Lecture 5 - May 20

Lexical Analysis

Formulating String, Language, Problem
Regular Language Operations:
Union, Concatenation, Kleene Closure

Formulating Strings

e.g. $\Sigma = \{0, 1\}$ → alphabet/symbol

$\Sigma^0 = \{\varepsilon\}$

$\Sigma^1 = \{0, 1\}$ → string of length 1

$\Sigma^2 = \{00, 01, 10, 11\}$ $\quad |\Sigma^2| = 4$

Set of Strings of Length k

$$\Sigma^k = \{w \mid w \text{ is a string over } \Sigma \wedge |w| = k\}$$

↳ $w = c_0 c_1 c_2 \dots c_{k-1} \wedge (\forall i \mid 0 \le i \le k-1 \bullet c_i \in \Sigma)$

↳ $\forall i \bullet 0 \le i \le k-1 \Rightarrow c_i \in \Sigma$

Set of Nonempty Strings

$$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \dots = \{w \mid w \in \Sigma^k \wedge k > 0\}$$

Set of Strings of All Possible Lengths

$$\Sigma^* = \Sigma^+ \cup \{\varepsilon\} = \{w \mid w \in \Sigma^k \wedge k \ge 0\}$$

Logical Quantification

$$\forall i \mid R(i) \bullet P(i)$$

- $\mid$ → such that
- $R(i)$ → range of dummy variable(s) to consider
- $\bullet$ → it's the case
- $P(i)$ → property

↳ alt: $\forall i \bullet R(i) \Rightarrow P(i)$

- $\bullet$ → it's the case
- $\Rightarrow$ → false $\Rightarrow$ ___ $\equiv$ true

$$\exists i \mid R(i) \bullet P(i)$$

- $\bullet$ → and

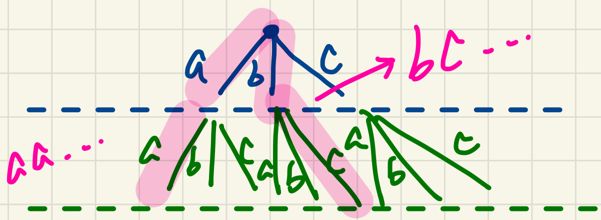
↳ alt: $\exists i \bullet R(i) \wedge P(i)$

- $\wedge$ → true $\wedge$ true $\equiv$ true.

1. $|\{a, b, \ldots, z\}^{5}| = 26^5$

$\underbrace{\_}_{26}\ \underbrace{\_}_{26}\ -\ -\ \underbrace{\_}_{26}$

2. Enumerate $\{a, b, c\}^{4}$ — 3^4 strings in this set



3. Σ (alphabet set) vs. Σ' (set of strings)

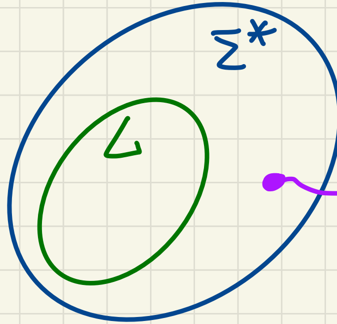
4. Prove or disprove:

$$\Sigma_1 \subseteq \Sigma_2 \Rightarrow \Sigma_1^* \subseteq \Sigma_2^*$$

($\Sigma_1 \subseteq \Sigma_2$: alphabets; $\Sigma_1^* \subseteq \Sigma_2^*$: strings)

Language

$L \subseteq \Sigma^*$



when $L \subset \Sigma^*$, there's at least one string w s.t. $w \in \Sigma^* \land w \notin L$

$\{1,2\} \subseteq \{1,2,3\}$

$\{1,2\} \subset \{1,2,3\}$

$\{1,2\} \not\subset \{1,2\}$

Σ_{key} : all ASCII chars. from keyboard

$\Sigma^*_{key} : \Sigma^0_{key} \cup \Sigma^1_{key} \cup \Sigma^2_{key} \cup \ldots$

$\Sigma^1_{key} \cup \Sigma^2_{key} \cup \ldots = \Sigma^+_{key}$

Σ^*_{key}: any texts that can be typed from keyboard.

$L_{java} = \{ prog \mid prog \in \Sigma^*_{key} \land \text{prog compiles in Eclipse} \}$

2. $\{\varepsilon\}$ vs. $\emptyset$

L_1 ↓ singleton set of an empty string

L_2 ↓ empty set of strings

$|L_1| = 1$

$|L_2| = 0$

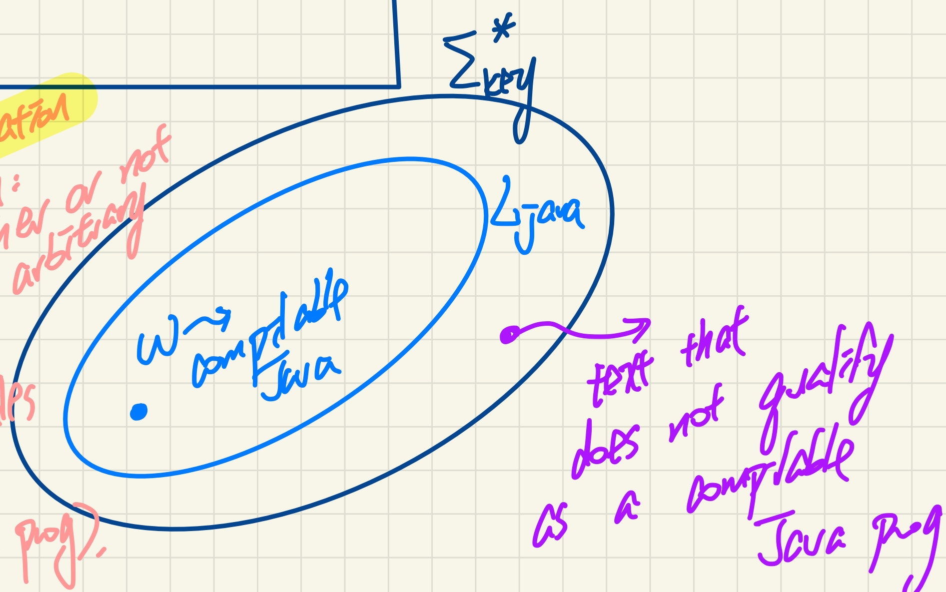
Problems

$w \in L$

$w \rightarrow w \in \Sigma^*$

$L \rightarrow L \subseteq \Sigma^*$

$\in$ → **compilation problem:** whether or not an arbitrary w compiles (e.g. as a Java prog).



Regular Language Operations

set of strings

Recall Σ^0 Σ^1 Σ^2 ...

$L = \{ab, bc, ca\}$

$M = \{ba, cb\}$

1. *Union*

$$L \cup M = \{w \mid w \in L \vee w \in M\}$$

$$L \cup M = \{ab, bc, ca, ba, cb\}$$

2. *Concatenation*

language L concat. with language M

string concat.

$$LM = \{xy \mid x \in L \wedge y \in M\}$$

$x \in L$ $y \in M$

$$|LM| = |L| \times |M|$$

$$LM = \{abba, abcb, bcba, bccb, caba, cacb\}$$

3. *Kleene Closure* (or *Kleene Star*)

$$L^* = \bigcup_{i \geq 0} L^i = L^0 \cup L^1 \cup L^2 \cup \ldots$$

all possible concatenations of L to itself.

Cardinalities?

$L = \{ab, bc\}$

Given some language L

concat. L to itself zero time

$L^0 = \{\varepsilon\}$

$L^1 = \{x \mid x \in L\}$

$L^2 = LL = \{x_1 x_2 \mid x_1 \in L \wedge x_2 \in L\}$

$\vdots$

$L^i = \underbrace{LL \cdots L}_{i \text{ times}} = \{x_1 x_2 \cdots x_i \mid (\forall j \cdot 1 \leq j \leq i \Rightarrow x_j \in L)\}$