Standard sets of strings

Definition of A^n, A^+ and A^*

Let A be an alphabet

- A^n = the set of all strings over A with length n
- A^+ = the set of all strings over A with length greater than 0
- A^* = the set of all strings over A
- $A^0 = \{\epsilon\}$
- $A^+ = \bigcup_{n>0} A^n$
- $A^* = \bigcup_{n\geq 0} A^n = A^0 \cup A^+$

Formal language: syntactic notion of language

Definition

A language L over an alphabet A is a subset of A^*

Example

The set L_{id} of all identifiers

- $A = \{\texttt{'a'}, \ldots, \texttt{'z'}\} \cup \{\texttt{'A'}, \ldots, \texttt{'Z'}\} \cup \{\texttt{'0'}, \ldots, \texttt{'9'}\}$
- $L_{id} = \{\ \texttt{"a"}, \texttt{"b"}, \ldots, \texttt{"a0"}, \texttt{"a1"}, \ldots\}$

Problem

Is it possible to define L in a finite way?
Solution: define L as the composition of simpler languages

Composition operators between languages

- concatenation: $L_1 \cdot L_2 = \{u \cdot w \mid u \in L_1, w \in L_2\}$
- union: $L_1 \cup L_2$

Intuition

Union

$L = L_1 \cup L_2$: any string of L is either a string of L_1 or a string of L_2

Example:

$L' = \{ \texttt{"a"}, \ldots, \texttt{"z"} \} \cup \{ \texttt{"A"}, \ldots, \texttt{"Z"} \}$

Concatenation

$L = L_1 \cdot L_2$: any string of L is a string of L_1 followed by a string of L_2

Examples:

- $\{\texttt{"a"},\texttt{"ab"}\} \cdot \{\epsilon,\texttt{"1"}\} = \{\texttt{"a"},\texttt{"ab"},\texttt{"a1"},\texttt{"ab1"}\}$
- $L_{id} = L' \cdot A^*$ with A = $\{\texttt{'a'},\ldots,\texttt{'z'}\} \cup \{\texttt{'A'},\ldots,\texttt{'Z'}\} \cup \{\texttt{'0'},\ldots,\texttt{'9'}\}$

More on languages

Monoids and languages

- concatenation is *associative*, but not *commutative*
- $A^0 (= \{\epsilon\})$ is the identity element

Iteration of concatenation

L^n defined by induction on n (natural number):

- $L^0 = A^0 (= \{\epsilon\})$
- $L^{n+1} = L \cdot L^n$

Intuition: L^n is L concatenated with itself n times

$+$ and $*$ operators

- $L^+ = \bigcup_{n>0} L^n$
- $L^* = \bigcup_{n\geq 0} L^n$ ($*$ is called the Kleene star)
- equivalently, $L^* = L^0 \cup L^+$

Intuition

L^+

Any string of L^+ is obtained by concatenating one of more strings of L

L^*

Any string of L^* is obtained by concatenating zero of more strings of L
Example: $\{\texttt{"0"},\texttt{"1"}\}^* = \{\epsilon,\texttt{"0"},\texttt{"1"},\texttt{"00"},\texttt{"01"},\texttt{"10"},\texttt{"11"},\ldots\}$
Remark 1: concatenating zero strings means the empty string
Remark 2: $L^+ = L \cdot L^*$

Regular expressions

A commonly used formalism for defining simple languages (=syntax)

What are regular expressions?

Inductive definition of regular expressions over an alphabet A

- base cases:
  - $\emptyset$ is a regular expression over A
  - ϵ is a regular expression over A
  - for all $\sigma \in A$, σ is a regular expression over A
- inductive cases:
  - if e_1 and e_2 are regular expression over A,
    then $e_1|e_2$ is a regular expression over A
  - if e_1 and e_2 are regular expression over A,
    then $e_1\ e_2$ is a regular expression over A
  - if e is a regular expression over A, then e^* is a regular expression over A
  - if e is a regular expression over A, then (e) is a regular expression over A