Lecture 12

February 5, 2024 10:18 PM

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

$\# \cap \sim a$

$L(\# \cap \sim a) = L(\#) \cap L(\sim a)$

$= \Sigma \cap (\Sigma^* - L(a))$

$= \{a, b\} \cap (\Sigma^* - \{a\}) = \{b\}$

---

* $\{ x \in \{a, b, c\}^*,\ x$ does not have any a in it $\}$

* $\alpha = \sim a$ ✗ $aa \in L(\alpha)$
* $\beta = \sim(@ a @)$ ✓
* $\gamma = \sim(a^+)$ ✗ $bac \in L(\gamma)$
* $\delta = (c + b)^*$ ✓

$L(\beta) = L(\delta)$

but $\beta \neq \delta$

Patterns are redundant. We actually don't need all of the atomic patterns or compound patterns.

* We can replace $\#$ with $a + b + c$ (for $\Sigma = \{a, b, c\}$)
* ε with $\emptyset^*$
* $\alpha \cap \beta$ with $\sim(\sim\alpha + \sim\beta)$
* $\sim\alpha$ with ..? (more difficult)

* $\sim\alpha$ with ... (cut off)

* $\alpha = \sim(@\, b @ b @)$ where $\Sigma = \{a, b\}$

* find "equivalent" β without $\sim$?

$a^* b a^* + a^*$

Regular Expressions

A regular expression is a pattern that only uses the following:

* Atomic: $\emptyset$, a, ε ($a \in \Sigma$)
* Compound: $+$, $\cdot$, $*$ ($\cdot$ concat)

* The order of precedence for compound operations: $* > \cdot > +$

$ac^* b + a^* b^* \equiv (a(c^*)b) + ((a^*)(b^*))$

* Regular expression define equivalency classes. $\alpha \equiv \beta$ iff $L(\alpha) = L(\beta)$

* Reflexive $\alpha \equiv \alpha$
* Symmetric: $\alpha \equiv \beta \iff \beta \equiv \alpha$
* Transitive: $\left.\begin{matrix}\alpha \equiv \beta \\ \beta \equiv \gamma\end{matrix}\right\} \Longrightarrow \alpha \equiv \gamma$

* $\alpha \leq \beta$ if $L(\alpha) \subseteq L(\beta)$

partial order

Theorem. The following statements are equivalent for every set $A \subseteq \Sigma^*$:

(i) A is regular (has a DFA)

(ii) $\exists$ NFA N, s.t. $L(N) = A$

(iii) " " N with ε-transitions s.t. $L(N) = A$

(iv) $\exists$ pattern α s.t. $L(\alpha) = A$

(v) $\exists$ regular expression β s.t. $L(\beta) = A$