

PAMUKKALE ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ  
2021 BAHAR

# Biçimsel Diller ve Otomata Teorisi

## Formal languages and automata theory

ALFABE ve KATAR-1

---

# Alphabets and Languages

- Bir **alfabe** sonlu sayıda sembolder oluşur

Örnek: *Roman alfabesi* =  $\{a, b, \dots, z\}$     *Binary alfabe* =  $\{0, 1\}$

- **String** bir alfabede tanımlanan sonlu sayıda sembolün sıralanışıyla elde edilir

"bilgisayar" Roman alfabesinde tanımlanmıştır.

"0111011" Binary alfabede tanımlanmıştır.

- Bir alfabedeki her sembol bir string'tir
- Sembol içermeyen string **empty string** olarak adlandırılır

# Alphabets and Languages

- String olarak genellikle  $u, v, w, x, y, z$  harfleri kullanılacaktır.  $w = abc$
- $\Sigma$  alfabeti için  $\Sigma^*$  ise bu alfabede oluşturulan boş string'te dahil tüm string'lerin kümesini göstermektedir (\*: Kleene Star)
- Bir string'in **length (uzunluk)** değeri  $|w|$  şeklinde gösterilir

$$|101| = 3, \quad |e| = 0,$$

- $w = \text{bilgisayar}, \quad w(3) = 1, \quad w(1) = b$
- **concatenation** iki string'in ardarda eklenmesidir

$$w = x \bullet y, \quad w = xy$$

$$|w| = |x| + |y|$$

$$w(j) = x(j), \quad j=1, \dots, |x| \quad w(|x| + j) = y(j), \quad j = 1, \dots, |y|$$

$$01 \bullet 001 = 01001$$

$$\text{bilgisayar} \bullet \text{mühendisligi} = \text{bilgisayarmühendisligi}$$

$$w \bullet e = e \bullet w = w$$

# Concatenation (o)

$$x \circ y = xy$$

- Kaynaştırma ilişkisellik özelliğine (associativity) sahiptir:  $x(yz) = (xy)z$
- Kaynaştırma değişme özelliğine (commutativity) sahip değildir:  $xy \neq yx$
- Kaynaştırmanın birleşme üzerine dağılma özelliği vardır  $x(yuz) = xy \cup xz$
- Kaynaştırmanın kesişim üzerine dağılma özelliği yoktur  $x(ynz) \neq xy \cap xz$

# Alphabets and Languages

- Bir string  $v$ ,  $w$  string'i içinde **substring** olarak belirtilir.

$w = xvy$ ,  $x$  ve  $y$ ,  $e$  (empty) string olabilir

- Eger bazı  $x$ 'ler için  $w = xv$  ise string  $v$ , string  $w$  içinde **suffix** olarak adlandırılır.
- Eger bazı  $y$ 'ler için  $w = vy$  ise string  $v$ , string  $w$  içinde **prefix** 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)

$w^{i+1} = w^i \circ w$ ,  $i \geq 0$

$ab^2 = abab$

$ab^* = \{a, ab, abb, \dots\}$



$(ab)^* = \{e, ab, abab, ababab, \dots\}$

# Alphabets and Languages

- Bir string  $w$  için  $w^R$  **reversal** olarak adlandırılır.

$w = bilgisayar, w^R = rayasiglib$

- Bir alfabe  $\Sigma$  üzerinde tanımlı string kümesi,  $\Sigma^*$  kümesinin altkümesidir ve **language (dil)** olarak adlandırılır.
- $\Sigma, \Sigma^*$  ve  $\emptyset$  birer dildir.

$\{aba, czr, d, f\}$  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

# Alphabets and Languages

- Bir dil  $L = \{w \in \Sigma^* : w \text{ string'i } P \text{ özelliğine sahiptir}\}$  şeklinde tanımlanır

$$L = \{w \in \{0, 1\}^* : w \text{ eşit sayıda 1 ve 0'a sahiptir}\}$$

$$L = \{w \in \Sigma^* : w = w^R\}$$

- Bir alfabe'de tanımlı iki dil arasında kaynaştırma (concatenation)

tanımlanabilir

$$L = L_1 \circ L_2 \quad L = L_1 L_2$$

$$L = \{w \in \Sigma^* : w = xoy \text{ ve } x \in L_1 \text{ ve } y \in L_2\} \text{ Örnek:}$$

$$\Sigma = \{0, 1\}, \quad L_1 = \{w \in \Sigma^* : w \text{ çift sayıda 0'a sahip}\} \quad \text{💬}$$

$$L_2 = \{w \in \Sigma^* : w \text{ 0'la başlar ve 1'le devam eder}\} \quad L_1 \circ L_2 = \{w \in \Sigma^* : w \text{ tek sayıda 0'a sahiptir}\} \quad \text{💬}$$

# Alphabets and Languages

- **Kleen star** işlemi  $L^*$  ile gösterilir ve bir dilde 0 veya daha fazla string'in concatenate edilmesiyle elde edilir.

$$L^* = \{w \in \Sigma^*: w = w_1 o \dots o w_k \quad k \geq 0 \text{ ve } w_1 o \dots o w_k \in L\} \quad \text{Örnek:}$$

$$L = \{01, 1, 100\}$$

$$110001110011 \in L^* \quad 1o100o01o1o100o1o1$$

- $L^+$  işlemi  $LL^*$  dilini ifade etmek için kullanılır.

$$L^+ = \{w \in \Sigma^*: w = w_1 o \dots o w_k \quad k \geq 1 \text{ ve } w_1 o \dots o w_k \in L\}$$

$L^+$  dili,  $L$  dilinin closure'u olup  $L$  dilini ve  $L$  dilindeki string'lerin eklenmesiyle elde edilen tüm string'leri içeren bir dildir.



# Alphabets and Languages

- $L = \{w \in \{0, 1\}^*: w \text{ iki veya üç tane } 1 \text{ tekrarını içerir ve birinci ile ikinci ardarda gelemez}\}$  şeklinde tanımlı bir dil olsun.

Bu dil sadece singleton kümelerle ve  $\cup$ ,  $\circ$ ,  $*$  ile ifade edilebilir.

$$L = \{0\}^* \circ \{1\} \circ \{0\}^* \circ \{0\} \circ \{1\} \circ \{0\}^* \circ ((\{1\} \circ \{0\}^*) \cup 0^*)$$

$L = 0^*10^*010^*(10^* \cup 0^*)$  şeklinde  $\{, \}$  ve  $\circ$  kaldırılarak kısaca gösterilebilir

- Yukarıdaki ifade **regular expression (düzenli ifade)** olarak adlandırılır. Regular expression'lar bir dili doğrudan  $\cup$ ,  $\circ$ ,  $*$  ile tanımlar.