Regular Expressions

Sunday, October 25, 2020 9:23 PM

Language Exponentiation

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

$L^{n+1} = LL^n$

L on L^n

concatenation

Kleene Closure

$$L^* = \{ w \in \Sigma^* \mid \exists n \in \mathbb{N}.\ w \in L^n \}$$

$w \in L^*$ iff $\exists n \in \mathbb{N}.\ w \in L^n$

i.e., all strings in successive concatenations of L

Regular Expressions

Defn. Way of describing language via string representation

Strings describing how to build larger language out of smaller pieces

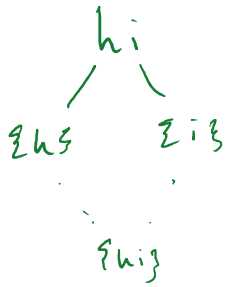
Base

$\emptyset \rightarrow \emptyset$

$a \rightarrow \{a\}$

$\varepsilon \rightarrow \{\varepsilon\}$

Building



Composing — If R_1 and R_2 are regex, R_1R_2 is their concatenation,

$R_1 \cup R_2$ is their union

R^* – Kleene Closure

R^+ – one or more copies of (RR^*)

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

$ab^* = \{a, ab, abb, abbb, \ldots\}$

$a(ab)^* = \{a, aab, aabab, aababab, \ldots\}$

Theorem — If R is a regular expression, then $\mathcal{L}(R)$ is regular.