

## SONLU ÖZDEVİNİRLER

**Sonlu özdevinir (finite automata FA):** modeli kesikli giriş ve çıkışları olan matematiksel bir modeldir. Bu model birçok donanım ve yazılım sisteminin modellenmesinde kullanılır.

Metin düzenleyici (text editor), ve derleyicilerin belirli kesimleri ile zaman uyumlu dizisel devreler sonlu özdevinir modeline uygun yazılım ve donanı sistemlerin örnek olarak gösterilebilir.

Bilgisayar bilimleri ve mühendisliği için sonlu özdevinir modeli çok kullanılan önemli bir modeldir.

Sonlu özdevinirlerin birçok türü vardır. Fakat sonlu özdevinirleri 2 sınıfa ayırmak mümkündür.

- **1.Sonlu durum tanıyıcıları: (Finite State Recognizer)**
- **2.Çıkış Üreten Özdevinirler:**

Bu sınıfların her birinin de kendi içinde alt sınıfları vardır. Ancak kısaca “sonlu özdevinir ” ya da FA denilince temel model olan deterministlik sonlu durum tanıyıcı modeli anlaşılır.

### **1. Deterministlik Sonlu Özdevinir (DFA) Modeli:**

Temel modelde deterministlik sonlu özdevinir modeli bir beşli olarak tanımlanır.

$$DFA = \langle Q, \Sigma, \delta, q_0, F \rangle$$

$Q$  : Sonlu Sayıda Durum İçeren Durumlar Kümesi

$\Sigma$  : Sonlu Sayıda Giriş Simge sin den Oluşan Giriş Alfabeti

$\delta$  : Durum Geçiş İşlevi (state transition function)

$q_0$  : Başlangıç Durumu ( $q_0 \in Q$ )

Başlangıç durumu durumlar küme sin in bir elemanı olduğuna göre

$Q$  boş olmayan (içinde en az  $q_0$  bulunan) bir kümedir.

$F$  : Uç durumlar kümesi

Durumlar küme sin in bir alt kümesidir.

Temel modelde geiş iřlevi ile her (durum, giriř simgesi) iftine bir durum eřlenmektedir. Bu durumlardan ilki “řimdiki durum”, ikincisi ise “sonraki durum” olarak adlandırılırsa, durum geiş iřlevi ile her (řimdiki durum, giriř simgesi) iftine bir (sonraki durum) eřlendięi sylenebilir.

Yukarıdaki tanımdan da anlařıldığı gibi temel FA modeli deterministik bir modeldir. Bu model kısaca DFA (deterministic finite automata ) olarak adlandırılır. Kısaca FA (finite automata) denildięinde de DFA modeli anlařılır. Bu modele gre, FA  $q_0$  durumunda bařlar ve uygulanan her giriř simgesi ile yeni bir duruma geer. Her an FA’nın bulunduęu durum kesin olarak bellidir.

**rnek 1:** Ařaęıda M makinasının geiş iřlevinin matematiksel yapısı verilmektedir.

$$M_{1,1} = \langle Q, \Sigma, \delta, q_0, F \rangle$$

$$Q: \{q_0, q_1, q_2\}$$

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

$$F: \{q_2\}$$

$$\delta: \delta(q_0, 0) = q_0$$

$$\delta(q_0, 1) = q_1$$

$$\delta(q_1, 0) = q_0$$

$$\delta(q_1, 1) = q_2$$

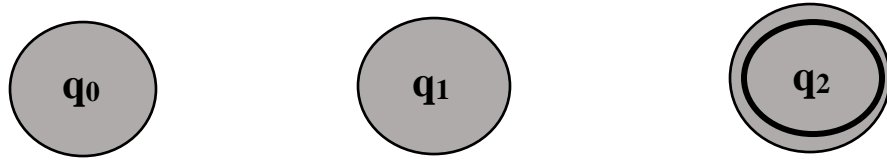
$$\delta(q_2, 0) = q_2$$

$$\delta(q_2, 1) = q_2$$

rnekte de grldę gibi geiş iřlevinin matematiksel tanımı hem uzun hemde anlařılması zor olan bir tanımdır. Bu nedenle geiş iřlevinin tanımı iin genellikle geiş izeneęi (transition diagram) olarak adlandırılan bir izenek kullanılır.

Ynl bir izge olan geiş izeneęinde her durum iin bir dęm bulunur. Durum geileri ise ynl yaylar ile gsterilir. Ve yayların zerine geiře neden olan giriř simgesi yazılır. Geiş izeneęinde bařlangı durumu “ $\rightarrow$ ” ile u durumlar ise ift ember ile gsterilir.

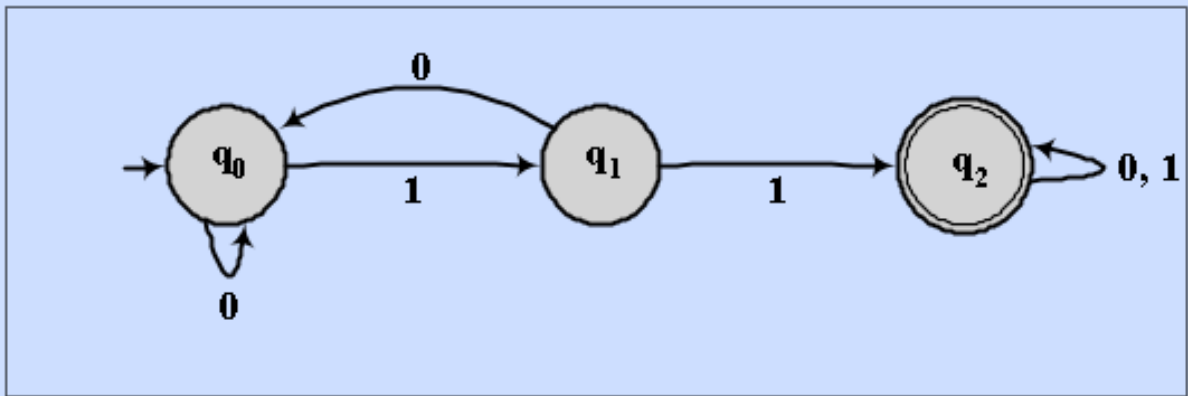
## Öncelikle Durumlar ve Final State (Final Durumu Çizilir)



Geçiş durumlarına göre durumlar arası bağlantılar sağlanır ve ilgili makinenin geçiş çizeneği elde edilir.



### Geçiş Çizeneği



## DFA'nın tanıdığı Dizgiler Kümesi:

Tanımlanan modele göre DFA'nın girişine her sferinde bir giriş simgesi uygulanır. Her giriş simgesi DFA'nın bir sonraki durumuna geçmesine neden olur. DFA, bulunduğu durumu yeni bir giriş simgesi uygulanana kadar korur.

Uygulanan giriş simgelerinin sayısı ne kadar çok olursa olsun deterministik olduğu için model çalışır.

Başlangıçta  $q_0$  durumunda bulunan DFA, belirli sayıda giriş simgesinden oluşan bir dizginin (string) uygulanmasından sonra belli bir duruma ulaşır.

Uygulanan giriş dizgisinin  $w$ , bu dizgi uygulandıktan sonra DFA'nın ulaştığı durumu da  $q_i$  ile göstereyim.  $q_0, w, q_i$  arasındaki ilişkiyi durum geçiş işlevinin kullanılarak ve bu işlevin anlamını biraz genişleterek aşağıdaki gibi gösterebiliriz.

$$\delta(q_0, w) = q_i$$

$q_0$  durumunda başlayan ve  $w$  giriş dizgisinin uygulanmasından sonra  $q_i$  durumuna ulaşan DFA eğer  $q_i$  bir uç durum ise  $w$  giriş dizgisini tanır. Giriş alfabesi  $\Sigma = \{I_1, I_2, \dots, I_k\}$  olan bir DFA'ya uygulanabilecek giriş dizgileri kümesi sonsuz bir kümedir. Giriş alfabesindeki simgelerden oluşan sonlu ya da sonsuz uzunluktaki her dizgi bu kümeden yer alır. Giriş alfabesindeki simgelerden oluşan dizgilerin bir kısmı DFA tarafından tanınan diğer bir kısmı ise tanınmayan dizgilerdir. Bu aşamada bir sonlu özdevinirin (M) tarafudan tanınan dizgiler kümesi aşağıdaki gibi tanımlamak mümkündür:

$$T(M) = \{w \mid \delta(q_0, w) = q_i \in F\}$$

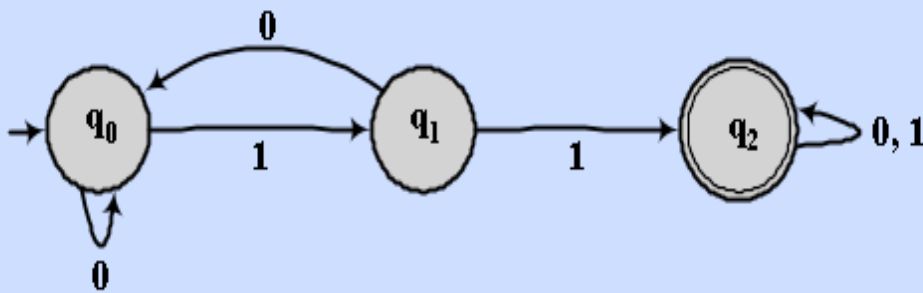
Buna göre DFA ya uygulanabilecek dizgiler kümesi DFA tarafından tanına ve tanımayan olmak üzere iki alt kümeye ayrılabilir.

DFA nın tanıdığı küme başlangıç durumundan bir uç duruma götüren dizgiler kümesidir.

Örneğin örnekte göstermiş olduğumuz  $M_{1,1}$  makinesi için 0101 dizgisinin  $M_{1,1}$  için tanınmadığı görülür.

Çünkü  $q_0$  durumunda başlayan M makinesi 0101 uygulandıktan sonra  $q_1$  durumuna ulaşır. Bu durum bir uç durum olmadığından dolayı makine bu dizgiyi tanımaz.

0110 dizgisinde ise makine en son olarak  $q_2$  durumuna ulaşır ve makine bu dizgiyi tanır.



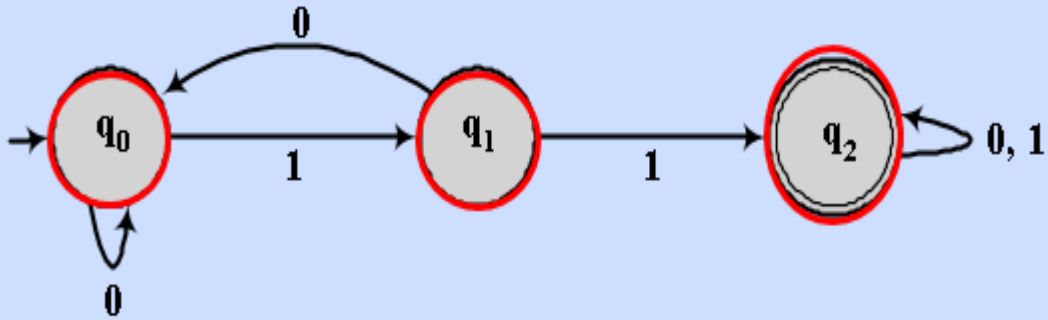
**DFA'nın Tanıdığı Dizgiler Kümesi**

$$T(M) = \{ w \mid \delta(q_0, w) = q_i \in F \}$$

$$T(M_{1,1}) = \{ 11, 011, 110, 0110, 0110, 01011, \dots \}$$

$T(M_{1,1})$  sözlü olarak,  $\{ 0, 1 \}$  alfabesinde, içinde 11 altdizgisi bulunan dizgiler kümesi olarak tanımlanabilir.

**Örnek 2:**  $w=01001100$  dizgisinin  $M_{1,1}$  Makinesinde tanınması



**$w = 0100110$**



$\delta(q_0, 0100110) = q_2$  olduğu ve  $q_2$  bir uç durum olduğu için 0100110  
 $M_{1,1}$  tarafından tanınır.

$\delta(q_0, 100101) = q_1$  olduğu ve  $q_1$  bir uç durum olmadığı için 100101  
 $M_{1,1}$  tarafından tanınmaz.