

# **Biçimsel Diller ve Otomata Teorisi**

*Dr. Öğr. Üyesi Hayri Volkan Agun  
Bilgisayar Mühendisliği Bölümü  
Bursa Teknik Üniversitesi*

# Kaynaklar

## Ders Kitabı

- An Introduction to Formal Languages and Automata, Peter Linz, 6th Edition, 2017.
- An Introduction to Computer Theory, Daniel Isaac Aryeh Cohen, 2nd Edition, 1996.

## İçerik

- %100 Teorik
- Klasik sınav
- Vize %40, Final %60



# Sonlu durum otomati

## Sonlu durum otomati kullanım alanları

- Yüksek seviyeli bir programa ait kodun alt koda çevrilmesinde
- DNA dizinden yer alan genlerin yakalanmasında
- Metinlerde Tarih, Adres, İsim gibi öbeklerin yakalanmasında
- Tekrar eden belirli durumların yakalanmasında
  - Uçak simulasyonunda stalt pozisyonuna neden olan pilot hatalarının saptanmasında.
  - Karmaşık bir devre tasarımında gerilimin artmasına sebep olan bağlantıların tespit edilmesinde
  - Hastalıkların semptomlarının bir arada geçtiği durumlarda hastalığın belirlenmesinde

# Sonlu durum otomati

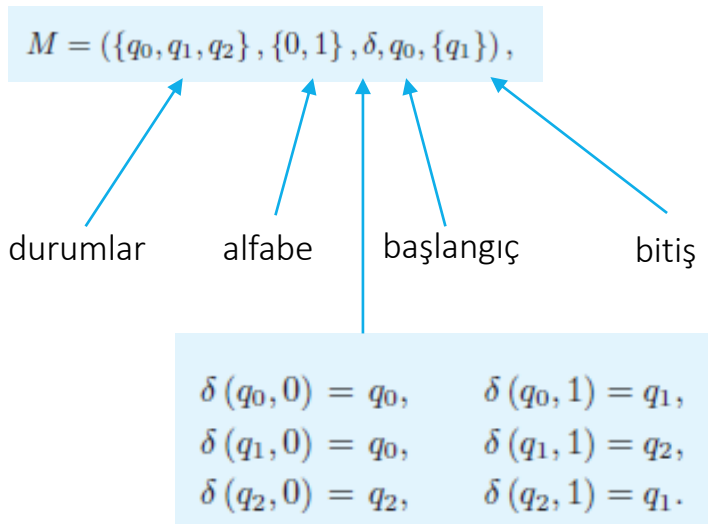
- Kararlı (Deterministic) Sonlu Durum Otomati aşağıdaki şekilde tanımlanır.

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

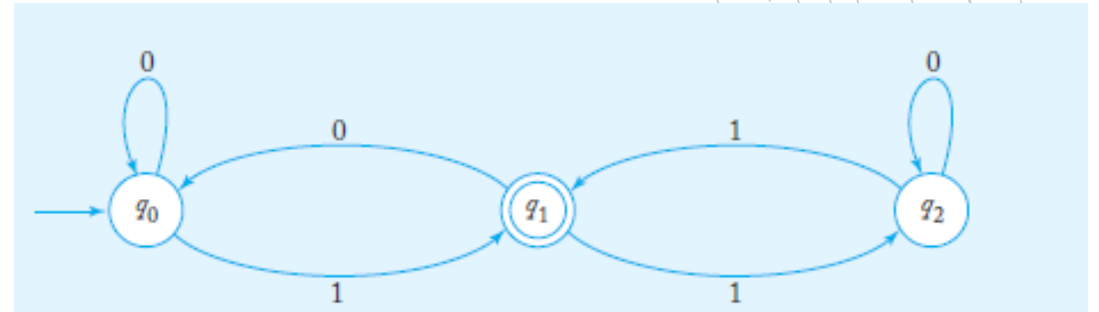
- $Q$ : Tanımlanan durumlar
- $\Sigma$  : Alfabe
- $\delta : Q \times \Sigma \rightarrow Q$  Geçiş sembolleri
- $q_0 \in Q$  : Başlangıç durumu
- $F \subseteq Q$  : Bitiş durum kümesi

# Sonlu durum otomati

## Sonlu durum otomati tanımlaması



## Sonlu durum otomati şekil



# Sonlu durum otomati

- Formal tanımlamada bir durumdan diğer bir duruma geçiler öz yinelemeli olarak aşağıdaki gibi tanımlanabilir.

$$\begin{aligned}\delta^*(q, \lambda) &= q, \\ \delta^*(q, wa) &= \delta(\delta^*(q, w), a),\end{aligned}$$

Boş sembol: boş sembol ile ancak aynı duruma geçilebilir.

- Örneğin;  $\delta(q_0, a) = q_1$  ve  $\delta(q_1, b) = q_2$ , tanımlı olsun. Buna göre yukarıdaki açılımı yaparsak  $\delta(\delta(q_0, a), b)$  gibi yazabiliriz. Bunu aşağıdaki şekilde açalım.

$$\delta^*(q_0, ab) = q_2.$$

$$\begin{aligned}\delta^*(q_0, a) &= \delta(\delta^*(q_0, \lambda), a) \\ &= \delta(q_0, a) \\ &= q_1.\end{aligned}$$



$$\delta^*(q_0, ab) = \delta(q_1, b) = q_2,$$

# Sonlu durum otomatı

- Formal olarak dil; bu dili tanımlayan bir otomata tarafından kabul edilen her türlü karakter katarını temsil eden bir kümedir.
- Otomatalar bir dili kabul etmesi için aşağıdaki iki şartı sağlamalıdır.
  - Başlangıç geçiş sembolleri kullanılarak her hangi bir son duruma dili soldan sağa tüketerek varılabiliyorsa bu dil bu otomat tarafından kabul edilir.
  - Dili temsil eden tüm karakter katarları yukarıdaki şartı sağlamalıdır. Eğer tek bir tanesi bile yukarıdaki şartı sağlamıyorsa o zaman dil otomat tarafından kabul edilemez.
- Yukarıdaki şartlar tek bir karakter katarı için çok basit bir şekilde sağlanabilir. Ancak bir dili temsil eden tüm karakter katarlarının test edilmesi sonlu olmadıkları için imkansızdır. İkinci durumun denklik ve eşitlik kavramları ile test edilmesi gerekir.

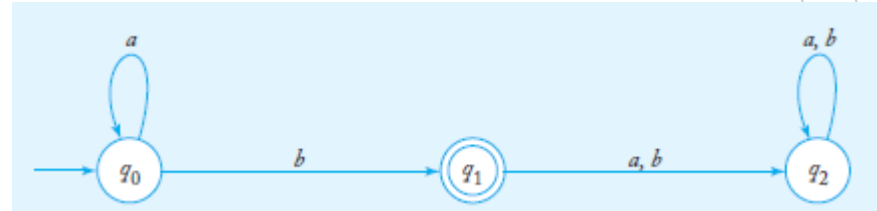
# Sonlu durum otomatı

- Yan tarafta şekli verilen otomat hangi dili kabul etmek için geliştirilmiştir.

- $L = \{a^n b : n \geq 0\}$

- $L = \{a^n b^n : n \geq 0\}$

- $L = \{a^n b^n a^m b^m : n \geq 0, m \geq 0\}$



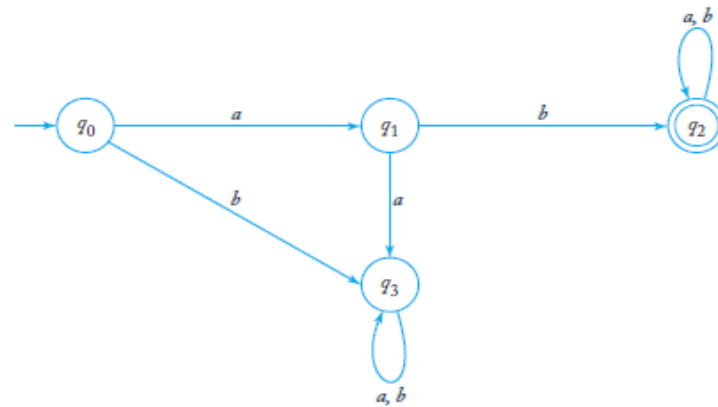


# Sonlu durum otomati

- a ve b alfabesinden oluşan bir dil için  $ab$  ön eki kabul eden otomati çiziniz?

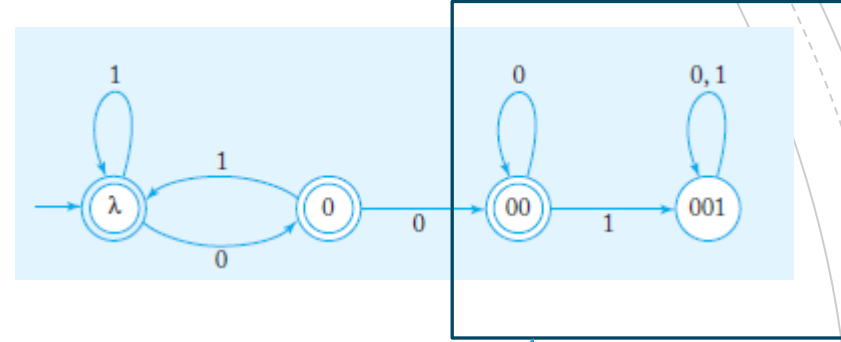


- a ve b alfabesinden tüm dili okuyan fakat  $ab$  ön eki yoksa sonlanmayan otomati çiziniz.



# Sonlu durum otomati

- 0 ve 1 alfabesinden oluřan ve ierisinde 001 gemeyen dili kabul eden otomati iziniz?
- Bu dil ierisinde herhangi bir ifadeyi barındırabilir ancak 001 barındıramaz.
- n ve son ek olarak 001 olamaz.
- Bu durumda 00 grr grmez 1 grrsek bunusonlu olmayan bir duruma iletmemiz gerekir. Diğerk durumda aynı durumda 0 geldiğisrece beklemeliyiz.  $\delta(00, 0) = 00$



Tuzak durumları 00 ve 001 gelirse

# Düzenli diller

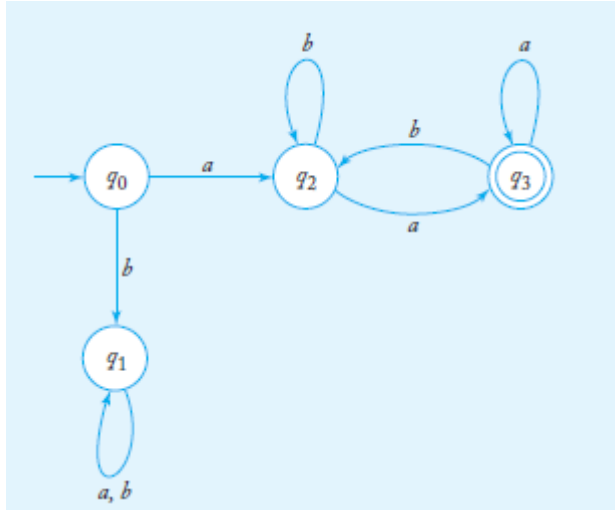
- Eğer bir dil kararlı bir sonlu durum otomatı ile ifade edilebiliyorsa bu dile *düzenli dil* denir.
- Kararlı sonlu durum otomatlarında her bir durumdan bir geçiş sembolü ile sadece bir sonlu duruma varılır. Kısaca bir geçiş sembolü farklı iki duruma ulaşmak için kullanılamaz.
- L dili M kabul otomatı ile ifade edilebiliyorsa düzenli bir dildir. Burada M otomatı kararlı sonlu durum otomatıdır.

$$L = L(M).$$

- Örneğin  $L = \{awa : w \in \{a,b\}^*\}$  dili kararlı sonlu durum otomatı ile ifade edilebilir. Bu durumda bu dil düzenli bir dildir.

# Düzenli diller

- Aşağıdaki otomat düzenli bir dile aittir. Bu dili belirten ifadeyi yazınız?



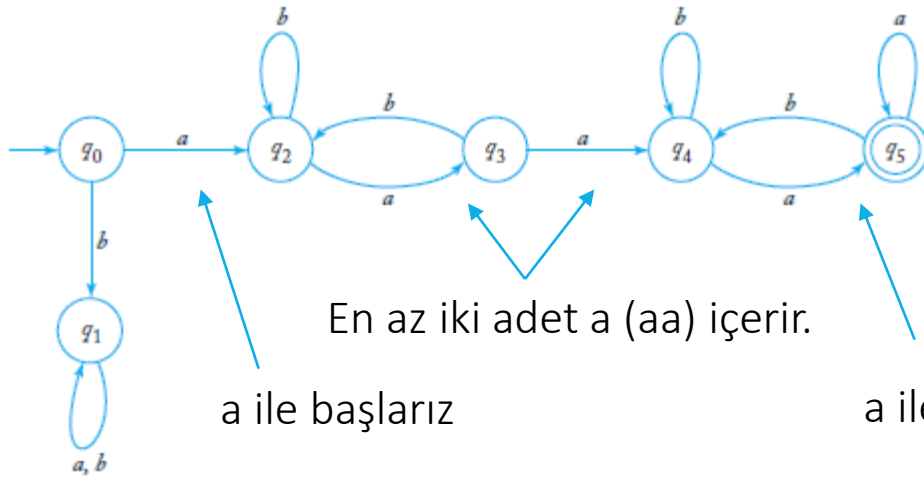
# Düzenli diller

- Aşağıda tanımlanan dil düzenli bir dil midir? Gösteriniz?

$$L^2 = \{aw_1aaw_2a : w_1, w_2 \in \{a, b\}^*\}.$$

- Bu dil a ile başlayan, aa içeren ve a ile sonlanan bir dildir. Bu dili yakalayan sonlu durum otomatınız çizerek bu dilin düzenli bir olduğunu kanıtlayabiliriz.

# Düzenli diller



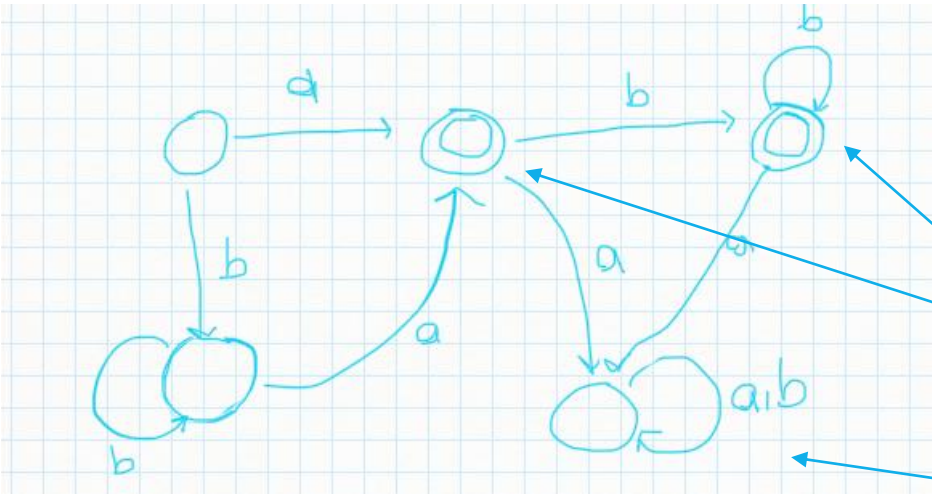
- Bu örnek ile  $L$  dili eğer düzenli ise  $L^2, L^3, \dots$  gibi dillerde düzenlidir diyebiliriz.
- Bunu önümüzdeki sunumlarda göstereceğiz.

# Örnekler

- a ve b alfabesinden oluşan ( $\Sigma=\{a,b\}$ ) aşağıdaki dil tanımlamalarına ait sonlu durum otomatını çiziniz?
  - a) Çift sayıda uzunlukta olan karakter katarları
  - b) Uzunluğu 5'den büyük olan karakter katarları
  - c) Her türlü sadece tek bir a barındıran tüm dil formasyonları
- Yandaki sorular için ilk adımda kabul edilen karakter dizilerini oluşturuz.
- Sonrasında kabul edilen durumlara ait sonlu durum otomatını çizeriz. Sadece kabul edilen durumlar olmalıdır.
- Kabul edilmeyen durumlar için tuzak durumları yukarıdaki çizime ekleriz.

# Örnekler

- Yukarıdaki sorunun c çıkki için olası karakter katarları aşağıda verilmiştir.
  - abbbbbbb...b, bbbbbba, bbbb..ab..., a



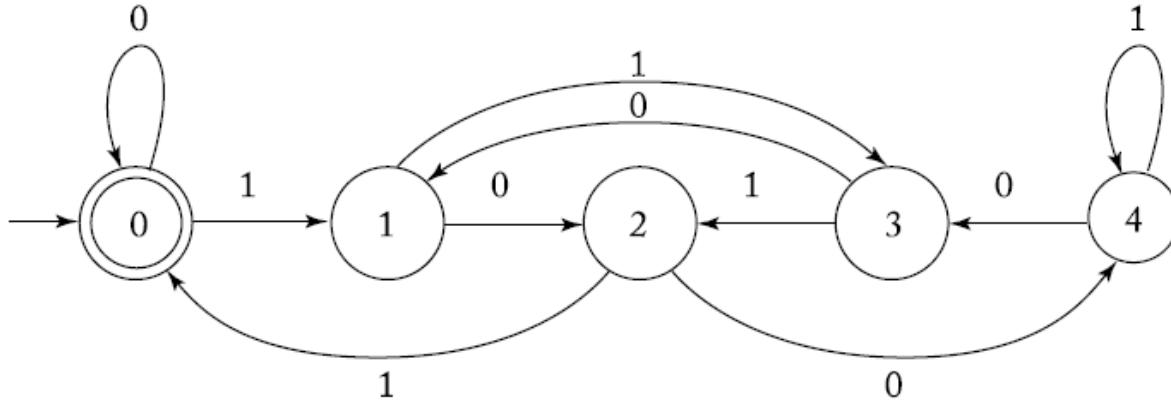
son durumlar

tuzak birden fazla a olması durumu



# Örnekler

- 1 ve 0 lardan oluşan ikili sayıların 10 luk düzende 5'e bölümünden kalanı kabul eden dile ait sonlu durum otomatını çiziniz?



Durumları kalan sayısı ile isimlendiririz. Örneğin: 100 = 4, 5'e bölümünden kalan 4 tür. Eğer sayının 5'e bölümünden 3 ise sonraki durum 3 numaralı durum olacaktır.

# Kararsız sonlu durum otomatları

- Kararsız sonlu durum otomatlarında bir durumdan diğer durumlara boş sembol ve/veya aynı sembol ile birden fazla duruma geçiş yapılabilir.
- Aşağıdaki tanımlama için

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

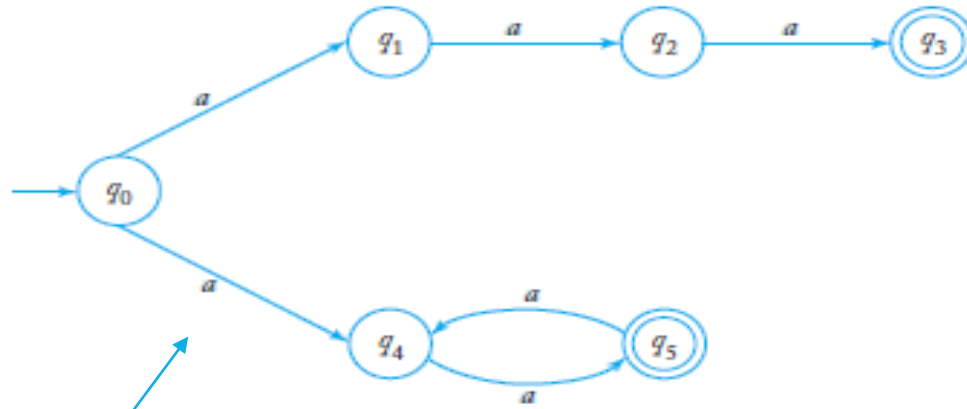
- Geçiş durumları:

$$\delta : Q \times (\Sigma \cup \{\lambda\}) \rightarrow 2^Q.$$

- Örneğin:

$$\delta(q_1, a) = \{q_0, q_2\},$$

# Kararsız sonlu durum otomatları

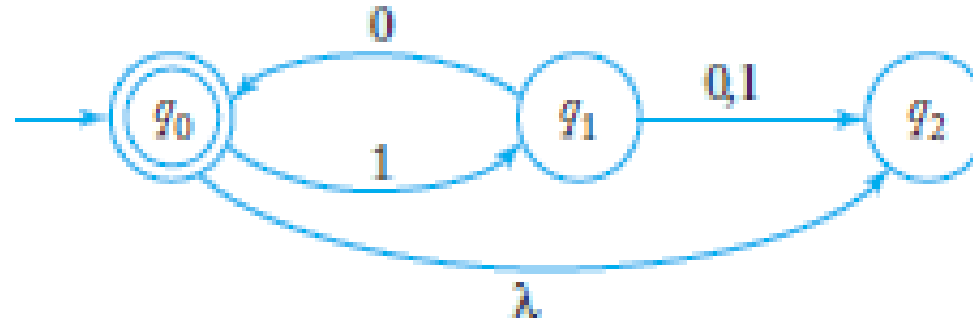


aynı alfabe (sembol) ile farklı iki  
duruma geçiş yapılıyor

# Kararsız sonlu durum otomatu

## Boş geiş

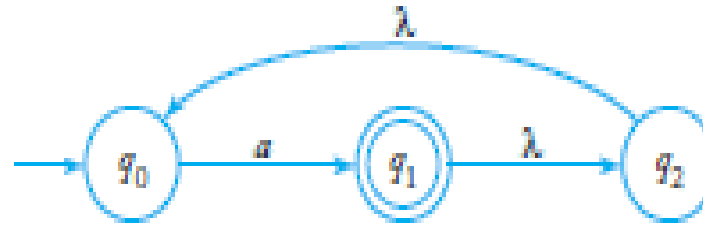
- Aşağıdaki otomatta  $\lambda$  boş geişı temsil etmektedir. Bu otomat 1010 ve 101010 karakter katarlarını kabul ederken 110 ve 10100 karakter katarlarını kabul etmez.
- Kararsız geiş durumlar  $q_1$  ve  $q_0$  arasında ve  $q_1$  ve  $q_2$  arasında gözlemlenmiştir. Burada 0 sembolü ile hem  $q_0$  hem de  $q_2$  ye geilmektedir.



# Kararsız sonlu durum otomatı

## Boş geiş

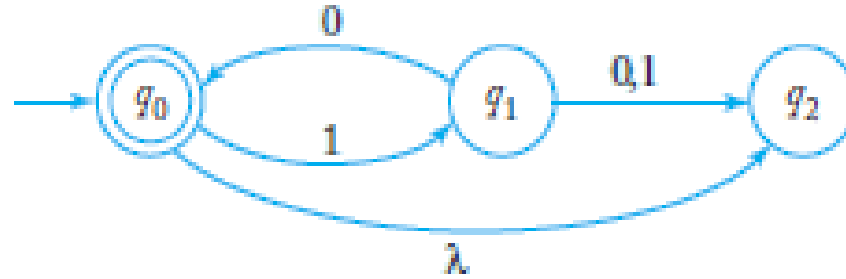
- Aşağıdaki otomat sonsuz uzunluktaki a sembollerini yakalayabilir. Bu otomatın kararsız olduğunu gösteriniz?
- Öncelikle kararsız sonlu durum otomatında bir durumdan aynı sembol ile farklı durumlara gidilebilmelidir. Burada görünürde böyle birşey söz konusu değildir. Ancak biraz sadeleştirirsek.
- $d(q_0, a) = q_1$  ve  $d(q_1, \lambda) = q_1$  ise o zaman  $d(q_0, a) = q_2$  dir. Benze şekilde  $d(q_2, a) = q_1$  ve  $d(q_2, \lambda) = q_2$  dir. İkinci durum kararsız sonlu durum otomatı tanımına uyduğu için bu otomat kararsız sonlu durum otomatıdır diyebiliriz.



# Kararsız sonlu durum otomatu

## Dil tanımı

- Aşağıdaki otomat için dil tanımı nedir?



- Dil tanımı yaparken önce örnek karakter katarlarına bakmamız gerekir. Sonra tekrar eden kısımları dil tanımında üstsel olarak sadeleştirebiliriz.
- Kabul edilen örnek karakter katarları: 10, 1010, 101010 .Bu durumda dil tanımını aşağıdaki gibi yazabiliriz.

$$L = \{(10)^n : n \geq 0\}.$$

# Kararsız sonlu durum otomatı

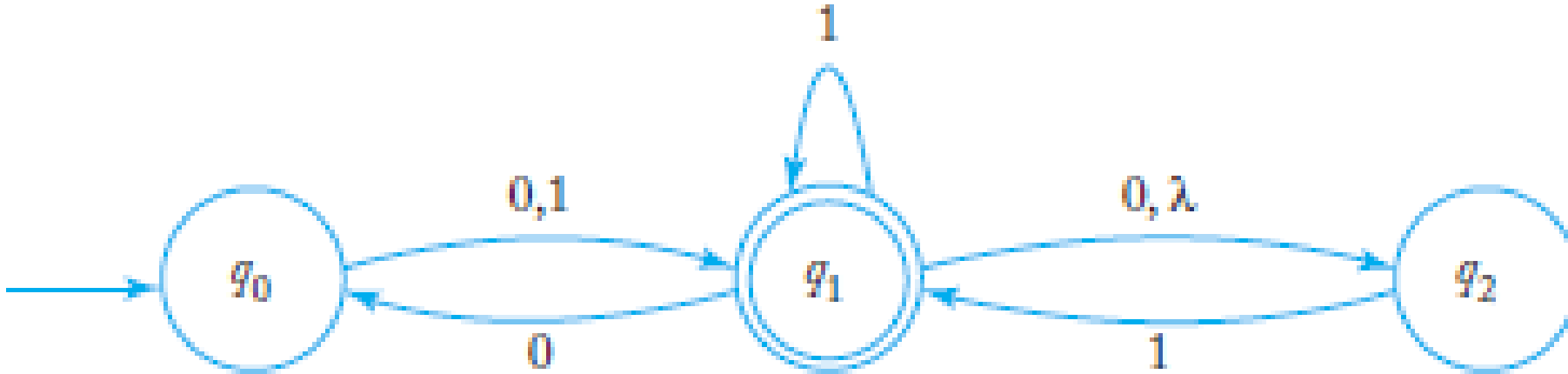
## Dil tanımı

- Normal şartlar altında kararsız durumlar bilgisayar ile işleme esnasında çok fazla işlem yapılmasına neden olur. Buyüzden de sonlu durum otomatlarında kararsız geçişler olabildiğince az kullanılır.
- Diğer bir durumda modern bilgisayar teknolojisi quantum bilgisayarlardan farklı olarak tümüyle kararlı bir yapıdadır. Dolayısıyla kararsız bir otomat için ek hesaplama yapması gerekecektir.
- Bazen bazı problemler doğası gereği kararsızlık içerir. Örneğin bir dilbilgisi (grammar) tanımlarken kullandığımız  $|$  operatörü ile tanımlanan aşağıdaki ifade kararsızdır. Bu  $S$  için iki farklı açılım önermektedir. İki durumda olasıdır.

$$S \rightarrow aSb|\lambda$$

# Örnekler

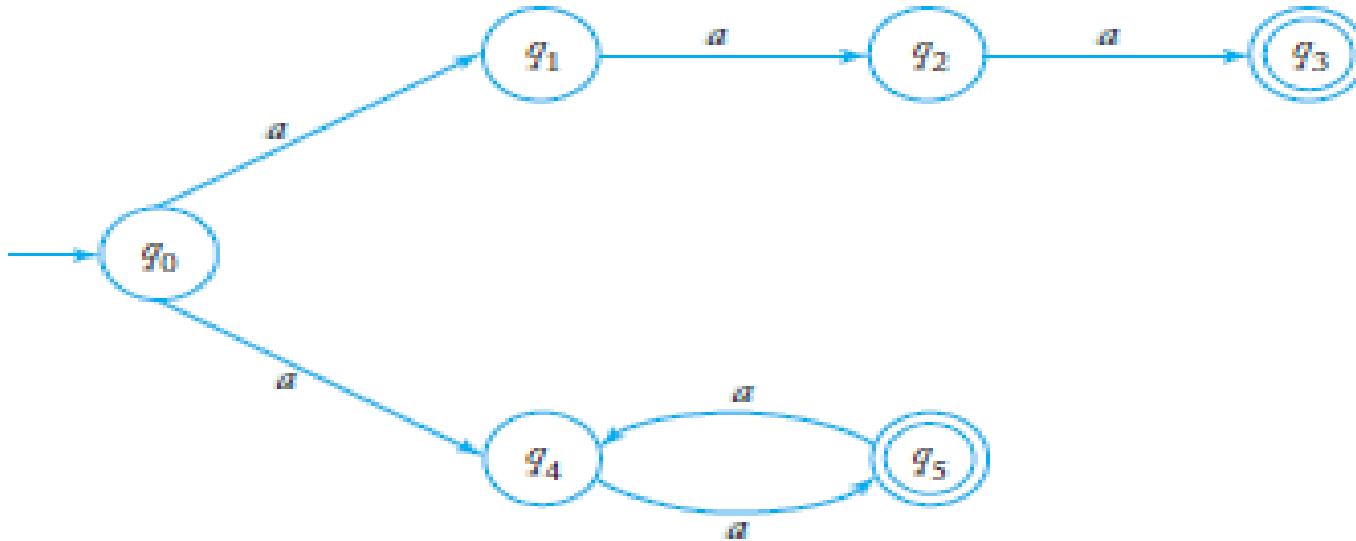
- Aşağıdaki otomatın türünü belirtiniz? Bu otomat  $\{00, \mathbf{01001}, 10010, \mathbf{000}, 0000\}$  kümesindeki hangi karakter katarlarını kabul eder?





# Örnekler

- Aşağıdaki otomatın türünü belirtiriz? Bu otomatın kabul ettiği dil ile  $a^5$  ifadesinin birleşimine sahip olan yeni dil için bu dili kabul eden otomati yeni durum eklemeden yadan durumlar üzerinde değişiklik yapmadan değiştiriniz?



# Örnekler

- $L = \{abab^n : n \geq 0\} \cup \{aba^n : n \geq 0\}$  dili için karasız sonlu durumu otomatını çiziniz?

