

Formal Languages

Formal Languages

- A **language** is a set of **strings/words**
- A **string** is a finite sequence of symbols

Example: “cat”, “dog”, “house”, ...

defined over an alphabet: $\Sigma = \{a, b, c, \dots, z\}$

Example: “aaaab”, “bbbbbb”, “ababababa”, ...

defined over an alphabet: $\Sigma = \{a, b\}$

Alphabets and Strings

- We usually use small alphabets:

$$\Sigma = \{a, b\}, \Sigma = \{0, 1\}$$

- Strings

a

u = ab

ab

v = bbbaaa

abba

w = abba

baba

aaabbbaabab

String Operations

 $w = a_1 a_2 \cdots a_n$

abba

 $v = b_1 b_2 \cdots b_m$

bbbaaa

Concatenation

 $wv = a_1 a_2 \cdots a_n b_1 b_2 \cdots b_m$

abbabbbaaa

String Operations

$w = a_1 a_2 \cdots a_n$

abbaabbb

Reverse

$w^R = a_n \cdots a_2 a_1$

bbbaabba

String Length

$$w = a_1 a_2 \cdots a_n$$

- **Length:** $|w| = n$
- **Examples:**
 - $|abba| = 4$
 - $|aa| = 2$
 - $|a| = 1$

Length of Concatenation

$$|uv| = |u| + |v|$$

- Example: $u = aab, |u| = 3$

$$v = abaab, |v| = 5$$

$$|uv| = |aababaab| = 8$$

$$|uv| = |u| + |v| = 3 + 5 = 8$$

Empty String

- A string without symbols: λ
- Observations: $|\lambda| = 0$

$$\lambda w = w\lambda = w$$

$$\lambda abba = abba\lambda = ab\lambda ba = abba$$

Substring

- A substring of a string is
a subsequence of consecutive symbols

String	Substring
abbab	ab
abbab	abba
abbab	b
abbab	ba
abbab	bbab

Prefix and Suffix

Prefixes

λ

a

ab

abb

abba

abbab

Suffixes

abbab

bbab

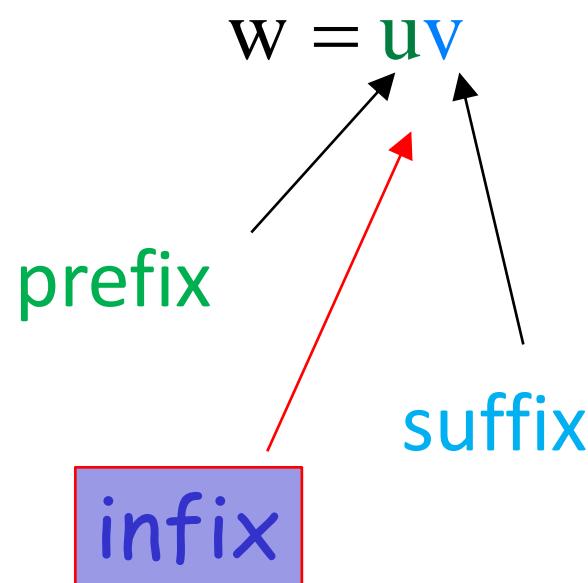
bab

ab

b

λ

abbab



Replication

$$w^n = \underbrace{ww \cdots w}_n$$

- **Example:** $(abba)^2 = abbaabba$

- **Definition:** $w^0 = \lambda$

$$(abba)^0 = \lambda$$

* Operation

- Σ^* : the set of all possible strings from alphabet Σ .
- Example:

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

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \dots\}$$

$^+$ Operation

- $\Sigma^+ : \text{the set of all possible strings from alphabet } \Sigma \text{ except } \lambda.$

- **Example:**

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

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \dots\}$$

$$\Sigma^+ = \Sigma^* - \{\lambda\}$$

$$\Sigma^+ = \{a, b, aa, ab, ba, bb, aaa, aab, \dots\}$$

Language

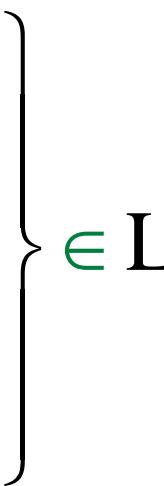
- A language is any subset of Σ^*
- Example: $\Sigma = \{a, b\}$
 $\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, \dots\}$
- Languages:
 - $\{\lambda\}$
 - $\{a, aa, aab\}$
 - $\{\lambda, abba, baba, aa, ab, aaaaaaa\}$

Language

- An **infinite** language $L = \{a^n b^n : n \geq 0\}$

λ
ab
aabb
aaaaabbfffff

$\in L$ abb $\notin L$



Operations on Languages

- The usual set operations: $\cup, \cap, -$

$$\{a, ab, aaaa\} \cup \{bb, ab\} = \{a, ab, bb, aaaa\}$$

$$\{a, ab, aaaa\} \cap \{bb, ab\} = \{ab\}$$

$$\{a, ab, aaaa\} - \{bb, ab\} = \{a, aaaa\}$$

$$\bar{L} = \Sigma^* - L$$

- Complement:

$$\overline{\{a, ba\}} = \{\lambda, b, aa, ab, bb, aaa, \dots\}$$

Reverse

- **Definition:** $L^R = \{w^R : w \in L\}$
 - **Examples:** $\{ab, aab, baba\}^R = \{ba, baa, abab\}$
- $L = \{a^n b^n : n \geq 0\}$
- $L^R = \{b^n a^n : n \geq 0\}$

Concatenation

- **Definition:** $L_1 L_2 = \{xy : x \in L_1, y \in L_2\}$
- **Example:**

$$\{a, ab, ba\} \{b, aa\} = \{ab, aaa, abb, abaa, bab, baaa\}$$

Replication

- **Definition:** $L^n = \underbrace{LL\cdots L}_n$

$$\{a, b\}^3 = \{a, b\}\{a, b\}\{a, b\} = \\ \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$$

- **Special case:** $L^0 = \{\lambda\}$

$$\{a, bba, aaa\}^0 = \{\lambda\}$$

Examples

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

$$L^2 = \{a^n b^n a^m b^m : n, m \geq 0\}$$

aabbaaabbb $\in L^2$

Star-Closure (Kleene *)

- **Definition:** $L^* = L^0 \cup L^1 \cup L^2 \dots$

- **Example:**

$$\{a, bb\}^* = \left\{ \lambda, a, bb, aa, abb, bba, bbbb, aaa, aabb, abba, abbbb, \dots \right\}$$

Positive Closure (Kleene $^+$)

- **Definition:** $L^+ = L^1 \cup L^2 \cup \dots = L^* - \{\lambda\}$

$$\{a, bb\}^+ = \left\{ \begin{array}{l} a, bb, \\ aa, abb, bba, bbbb, \\ aaa, aabb, abba, abbbb, \dots \end{array} \right\}$$

Exercises

1. Let $L = \{ab, aa, baa\}$. Which strings are in L^2, L^3 , and L^4 ?
2. Let $\Sigma = \{a, b\}$ and $L = \{aa, bb\}$. Use set notation to describe \bar{L} .
3. Prove that $(w^R)^R = w$ for all $w \in \Sigma^*$.
4. Prove that $(L_1 L_2)^R = L_2^R L_1^R$ for all languages L_1 and L_2 .