Concepts and Notations

- **Alphabet:** A finite, nonempty set of symbols. Conventionally, we use the symbol Σ for an alphabet.
  - Examples:
    - The set of all ASCII characters, or the set of all printable ASCII characters.
    - Σ_1 = { a, b }
    - Σ_2 = { Spring, Summer, Autumn, Winter }
    - Σ_3 = { 0, 1 }
- **String**: A finite sequence of zero or more symbols from an alphabet.
  - The empty string: ε
  - 01101 is a string from the binary alphabet Σ= { 0, 1 }

Concepts and Notations

- **Powers of an Alphabet**: If Σ is an alphabet, we denote by Σ^{κ} the set of all strings of length k.
  - Examples: Let $\Sigma = \{a, b, c\}$
    - $\Sigma^0 = \varepsilon$
    - $\Sigma^1 = \{ a, b, c \}$
    - $\Sigma^2 = \{ aa, ab, ac, ba, bb, bc, ca, cb, cc \}$
    - $\Sigma^3 = \{ aaa, aab, aac, aba, abb, abc, aca, acb, ... \}$

  Σ^* = The set of all strings over $\Sigma = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup ....$
  Σ^+ = The set of nonempty strings over $\Sigma = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup ....$
  $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$

- Exercise: Given $\Sigma = \{ 0, 1 \}$, compute Σ^+ and Σ^*.

Formal Language

- Language: A set of strings over an alphabet.
  - If Σ is an alphabet, and $L \subseteq \Sigma^*$, then L is a language over Σ.
  - Also known as a formal language.

- Examples:
  - The language of all strings consisting of n 0's followed by n 1's for some n>=0:

    $$\{\epsilon, 01, 0011, 000111, \ldots\}.$$

  - The set of string with equal numbers of 0's and 1's

    $$\{\epsilon, 01, 10, 0011, 0101, 1001, \ldots\}$$

  - The set of binary numbers whose value is a prime

    $$\{10, 11, 101, 111, 1011, \ldots\}$$

  - The empty language, denoted $\varnothing$, is a language over any alphabet.

Operations on Languages

- Suppose L_1 and L_2 are languages over some common alphabet.

- Union ($L_1 \cup L_2$): $\{w | w \in L_1 \vee w \in L_2\}$
- Concatenation ($L_1.L_2$): $\{w \cdot z | w \in L_1 \wedge z \in L_2\}$
- The Kleene Closure (L_1^*): $\{\epsilon\} \cup \{w \cdot z | w \in L_1 \wedge z \in L_1^*\}$

Regular Language

- Regular Languages are the simplest class of formal languages.

- Regular languages can be specified by
  - regular expressions (REs),
  - finite-state automata (FSAs),
  - regular grammars.