

Ders Planı

- 1- Sonlu Otomatlar
- 2- Düzgün Kümeler, Düzgün Deyimler
- 3- Dilbilgisi ve Diller
- 4- Bağlamdan bağımsız dilbilgisi
- 5- Yığıtlı Otomatlar
- 6- Turing Makinesi

Kaynak:

- Özdevinirler (Otomatlar) Kuramı ve Biçimsel Diller, Ünal Yarımağan
- <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2011/lecture-notes/>

Hesaplama Teorisi

- Teorik bilgisayar bilimlerinde hesaplama teorisi, bir algoritma kullanarak bir hesaplama modelinde problemlerin çözülüp çözülmeceğini ve ne kadar verimli bir şekilde çözülebileceğini ele alan bir daldır
- Alan üç ana branşta ayrılmıştır:
 - otomat teorisi
 - hesaplanabilirlik teorisi
 - hesaplama karmaşıklık teorisi
- Sıkı bir hesaplama çalışması yapmak için bilgisayar bilimcileri, bir hesaplama modeli adı verilen bilgisayarların matematiksel bir soyutlaması ile çalışırlar. Kullanımda birçok model var, ancak en sık incelenen Turing makinesi.
- Otomata teorisi: soyut makinelerin (veya daha uygun bir şekilde, soyut 'matematiksel' makinelerin veya sistemlerin) ve bu makineleri kullanarak çözülebilecek hesaplama problemlerinin incelenmesidir. Bu soyut makinelere otomata denir.
- Bir otomat:
 - durumlar (şekilde daireler ile temsil edilir),
 - geçişler (oklarla gösterilir).
- Otomat bir giriş sembolü gördüğünde, başka bir duruma geçiş (veya zıplama) yapar, geçiş işlevine göre.

Sonlu Otomata (Finite Automata)

- Birçok önemli yazılım ve donanım türü için kullanışlı bir modeldir:
 - Dijital devrelerin davranışını tasarlama ve kontrol etme yazılımı
 - Tipik bir derleyicinin, yani metin metnini mantıksal birimlere bölen derleyici bileşeninin sözcüksel çözümleyicisi
 - Kelimelerin, ifadelerin veya diğer kalıpların oluşumlarını bulmak için Web sayfalarının koleksiyonları gibi büyük metin gövdelerini taramaya yönelik yazılım
 - Güvenli bilgi alışverişi için protokollerin iletişim protokolleri gibi sınırlı sayıda farklı durumu olan her türden sistemi doğrulamak için yazılım

0 –Kümeler, Alfabe, Dil

Bir küme nesnelerden oluşur

- $K = \{a, b, c, d\}$
 - a, b, c, d kümenin elemanları veya üyeleridir
 - $c \in K, k \notin K$

Kümeler özellikleriyle de tanımlanabilir

- $G = \{x \mid x \in I \text{ and } x \text{ is greater than } 2\}$
 - 2 den büyük sayılar
- $O = \{x \mid x \in N \text{ and } x \text{ is not divisible by } 2\}$ tek sayılar kümesi
 - Tek sayılar kümesi

Dil

- Diller iki ana başlık altında incelenebilir:
- **Doğal (Natural) Diler**
 - Kelimelerin, eklerin ve benzeri dil yapılarının sonlu dizini.
- **Yapay (Artificial) Diller**
 - Bilgisayarların işlem yapmalarına olanak sağlayan semboller dizini (örn. programlama dilleri)

Alfabe

- Bir alfabe sonlu sayıda sembolderen oluşur
- *Roman alfabesi* = $\{a, b, \dots, z\}$
- *Binary alfabe* = $\{0, 1\}$
- String bir alfabede tanımlanan sonlu sayıda sembolün sıralanışıyla elde edilir
- *"bilgisayar"* Roman alfabesinde tanımlanmış bir string.
- *"0111011"* Binary alfabede tanımlanmış bir string.
- *String == Dizgi*

Stringler Üzerinde Tanımlar

- Bir alfabe'deki her sembol bir stringtir.
- Sembol içermeyen String, empty-string olarak adlandırılır ve “*e*” *sembolüyle ifade edilir.*
- String olarak genellikle *u, v, w, x, y, z harfleri kullanılacaktır.*
- $w = abc$
- Σ alfabe için Σ^* ise bu alfabede oluşturulan boş String dahil tüm String’lerin kümesini göstermektedir
- Bir String’in length (uzunluk) değeri $|w|$ *şeklinde gösterilir.*
- $|101| = 3, |acrd|=4, |e| = 0$
- $w = bilgisayar, w(2) = w(5)= i, w(1) = b$

Concatenation (Birbirine bağlama)

- İki string'in artarda eklenmesidir
 - $w = xoy$, $w = xy$
 - $w(j) = x(j)$, $j=1,\dots, |x|$ & $w(|x| + j) = y(j)$, $j = 1,\dots, |y|$
 - $|w| = |x| + |y|$ ortaya çıkan sonuç String'inin uzunluğu eklenen String'lerin uzunluklarının toplamına eşittir.
- Örnek:
- $010001 = 01001$
- $\text{bilgisayaromühendisliği} = \text{bilgisayarmühendisliği}$
- $woe = eow = w$
- $(wx)y = w(xy)$

Reversal (Tersine çevirme)

- Bir String w için w^R reversal (tersi) olarak adlandırılır.
- $|w| = |w^R|$ bir String'inin tersinin uzunluğu kendi uzunluğuna eşittir.
- $|w| = k+1 > 0$ ve $a \in A$ ve $w \in A^*$ iken
- $x = wa$ ise $x^R = (wa)^R = a \circ w^R$
- $(x \circ y)^R = y^R \circ x^R$
- Örnekler:
 - $(reverse)^R = esrever$
 - $e^R = e$
 - $w = bilgisayar, w^R = rayasiglib$
 - $(kedi \circ köpek)^R = (köpek)^R \circ (kedi)^R = kepökidek$

Prefix ve Suffix

- Bir String v , w String'i içinde substring olarak belirtilir.
- $w = xvy$, x ve y empty string olabilir
- Eğer bazı x 'ler için $w = xv$ ise String v , String w içinde suffix (sonek) olarak adlandırılır.
- Eğer bazı y 'ler için $w = vy$ ise String v , String w içinde prefix (önek) olarak adlandırılır.
 - road
 - roadrunner 'da prefix,
 - abroad 'da suffix ve
 - broader 'da substring'tir
- w^i bir string'in i kez tekrarını gösterir
- $w^0 = e$ empty string
- $w^{i+1} = w^i \circ w$, $i \geq 0$
- $(do)^2 = dodo$

Alfabe ve Dil İlişkisi

- Bir alfabe Σ üzerinde tanımlı String kümesi, Σ^* kümesinin altkümesidir ve language (dil) olarak adlandırılır.
- Σ , Σ^* ve \emptyset birer dildir.
- *$\{aba, cza, d, f\}$ bir sonlu dildir ve $\Sigma = \{a, b, c, \dots, z\}$ alfabesi üzerinde tanımlıdır*
- *$\{0, 01, 011, 0111, \dots\}$ bir sonsuz dildir ve $\Sigma = \{0, 1\}$ alfabesi üzerinde tanımlıdır*

Alfabe ve Dil İlişkisi

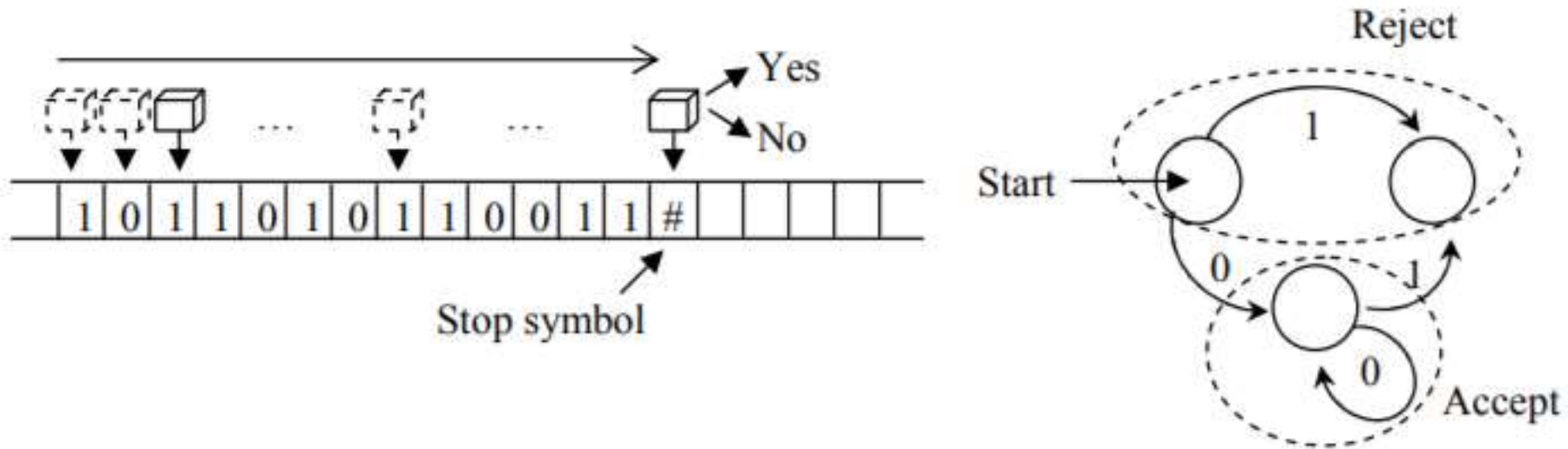
- $\{a,b,c,d\}$ ve $\{0,1\}$ alfabelerinde oluşturulmuş düzenli ifadeler ve bunlara karşı gelen kümeler şu şekillerde gösterilebilir:
- $a(b \cup c) \rightarrow \{ab, ac\}$
- $(00 \cup 11) \rightarrow \{00, 11\}$
- $a^* \rightarrow \{e, a, aa, aaa, aaaa, aaaaa, aaaaaa, \dots\}$
- $ab^* \rightarrow \{a, ab, abb, abbb, abbbb, abbbbbb, \dots\}$
- $(0 \cup 1)^* \rightarrow \{e, 0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, \dots\}$
- $a(b \cup cd^*)^*a \rightarrow \{aa, aba, acda, acdda, acddda, abca, acba, \dots\}$

Finite Automata (Sonlu Otomatlar)

- Formal bir sistem
- Sonlu miktarda bilgi tutar
- Bilgi, durumlarla (state) ifade edilir
- Bir State'den diğer bir state'e geçiş, inputlara (girdi) göre değişir
- Farklı durumlarda farklı inputlar farklı durumlara sevk eder, buna state transition (durum geçiş) denir. Bu kurallar geçiş fonksiyonları olarak adlandırılır.

Finite Automata (Sonlu Otomatlar)

Soyut Makine Modeli

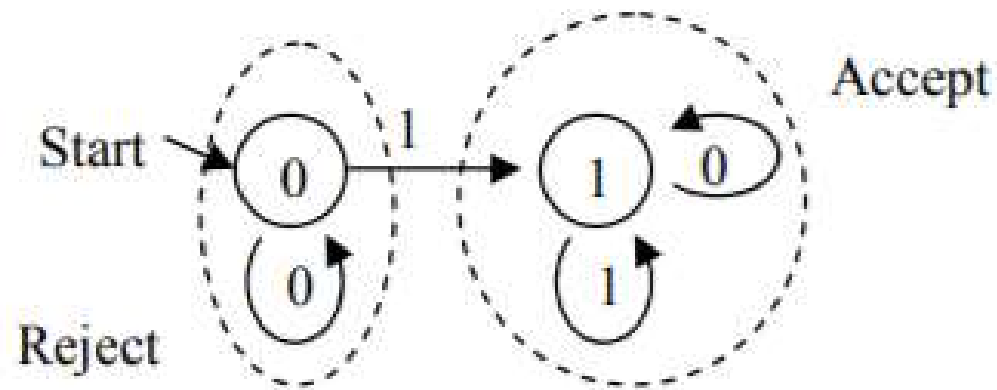


- 0 ve 1 alfabesi – input semboller
- Bunlardan kelimeler türetelim ve bu kelimeleri yukarıda sağdaki otomatin kabul edip etmeyeceğini kontrol edelim.
- Sağdaki şekil otomatımız
- Soldaki şekil ise otomatanın çalışma prensibini gösteren –bir şerit ve okuma kafasından oluşan- soyut bir model.

Sonlu Otomatlara Örnekler

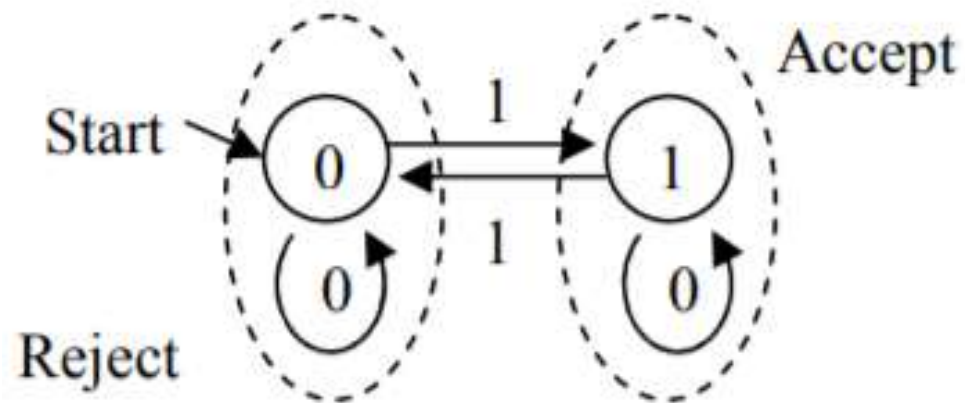
Örnek-1

- Aşağıdaki Otomata Hangi tür kelimeleri tanır (kabul eder)?



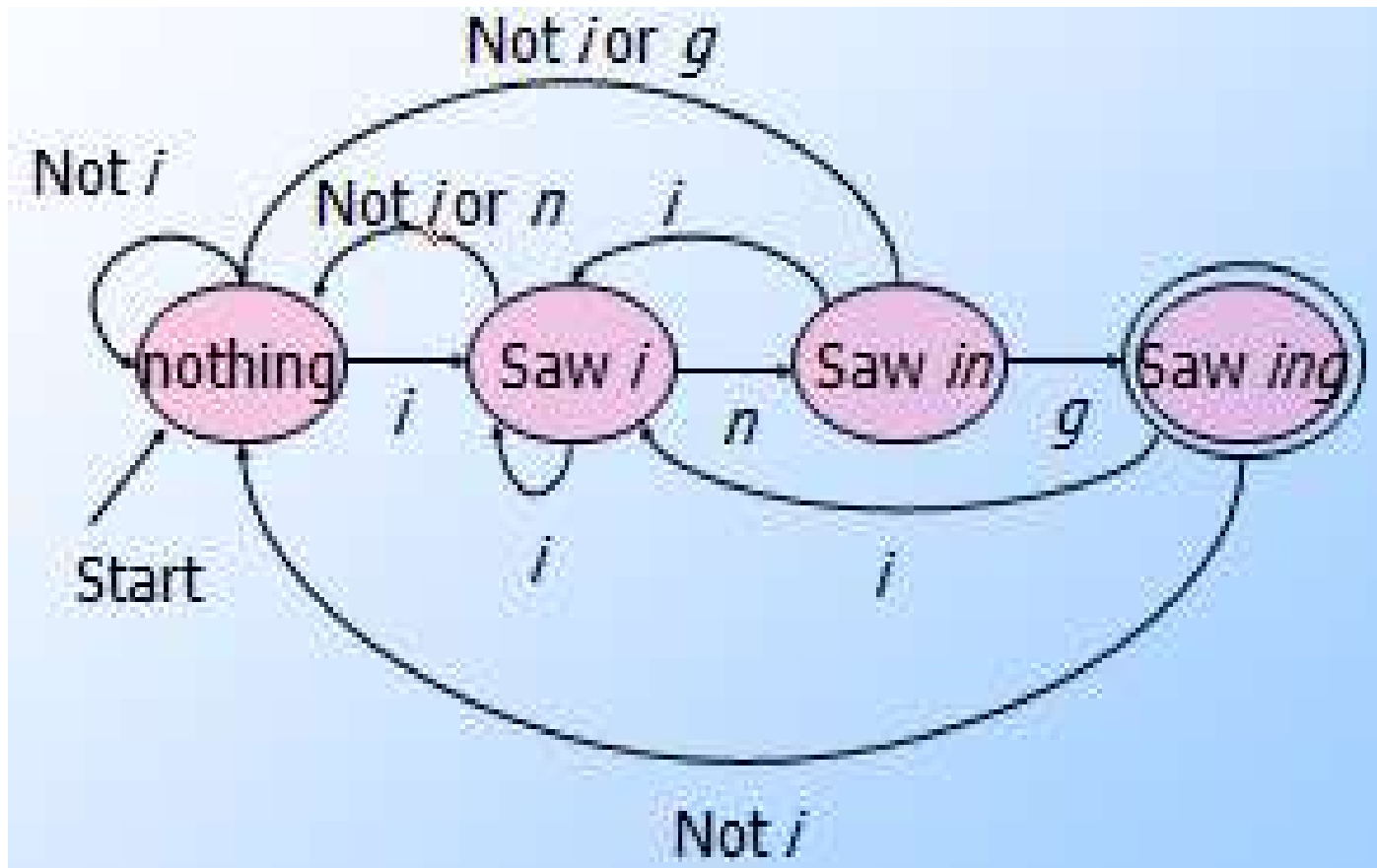
Örnek-2

- Aşağıdaki Otomata Hangi tür kelimeleri tanır (kabul eder)?



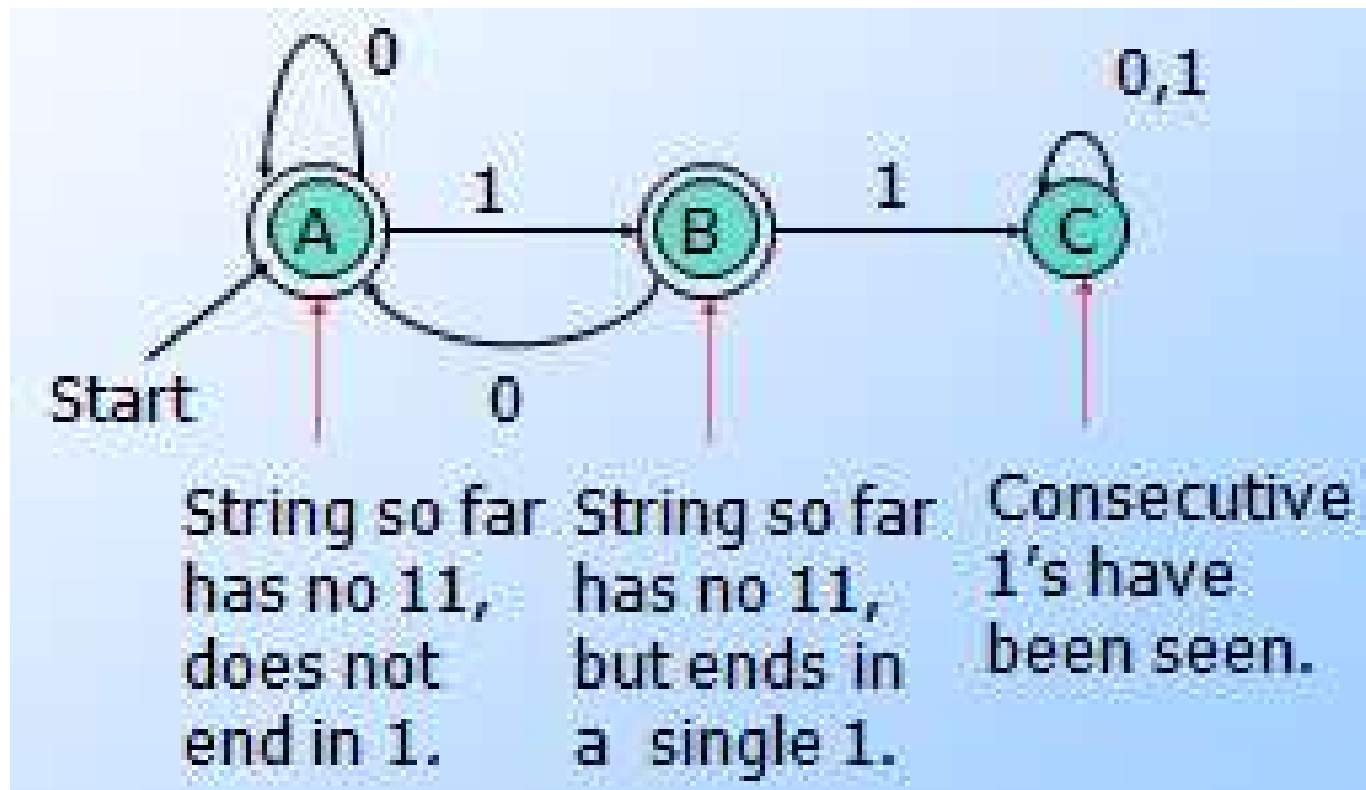
Örnek-3:

-ing ile Biten Stringleri Tanıyan Automata



Örnek-4:

İçinde 11 Olmayan Stringleri Tanıyan Otomata



Örnek-5: Tenis Oyunu Otomatası

