Lecture 2 — Languages and Grammars (Review)

CSE205: Theory of Computation

Minseok Jeon

2026 Fall

Overview

The build-up of Lecture 2 runs in one direction:

$$\text{symbol} \longrightarrow \text{alphabet} \longrightarrow \text{string} \longrightarrow \text{language} \longrightarrow \text{grammar}$$

Everything before *grammar* is build-up; grammar is the point of the lecture.

Review: Alphabet and String

Definition

An *alphabet* is a finite, non-empty set of symbols, denoted Σ.

Any symbols will do — finitely many, at least one. E.g. $\{0, 1\}$, $\{a, \ldots, z\}$, ASCII.

Definition

A *string* is a finite sequence of symbols chosen from an alphabet.

Notation you must read without a gloss:

- ε: the empty string
- wv: concatenation / w^R: reverse / $|w|$: length
- $w = vu$: v is a *prefix*, u a *suffix* of w

Review: Powers of an Alphabet

- $\Sigma^{\boldsymbol{k}}$: the set of strings over Σ of length $\boldsymbol{k}$ — always finite, e.g. $|\{0,1,2\}^3| = 27$.
- All strings, and all non-empty strings:

$$\Sigma^* = \bigcup_{\boldsymbol{k} \geq 0} \Sigma^{\boldsymbol{k}}, \qquad \Sigma^+ = \bigcup_{\boldsymbol{k} \geq 1} \Sigma^{\boldsymbol{k}}$$

- Since $\Sigma^0 = \{\boldsymbol{\varepsilon}\}$: $\quad \Sigma^* = \Sigma^+ \cup \{\boldsymbol{\varepsilon}\}$.

Review: Language and Its Operations

Definition

A *language* L is a set of strings: $L \subseteq \Sigma^*$ (equivalently, $L \in 2^{\Sigma^*}$).

Because a language *is* a set, all set operations carry over:

- $L_1 \cup L_2, \quad L_1 \cap L_2, \quad L_1 - L_2$
- reverse: $L^R = \{w^R \mid w \in L\}$
- complement: $\overline{L} = \Sigma^* - L$ (the universe is Σ^*)
- concatenation: $L_1 L_2 = \{xy \mid x \in L_1 \wedge y \in L_2\}$

Review: Grammar

Definition

A *grammar* is a quadruple $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$:

- $\boldsymbol{V}$: a finite set of *variables* (non-terminals)
- $\boldsymbol{T}$: a finite set of *terminal* symbols
- $\boldsymbol{S} \in \boldsymbol{V}$: the *start variable*
- $\boldsymbol{P}$: a finite set of *productions* $\boldsymbol{x} \to \boldsymbol{y}$ with $\boldsymbol{x} \in (\boldsymbol{V} \cup \boldsymbol{T})^+$, $\boldsymbol{y} \in (\boldsymbol{V} \cup \boldsymbol{T})^*$

Example: Defining Korean (한글)

A grammar for a single 한글 syllable: $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ with

$$\boldsymbol{V} = \{\boldsymbol{S}, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}\}, \qquad \boldsymbol{T} = \{\text{ㄱ}, \text{ㄲ}, \ldots, \text{ㅎ}, \ \text{ㅏ}, \text{ㅐ}, \ldots, \text{ㅘ}, \ldots, \text{ㅣ}, \ \text{ㄳ}, \text{ㄵ}, \ldots\}$$

and productions $\boldsymbol{P}$:

$\boldsymbol{S} \to \boldsymbol{A}\ \boldsymbol{B}\ \boldsymbol{C}$	(tuple: 초성, 중성, 종성)
$\boldsymbol{A} \to$ ㄱ | ㄲ | ㄴ | $\cdots$ | ㅎ	(쌍자음 포함)
$\boldsymbol{B} \to$ ㅏ | ㅐ | $\cdots$ | ㅘ | $\cdots$ | ㅣ	(복합 모음은 한 단위)
$\boldsymbol{C} \to \boldsymbol{\varepsilon}$ | ㄱ | ㄲ | ㄳ | $\cdots$ | ㅎ	($\boldsymbol{\varepsilon}$ = 받침 없음)

Example derivation: $\boldsymbol{S} \Rightarrow^{*}$ ㄱ ㅏ ㄱ.