

BM 305 Biçimsel Diller ve Otomatlar (Formal Languages and Automata)

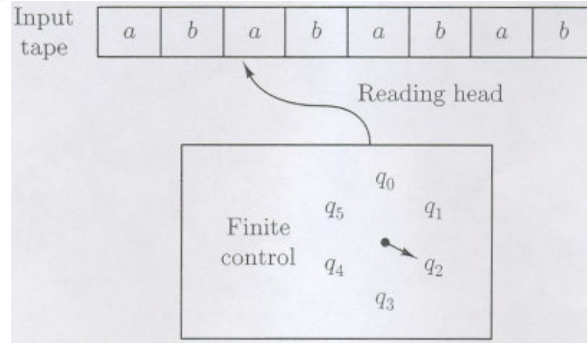
Hazırlayan: M.Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü



Deterministic Finite Automata

- Bir **finite automaton**(**automata çoğul**) sabit ve sonlu kapasiteye sahip bir merkezi işlem ünitesine sahiptir
- Giriş bilgisini string olarak bir **input tape** üzerinden alır.
- Bir çıkış üretmez, giriş bilgisinin kabul edilip edilmediğini gösterir.
- **Language recognition device** (**dil tanıyıcı cihaz**) olarak işlem yapar.
- Compiler'da **lexical analysis**(begin, +, for gibi program birimleri belirlenir) aşamasında kullanılır.
- Protokol tanımlamalarında kullanılır.

Deterministic Finite Automata



- Giriş bilgisi input tape üzerinden string olarak elde edilir.
- Makinenin ana kısmı "black box" tır ve sonlu sayıda farklı duruma sahiptir.
- Bu black box finite control olarak adlandırılır ve hareketli **reading head(okuma kafası)** ile input tape üzerinde herhangi bir pozisyonda bulunan sembolü algılar.
- Başlangıçta okuma kafası en soldaki kare üzerinde bulunur ve finite control **initial state(başlangıç durumu)** durumundadır.

Deterministic Finite Automata

- Automaton her seferinde input tape'ten bir sembol okur ve yeni bir duruma geçer.
- **Yeni duruma geçme sadece ve sadece mevcut durum ile okunan sembole bağlıdır !!!**
- Bu yüzden **deterministic finite automaton(DFA)**'dur.
- Her okumadan sonra okuma kafası bir sağa geçer ve string sonuna kadar okuma devam eder.
- Eğer string bittiğinde DFA **final state(s) (sonuç durumları)** lerden birisinde ise bu string kabul edilir.
- Bu DFA tarafından kabul edilen dil, kabul edilen stringler kümesidir.

Deterministic Finite Automata

Definition:

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

K sonlu sayıda durumlar kümesi

Σ alfabe

δ transition function (geçiş fonksiyonu) $K \times \Sigma$ dan K ' ya

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

$F \subseteq K$ final state(s) kümesi

- M otomatının sonraki duruma geçişi transition function ile belirlenir.
- Eğer M otomatı $q \in K$ durumunda iken input tape'ten $a \in \Sigma$ okumuşsa, $\delta(q, a) \in K$ unique (tek) durumuna geçer.

Deterministic Finite Automata

Örnek:

M bir DFA ve $M = (K, \Sigma, \delta, s, F)$ şeklinde tanımlanmıştır.

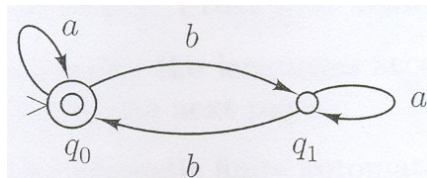
$K = \{q_0, q_1\}$,

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

$s = q_0$

$F = \{q_0\}$

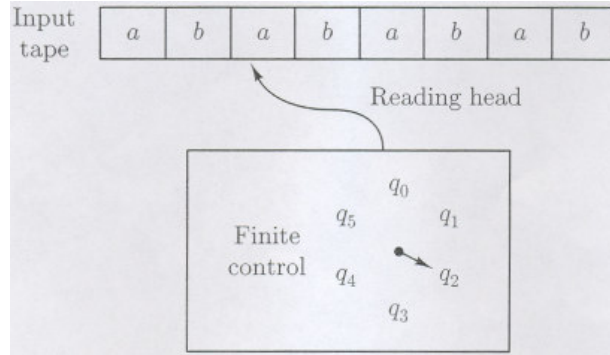
q	σ	$\delta(q, \sigma)$
q_0	a	q_0
q_0	b	q_1
q_1	a	q_1
q_1	b	q_0



$L(M)$ içerisinde çift sayıda b bulunduran tüm stringlerin kümesidir.

Deterministic Finite Automata

- **Configuration**, otomatın herhangi bir andaki durumu ile input tape'te sağ kısımdaki string'i(okunmamış) ifade eder.
- Configuration $K \times \Sigma^*$ 'ın bir elemanıdır. Aşağıdaki otomat için konfigürasyon $(q_2, ababab)$ 'dır.



- \vdash_M ardarda iki konfigürasyon arasında binary relation'ı ifade eder.

Deterministic Finite Automata

- (q, w) ve (q', w') ardarda iki konfigürasyon ise $(q, w) \vdash_M (q', w')$ şeklinde belirtilir.
burada $w = aw'$, $a \in \Sigma$ ve $\delta(q, a) = q'$ olmak zorundadır.
- \vdash_M fonksiyonu $K \times \Sigma^+$ 'dan $K \times \Sigma^*$ 'ya bir fonksiyondur.
- (q, e) konfigürasyonu giriş string'inin sonunu gösterir ve otomat işlemini bitirir.
- \vdash_M fonksiyonunun reflexive, transitive closure'u \vdash_M^* şeklinde tanımlanır.
- 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 $L(M)$ olarak gösterilir ve tüm kabul edilen string'ler kümesidir.

Deterministic Finite Automata

Örnek:

M bir DFA ve $M = (K, \Sigma, \delta, s, F)$ şeklinde tanımlanmıştır.

$K = \{q_0, q_1\}$,

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

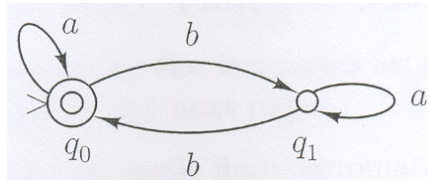
$s = q_0$

$F = \{q_0\}$

q	σ	$\delta(q, \sigma)$
q_0	a	q_0
q_0	b	q_1
q_1	a	q_1
q_1	b	q_0

$L(M)$ içerisinde çift sayıda b bulunduran tüm stringlerin kümesidir.

$(q_0, aabba) \vdash_M (q_0, aabba)$
 $\vdash_M (q_0, bba)$
 $\vdash_M (q_1, ba)$
 $\vdash_M (q_0, a)$
 $\vdash_M (q_0, e)$



$(q_0, aabba) \vdash_M^* (q_0, e)$ olduğu için $aabba$ M tarafından kabul edilir.

Deterministic Finite Automata

Örnek:

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

$L(M) = \{w \in \{a, b\}^* : w \text{ ardarda üç tane } b \text{ bulundurmaz}\}$ dilini tanır.

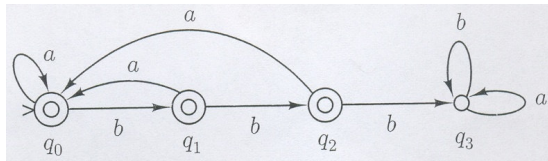
$K = \{q_0, q_1, q_2, q_3\}$,

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

$s = q_0$

$F = \{q_0, q_1, q_2\}$

q	σ	$\delta(q, \sigma)$
q_0	a	q_0
q_0	b	q_1
q_1	a	q_0
q_1	b	q_2
q_2	a	q_0
q_2	b	q_3
q_3	a	q_3
q_3	b	q_3

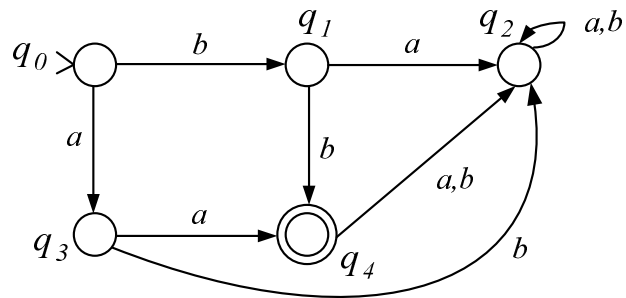


burada q_3 **dead state** olarak adlandırılır.

Deterministic Finite Automata

Örnek:

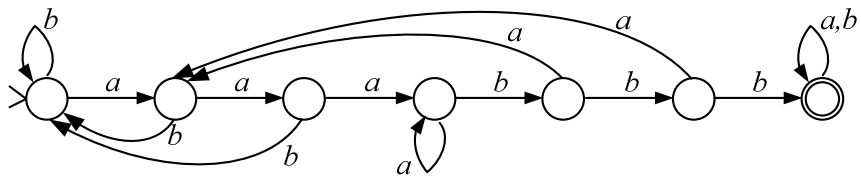
Aşağıdaki otomatın tanıdığı dili regular expression ile ifade ediniz.



Deterministic Finite Automata

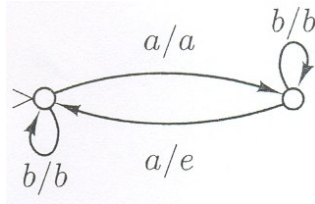
Örnek:

$L(M) = \{w \in \{a, b\}^* : w \text{ içerisinde } aaabbb \text{ substring' i bulunur}\}$



Deterministic Finite Automata

- **Deterministic finite-state transducer**, giriş string'ini çıkış string'ine çeviren bir sonlu otomattır.
- Transducer her durum geçişinde output tape üzerine bir sembol, boş sembol veya string yazar.
- a/w girişten a okur ve çıkışta w yazar.

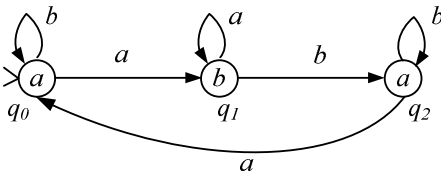


- Yukarıdaki otomat bütün b 'leri geçer ve her iki a 'dan ikincisinin yerine e yazar.

Deterministic Finite Automata

- Deterministic finite-state transducer, moore ve mealy makinesi olarak iki türdedir.

Moore makinesi



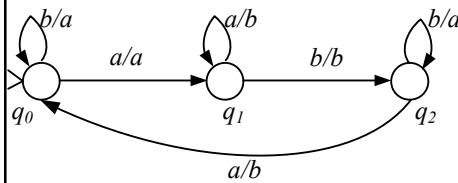
state	input	next state	output
q_0	a	q_1	a
q_0	b	q_0	a
q_1	a	q_1	b
q_1	b	q_2	b
q_2	a	q_0	a
q_2	b	q_2	a

Giriş	a	b	b	a	a	a	b	
Durum	q_0	$\longrightarrow q_1$	$\longrightarrow q_2$	$\longrightarrow q_2$	$\longrightarrow q_0$	$\longrightarrow q_1$	$\longrightarrow q_1$	$\longrightarrow q_2$
Çıkış	a	b	a	a	a	b	b	a



Deterministic Finite Automata

Mealy makinesi



state	input	next state	output
q_0	a	q_1	a
q_0	b	q_0	a
q_1	a	q_1	b
q_1	b	q_2	b
q_2	a	q_0	b
q_2	b	q_2	a

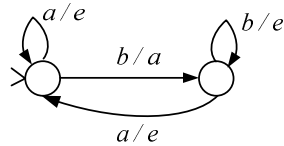
Giriş	a	b	b	a	a	a	b
Durum	$q_0 \rightarrow q_1$	$\rightarrow q_2$	$\rightarrow q_2$	$\rightarrow q_0$	$\rightarrow q_1$	$\rightarrow q_1$	$\rightarrow q_2$
Çıkış	a	b	a	b	a	b	b



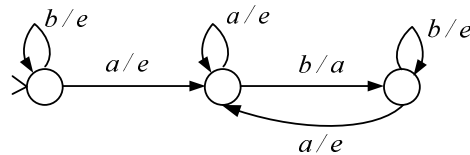
Deterministic Finite Automata

Örnek: (2.1.4.a(i))

- w giriş için a^n üreten transducer durum çizeneğini oluşturunuz. n sayısı a 'nın tekrar sayısını gösterir ve w içinde ab substring'inin sayısına eşittir.



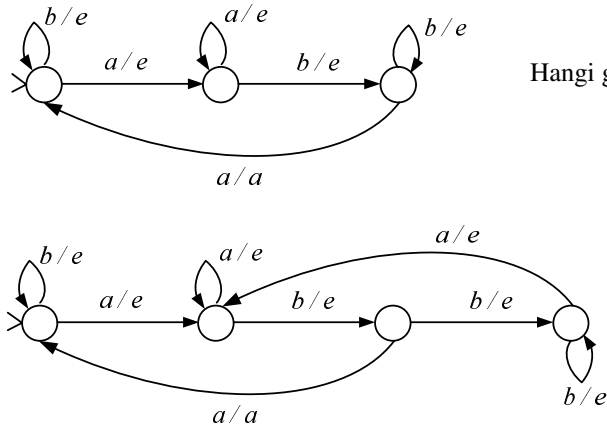
Hangi girişler için hatalı çıkış üretir ?



Deterministic Finite Automata

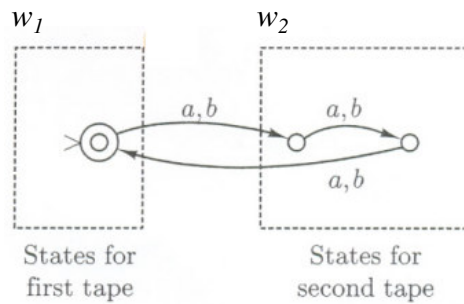
Örnek: (2.1.4.a(ii))

- w giriş için a^n üreten transducer durumu çizeneğini oluşturunuz. n sayısı a 'nın tekrar sayısını gösterir ve w içinde aba substring'inin sayısına eşittir.



Deterministic Finite Automata

- **Deterministic 2-tape finite automaton**, string çiftlerini kabul eden ve iki girişe sahip olan sonlu otomattır.
- Transition function 1.tape ve 2.tape için ayrı ayrı tanımlıdır.



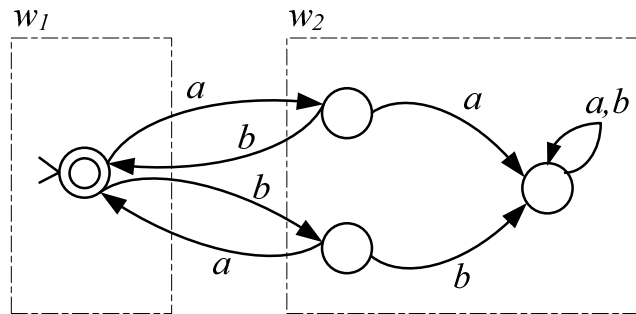
- $(w_1, w_2) \in \{a, b\}^* \times \{a, b\}^*$ stringlerini $|w_2| \geq 2 |w_1|$ olması koşuluyla kabul eder.



Deterministic Finite Automata

Örnek:

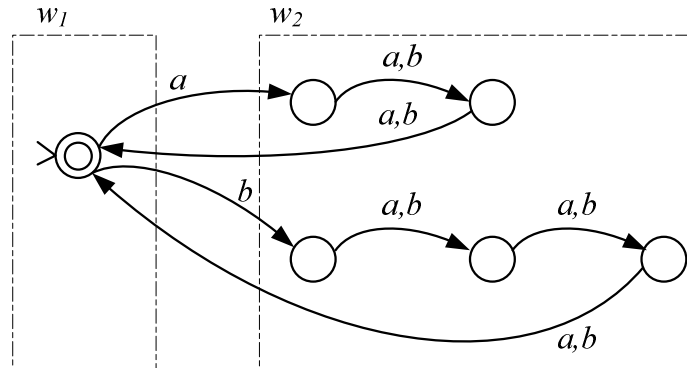
Bir 2-tape sonlu otomat, tüm $(w_1, w_2) \in \{a, b\}^* \times \{a, b\}^*$ string çiftlerini $|w_1| \leq |w_2|$ olması ve tüm i 'ler için $w_1(i) \neq w_2(i)$ olması koşuluyla kabul eder.



Deterministic Finite Automata

Örnek:

Bir 2-tape sonlu otomat, tüm $(w_1, w_2) \in \{a, b\}^* \times \{a, b\}^*$ string çiftlerini $|w_2|$ değerinin, w_1 içindeki a sayısının iki katı artı b sayısının üç katına eşit veya büyük olması koşuluyla kabul eder.





Ödev

- Problemleri çözünüz 2.1.2, 2.1.3, 2.1.7 (sayfa 60-63)