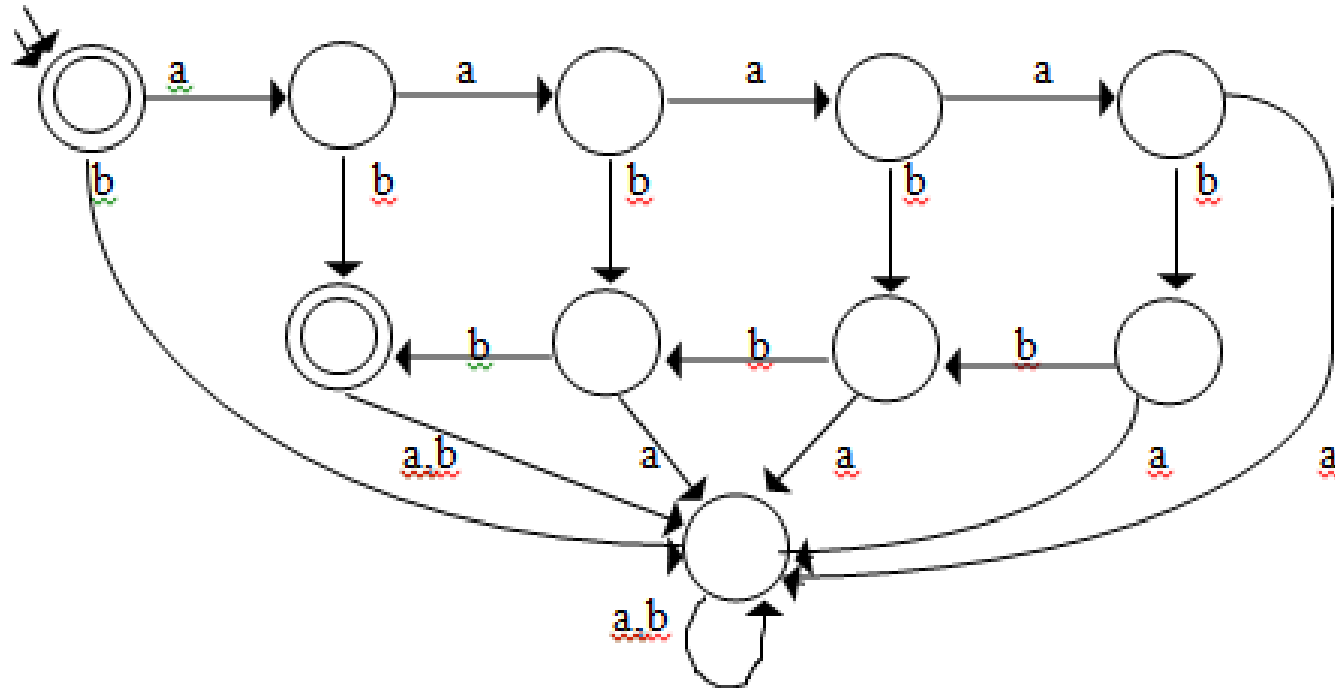
## Exercise : DFA

$L = \{a^n b^n : 0 \leq n \leq 4\}$  verilmiş olsun:

L dilinin düzenli olduğunu ispatlayınız.

# İspat-1

- Bir FA tasarlanabilirse RL'dir



## İspat-2: Bir RE yazılabilirse RL'dir.

- $(e \cup ab \cup aabb \cup aaabbb \cup aaaabbbb)$

PAMUKKALE ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ  
2021 BAHAR

# Biçimsel Diller ve Otomata Teorisi

## Formal languages and automata theory

NFA- Nondeterministic Finite Automata

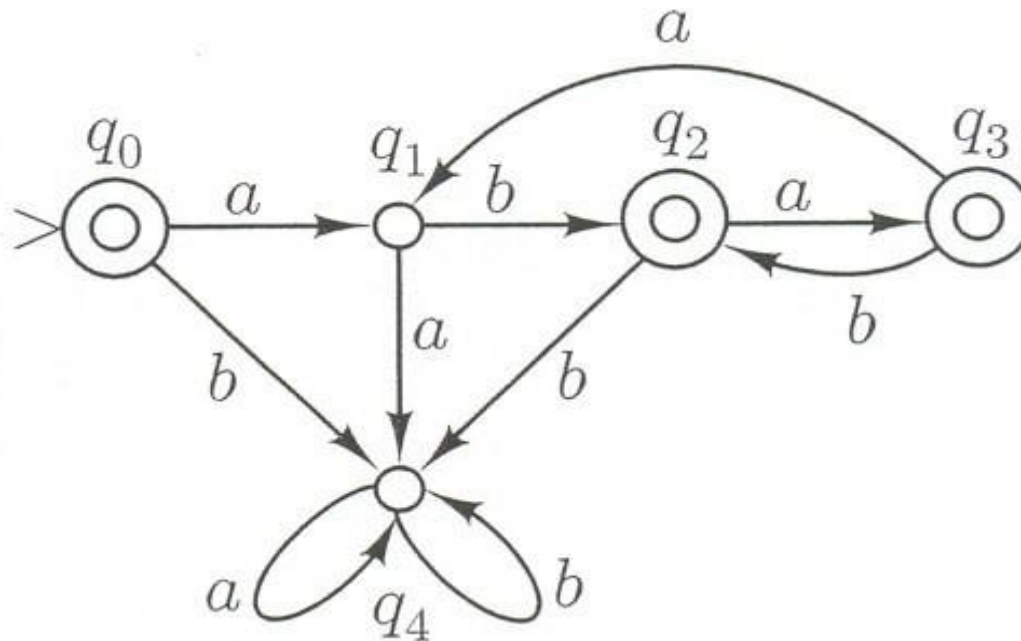
---

# Nondeterministic Finite Automata

- Aynı giriş bilgisi ve aynı durum için birden fazla sonraki durum olabilir.
- Bu durumlardan herhangi birine geçebilir, bu yüzden **nondeterministic** olarak adlandırılır.
- Bilgisayarların gerçek modellenmesinde kullanılmazlar, somut bir makine karşılığı yoktur.
- Automata tanımlamasını basitleştirmek için kullanılır.
- Her NFA'nın DFA karşılığı vardır.

# Nondeterministic Finite Automata

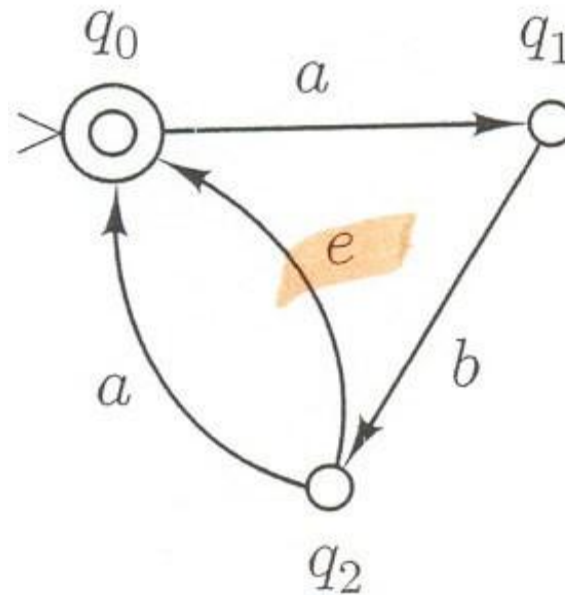
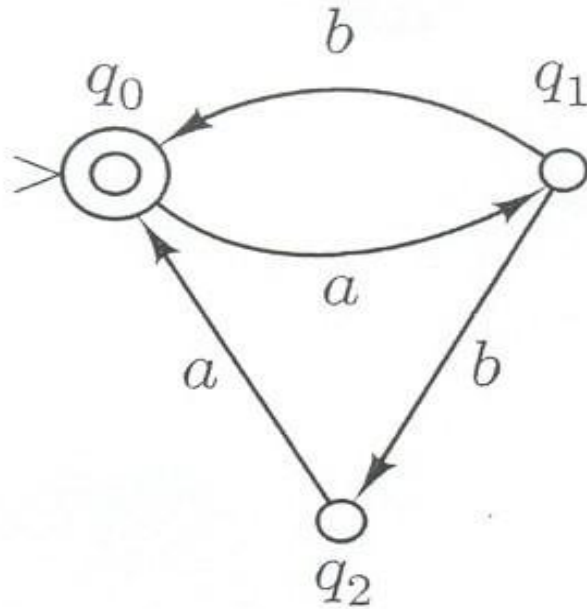
$L = (ab \cup aba)^*$  dilini tanıyan **deterministic automaton**



- Her node'dan  $a$  ve  $b$  olmak üzere iki çıkış vardır.

# Nondeterministic Finite Automata

Aynı dil  $L = (ab \cup aba)^*$ , aşağıdaki nondeterministic automata tarafından tanımlanabilir.



- Bir string başlangıç durumundan bir sonuç durumuna geçişi sağlayabiliyorsa kabul edilir.
- Nondeterministic automaton tarafından  $e$  string için de geçiş tanımlanabilir.

# Nondeterministic Finite Automata

## Definition:

Bir NFA quintuple olarak tanımlanır.  $M = (K, \Sigma, \Delta, s, F)$

$K$  sonlu sayıda durumlar kümesi

$\Sigma$  alfabe

$\Delta$  transition relation (fonksiyon değil)  $K \times (\Sigma \cup \{e\}) \rightarrow K$

$s \in K$  başlangıç durumu (sadece bir tane)

$F \subseteq K$  final state(s) kümesi

- Her  $(q, u, p) \in \Delta$  üçlüsü  $M$ 'in geçişi olarak adlandırılır.



# Nondeterministic Finite Automata

- $M$ 'nin configuration'ı  $K \times \Sigma^*$ 'dır.  $(q, w) \vdash_M (q', w')$  geçişi için

$w = uw', u \in I \cup \{e\}$  ve  $(q, u, q') \in L$  olmak zorundadır.

- $\vdash_M$  bir fonksiyon değildir çünkü bir  $(q, w)$  konfigürasyonu için çok sayıda  $(q', w')$  konfigürasyonu olabilir.

- Bir string  $w \in \Sigma^*$  kabul edilir, eğer sadece ve sadece

$(s, w) \vdash_M^* (q, e)$  ve  $q \in F$  ise

- Sonuç olarak bir  $M$  otomatı tarafından tanınan dil  $L(M)$  şeklinde gösterilir ve tüm kabul edilen string'ler kümesidir.

# Nondeterministic Finite Automata

Örnek:

$M$  bir NFA ve  $M = (K, \Sigma, \Delta, s, F)$  şeklinde tanımlanmıştır ve içerisinde  $bb$  veya  $bab$  substring'i bulunduran stringleri tanır.

# Nondeterministic Finite Automata

## Örnek:

$M$  bir NFA ve  $M = (K, \Sigma, \Delta, s, F)$  şeklinde tanımlanmıştır ve içerisinde  $bb$  veya  $bab$  substring'i bulunduran stringleri tanır.

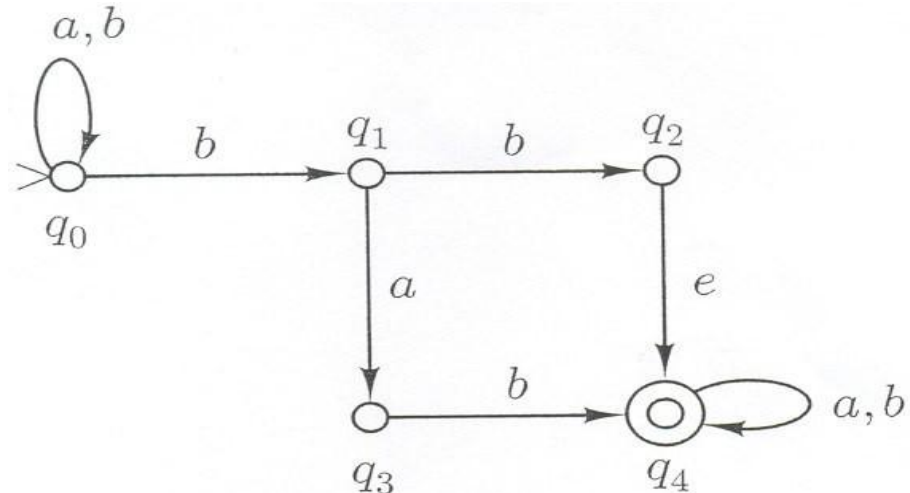
$$K = \{q_0, q_1, q_2, q_3, q_4\}$$

$$\Sigma = \{a, b\}$$

$$\Delta = \{(q_0, a, q_0), (q_0, b, q_0), (q_0, b, q_1), (q_1, b, q_2), (q_1, a, q_3), (q_2, e, q_4), (q_3, b, q_4), (q_4, a, q_4), (q_4, b, q_4)\}$$

$$s = q_0$$

$$F = \{q_4\}$$



# Nondeterministic Finite Automata

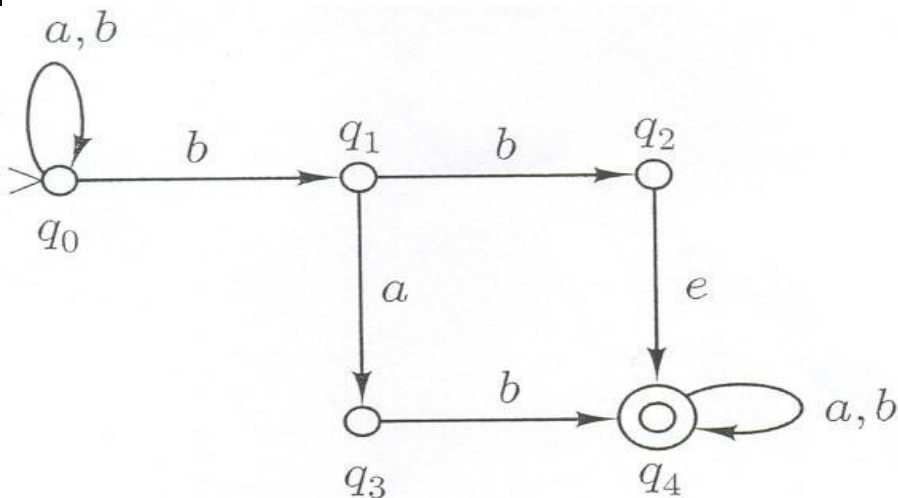
Örnek:(Devam) bababab string'ini tanır mı ?

$(q_0, bababab)$

$\vdash_M (q_1, ababab)$   
 $\vdash_M (q_3, babab)$   
 $\vdash_M (q_4, abab)$   
 $\vdash_M (q_4, bab)$   
 $\vdash_M (q_4, ab)$   
 $\vdash_M (q_4, b)$   
 $\vdash_M (q_4, e)$

$(q_0, bababab)$

$\vdash_M (q_0, ababab)$   
 $\vdash_M (q_0, babab)$   
 $\vdash_M (q_0, abab)$   
 $\vdash_M (q_0, bab)$   
 $\vdash_M (q_0, ab)$   
 $\vdash_M (q_0, b)$   
 $\vdash_M (q_0, e)$



## Nondeterministic Finite Automata

### Örnek:

$M$  bir NFA ve  $M = (K, \Sigma, \Delta, s, F)$  şeklinde tanımlanmıştır.  $M$  otomatu

$L(M) = \{w \in \Sigma^* : w \text{ string'i alfabedeki en az bir elemanı bulundurmaz}\}$  dilini tanır.

$$K = \{s, q_1, q_2, q_3\}$$

$$\Sigma = \{a_1, a_2, a_3\},$$

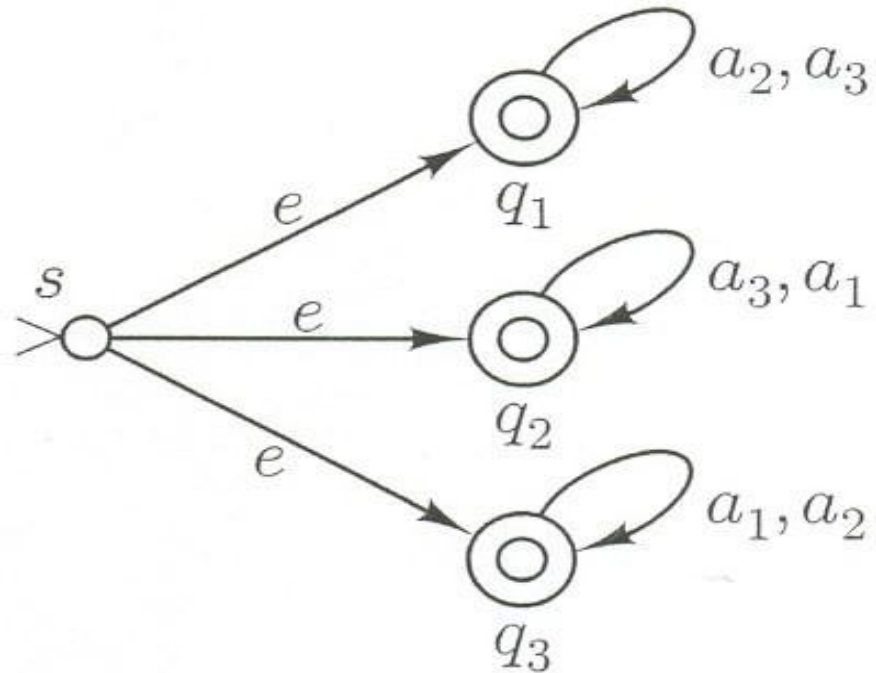
$$F = \{q_1, q_2, q_3\}$$

$(s, e, q_i)$  initial transitions

$(q_i, a_j, q_i)$  main transitions  $i \neq j$

$$e, a_1, a_2, a_1a_1a_3a_1 \in L$$

$$a_3a_1a_3a_1a_2 \notin L$$



# Nondeterministic Finite Automata

- Deterministic automata'da  $\delta$  transition  $K \times \Sigma \rightarrow K$  'ya bir fonksiyondur.
- Deterministic automata'da  $(q, e, p) \notin \delta$  'dır.
- Deterministic automata'da her  $q \in K$  ve  $a \in \Sigma$  için  
sadece bir tane  $p \in K$  vardır ve  $(q, a, p) \in \delta$  'dır.
- Bir nondeterministic automata'nın kendisine eşit bir deterministic karşılığı her zaman bulunabilir (NFA to DFA conversion).
- iki automata  $M_1$  ve  $M_2$  eşittir sadece ve sadece

$$L(M_1) = L(M_2) \text{ ise}$$

# Nondeterministic Finite Automata

Örnek:

Aşağıdaki nondeterministic finite automaton hangi stringleri kabul eder.

*b*

*bab*

*e*

*aa*

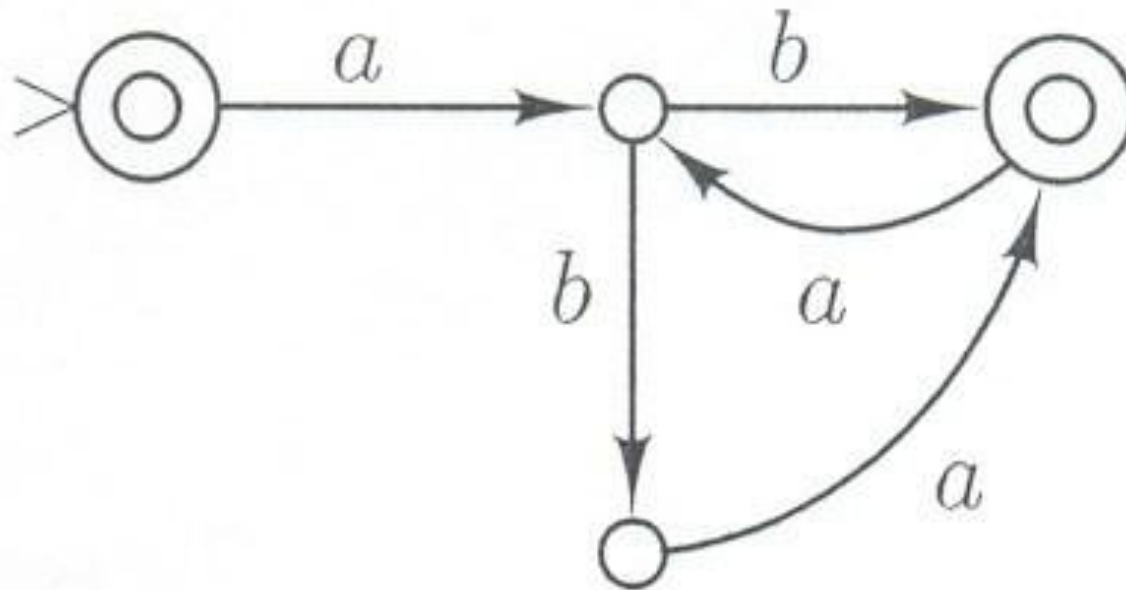
*abab*

*ab(aba)\**

*abaab*

*abaaa*

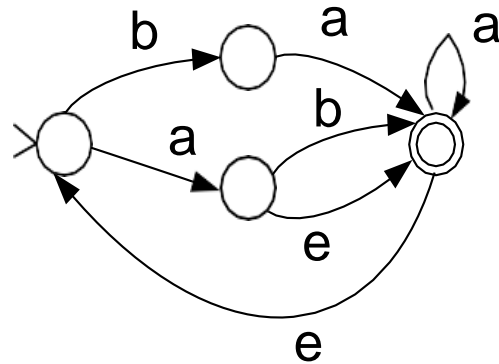
*abb*



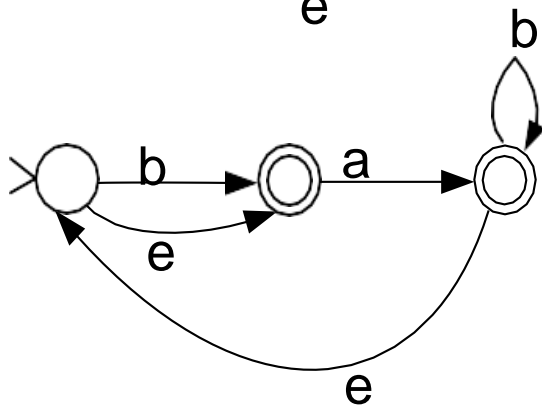
# Nondeterministic Finite Automata

Örnek:

$((ab)^*(ba)^*) \cup aa^*$  dilini tanıyan nondeterministic automata'nın state diagram'ını çiziniz.



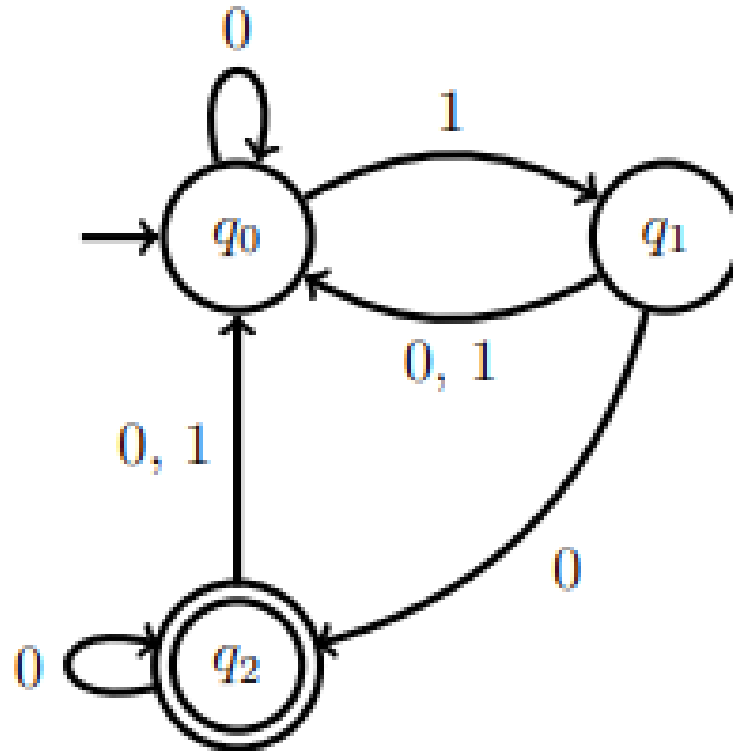
Hangi girişler için hata oluşur?





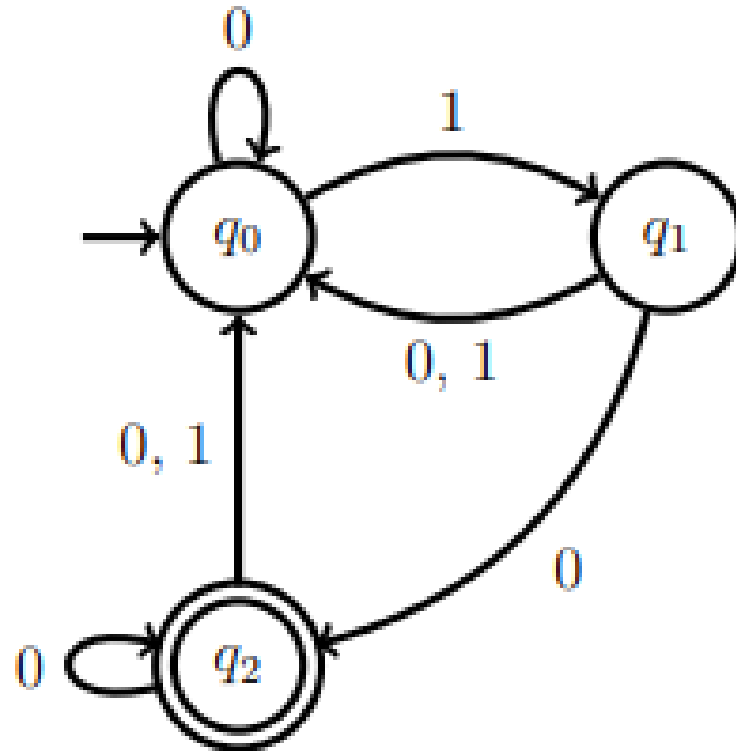
# Exercise

Aşağıdaki NFA için 1011 girişi kabul edilir mi?



# reachable states

- $\epsilon : \{q_0\}$
- $1 : \{q_1\}$
- $10 : \{q_0, q_2\}$
- $101 : \{q_0, q_1\}$
- $1011 : \{q_0, q_1\}$

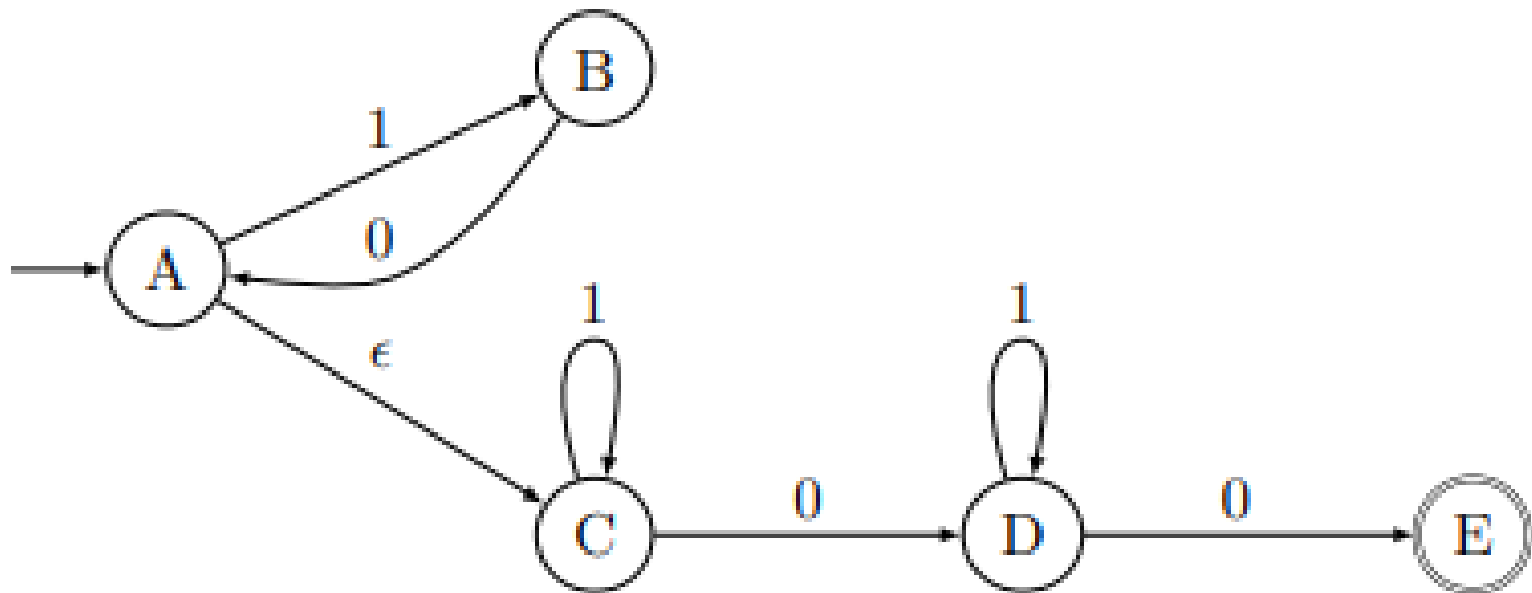


## Exercise

Aşağıdaki dili kabul eden NFA için durum diyagramını çiziniz.

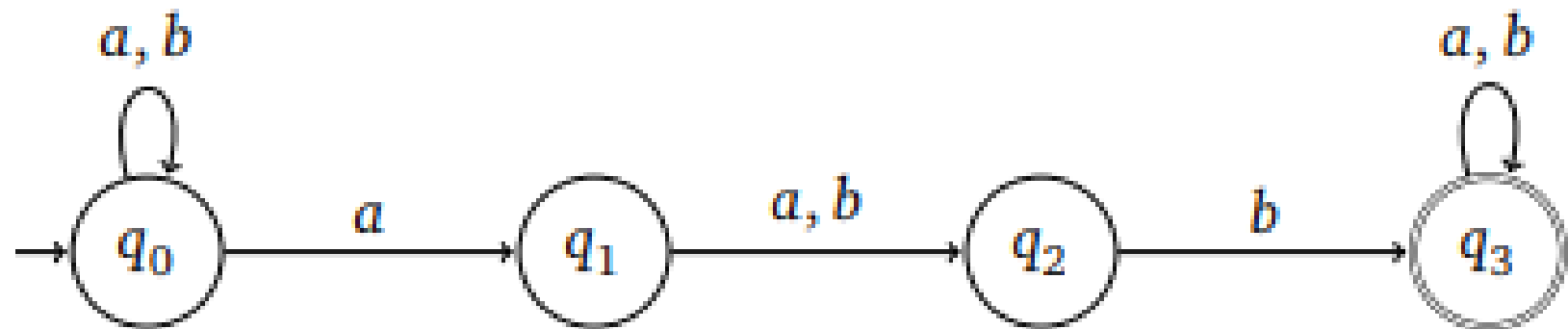
$(10)^*1^*01^*0$

$(10)^*1^*01^*0$



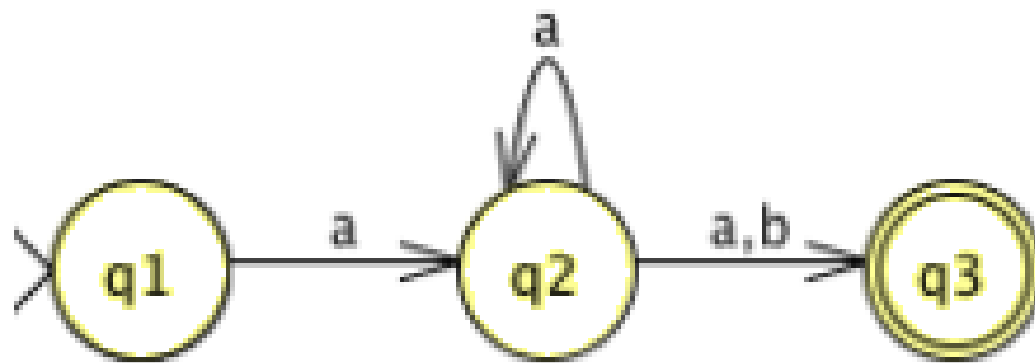
# Exercise

$L = \{w \in \{a, b\}^* \mid \text{en az bir } a \text{ sembolü } w \text{ katarının herhangi bir } i. \text{ konumunda oluşur ve bir } b \text{ de } i + 2. \text{ pozisyonunda oluşur}\}.$



# Exercise

- $aa^* (a \cup b)$  RE tanıyan NFA çiziniz.





# Ödev

- Problemleri çözünüz 2.2.1, 2.2.2, 2.2.3, 2.2.6(a) (sayfa 73- 63)