

MATH 2301

* Welcome back! Announcements:

- PDF notes updated
- Exam results on Gradescope.

* Before the break: posets & Möbius inversion.
(See Worksheet 6)

* Today: Regular expressions & finite automata.

- An alphabet Σ = a finite set of symbols or "letters"
Usually, we'll take $\Sigma = \{0, 1\}$

- Strings or words: A string/word on Σ is a finite ordered list of letters, written
 $w = a_1 a_2 a_3 \dots a_k$, each $a_i \in \Sigma$, or w can be empty. We write ε for the empty string.
(We assume that ε is not a symbol in Σ .)
↑ epsilon

- A language on Σ is a set of words on Σ

We say $\Sigma^* :=$ set of all words on Σ

In other words, any subset $L \subseteq \Sigma^*$ is called a language.

Note: Σ^* is typically infinite

If $\Sigma = \phi$, then $\Sigma^* = \{\varepsilon\}$. is the only situation in which Σ^* is finite.

* Examples

$$\Sigma = \{0, 1\}$$

Some strings: 10, 000, ε , 110, 1111, etc.

Some languages: ϕ , Σ^* , $\{0\}$, $\{\varepsilon\}$, $\{1\}$, $\{2, 00\}$, $\{\varepsilon, 011, 1100, 111\}$.

$L =$ set of strings without 0s

$L =$ set of strings that begin with a zero

...

* a language need not fit neatly into any rule.

But, we'll use regular expressions to identify languages that do follow some patterns.

* Operations on strings/languages

Fix a Σ .

- Concatenation (strings) If v and w are strings,
 $v = a_1 a_2 \dots a_k$ & $w = b_1 b_2 \dots b_\ell$ with $a_i, b_j \in \Sigma$
then $vw = \text{concatenation} = a_1 a_2 \dots a_k b_1 \dots b_\ell$.

Note: $\varepsilon w = w \varepsilon = w$ for any w .

- Concatenation (languages)

Let $L_1, L_2 \subseteq \Sigma^*$ be languages.

Then their concatenation is

$$L_1 \circ L_2 = \{vw \mid v \in L_1, \text{ and } w \in L_2\}$$

- Union (languages)

Let $L_1, L_2 \subseteq \Sigma^*$. Their union is

$$L_1 \cup L_2 = \{v \mid v \in L_1 \text{ or } v \in L_2\}$$

= set union of L_1 & L_2 .

- Star (of a language)

Let $L \subseteq \Sigma^*$. Then

the star of $L = L^* = \text{any number of concatenations of (possibly different) elements of } L$

$$L^* = \{w_1 w_2 \dots w_k \mid w_i \in L \text{ for each } i\} \cup \{\varepsilon\}$$

* Note: the star of Σ is just Σ^*

* Examples

$$\Sigma = \{0, 1\}$$

$$L_1 = \emptyset, L_2 = \{\varepsilon, 0\}, L_3 = \{1, 11\}$$

$$- L_1 \circ L_2 = \emptyset$$

$$- L_2 \circ L_3 = \{1, 11, 01, 011\}$$

$$- L_3 \circ L_2 = \{1, 11, 10, 110\}$$

concatenation is directional (not commutative)

$$- L_1 \cup L_2 = \{\varepsilon, 0\}$$

$$- L_2 \cup L_3 = \{\varepsilon, 0, 1, 11\}$$

union is commutative

$$= L_3 \cup L_2$$

$$- L_1^* = \{\varepsilon\}$$

$$- L_2^* = \{\varepsilon, 0, 00, 000, \dots\}$$

↑ strings that don't contain any 1s.

Note:

$$\varepsilon 0 \varepsilon 00 = 000$$

$$- L_3^* = \{\varepsilon, 1, 11, 111, \dots\}$$

↑ strings that don't contain 0s.

Note: L^* is infinite unless:

$$- L = \emptyset \leadsto L^* = \{\varepsilon\}$$

$$- L = \{\varepsilon\} \leadsto L^* = \{\varepsilon\}$$

* Lexicographic (dictionary) order.

Fix Σ and a total order on Σ .

Now we can order the elements of Σ^* , as follows.

Let $v, w \in \Sigma^*$

1) If $\text{length}(v) \neq \text{length}(w)$, then the shorter one comes first

2) If $\text{length}(v) = \text{length}(w)$, then compare letter by letter.

If $v = a_1 \dots a_n$

$w = b_1 \dots b_n$.

If $v \neq w$, then find the first i where $a_i \neq b_i$

If $a_i < b_i$ then $v < w$, otherwise $w < v$.

Note: We use the same system of ordering on any language $L \subseteq \Sigma^*$.

* Regular expression syntax.

Informal def.: A regular expression is a pattern that corresponds to zero or more words in Σ^* , specified according to certain rules