1 Alphabets and Languages

Look at handout 1 (inference rules for sets) and use the rules on some examples like

$\{a\} \subseteq \{\{a\}\}$
$\{a\} \in \{a, b\}$,
$\{a\} \in \{\{a\}\}$,
$\{a\} \subseteq \{\{a\}\}$,
$\{a\} \subseteq \{a, b\}$,
$a \subseteq \{\{a\}\}$,
$a \in \{a, b\}$,
$a \in \{\{a\}\}$,
$a \subseteq \{a, b\}$

Example: To show $\{a\} \subseteq \{a, b\}$, use inference rule L1 (first one on the left). This asks us to show $a \in \{a, b\}$. To show this, use rule L5, which succeeds.

To show $\{a\} \in \{a, b\}$, which rule applies?

- The only one is rule L4. So now we have to either show $\{a\} = a$ or $\{a\} = b$. Neither one works.
- To show $\{a\} = a$ we have to show $\{a\} \subseteq a$ and $a \subseteq \{a\}$ by rule L8. The only rules that might work to show $a \subseteq \{a\}$ are L1, L2, and L3 but none of them match, so we fail.
- There is another rule for this at the very end, but it also fails.
- To show $\{a\} = b$, we try the rules in a similar way, but they also fail.

Therefore we cannot show that $\{a\} \in \{a, b\}$. This suggests that the statement $\{a\} \in \{a, b\}$ is false.

Suppose we have two set expressions only involving brackets, commas, the empty set, and variables, like $\{a, \{b, c\}\}$ and $\{a, \{c, b\}\}$. Then there is an easy way to test if they are equal. If they can be made the same by

- permuting elements of a set, and
- deleting duplicate items of a set

then they are equal, otherwise they are not equal.

- So $\{a, \{b, c\}\}$ and $\{a, \{c, b\}\}$ are equal and $\{a, a, b\}$ and $\{b, a\}$ are equal, but $\{a, b\}$ and $\{c, b\}$ are not.
- These rules show, for example, that $\{a\} = a$ and $\{a\} = b$ are both false, which shows quickly that $\{a\} \in \{a, b\}$ is false.

These rules are not needed for simple examples, but they can help when there are many brackets to consider.

1.1 Alphabets

An *alphabet* is a finite set of *symbols*.

Alphabets are denoted by Σ.

1.2 Strings

- A *string* over an alphabet is a finite sequence of symbols from the alphabet.

  Example: If Σ is $\{0, 1\}$ then 01101 is a string over Σ.
- The *empty string* is ϵ. The book uses e for this.
- If Σ is an alphabet then Σ^* is the set of strings over Σ.

  For example, if Σ is $\{0, 1\}$ then Σ^* is the set of binary sequences.
- The *length* of a string is the number of symbol occurrences in it.

  The length of 01101 is 5.
- The *concatenation* of two strings x and y is denoted by xy or $x \circ y$.

  Note that $x \circ \epsilon = \epsilon \circ x = x$.
- v is a *substring* of w if there are strings x and y such that $w = xvy$.

  Thus bc is a substring of $abcab$.

  Question: How many substrings are there in a string of length n?
- A *suffix* of a string is a substring that ends at the end of the string.

- A *prefix* of a string is a substring that begins at the beginning of the string.

  Thus bc is a suffix of abc and ab is a prefix of abc.

  How many suffixes are there of a string of length n?

- w^i is the string w repeated i times.

  Thus $ababab = (ab)^3$.

- w^R is the string w with the letters in reverse order.

  Thus $(ab)^R = ba$.

1.3 Languages

A *language* over an alphabet Σ is a subset of Σ^*, that is, it is a set of strings over Σ.

Thus $\{0, 1, 00, 11\}$ is a language over $\{0, 1\}$.

The set of odd length binary strings is also a language over $\{0, 1\}$.

The set of all binary strings (that is, $\{0, 1\}^*$) is also a language over $\{0, 1\}$.

1.4 Operations on Languages

Now we will study *operations* on languages. These take one or two languages and produce another language from them.

1.4.1 Complement

If A is a language over Σ then $\overline{A}$, the *complement* of A, is $\Sigma^* - A$.

Example: If Σ^* is $\{0, 1\}^*$, what is the complement of $\{00\}$?

1.4.2 Concatenation

If L_1 and L_2 are languages then $L_1 \circ L_2$, or L_1L_2, the *concatenation* of L_1 and L_2, is

$$\{xy : x \in L_1, y \in L_2\}$$

.

- Thus if L_1 is $\{a, b\}$ and L_2 is $\{c, d\}$ then $L_1 \circ L_2$ is $\{ac, ad, bc, bd\}$.

- If L_1 is $\{\epsilon, a, aa, aaa, \ldots\}$ and L_2 is $\{\epsilon, b, bb, bbb, \ldots\}$ then what is $L_1 \circ L_2$?
- Note that for any L, $\{\epsilon\} \circ L = L \circ \{\epsilon\} = L$.

To get $L_1 \circ L_2$, write L_1 as a sequence above L_2, then selecting one element from each and concatenating them gives an element of $L_1 \circ L_2$:

$$L_1 = \{x_1, x_2, x_3, \ldots\}$$
$$L_2 = \{w_1, w_2, w_3, \ldots\}$$

1.4.3 Kleene Star

If L is a language then L^*, the *Kleene star* of L, is

$$\{w_1 w_2 \ldots w_k : k \geq 0, w_i \in L \text{ all } i\}$$

.

This can also be written as

$$\{\epsilon\} \cup L \cup (L \circ L) \cup (L \circ L \circ L) \cup \ldots$$

or as

$$\{\epsilon\} \cup L \cup L^2 \cup L^3 \cup \ldots.$$

To get the Kleene star, write a language above itself infinitely many times, then select one element from some finite number of the lists (possibly zero) and concatenate them:

$$L = \{x_1, x_2, x_3, \ldots\}$$
$$L = \{x_1, x_2, x_3, \ldots\}$$
$$L = \{x_1, x_2, x_3, \ldots\}$$
$$L = \{x_1, x_2, x_3, \ldots\}$$
$$\ldots$$

This gives an element of L^*.
Examples: If L is $\{00, 01, 10, 11\}$, what is L^*?
If L is $\{00\}$, what is L^*?
If L is $\{00, 11\}$, what are some strings in L^*?
If L is $\{\}$, then L^* is $\{\epsilon\}$.
If L is $\{\epsilon\}$, then L^* is also $\{\epsilon\}$.

1.4.4 Plus

Another operation on languages:

$$L^+ = L \circ L^* = \{w_1 w_2 \dots w_k : k > 0, w_i \in L \text{ all } i\}.$$

To get L^+, write a language above itself infinitely many times, then select one element from some finite number of the lists (at least one) and concatenate them:

$$L = \{x_1, x_2, x_3, \dots\}$$

$$L = \{x_1, x_2, x_3, \dots\}$$
$$L = \{x_1, x_2, x_3, \dots\}$$
$$L = \{x_1, x_2, x_3, \dots\}$$
$$\dots$$

This gives an element of L^+.

What is L^+ for the three languages L given above?

If L is $\{e, 00, 01, 10, 11\}$, what is L^*? What is L^+?

1.4.5 Union

Of course, if L_1 and L_2 are languages, then $L_1 \cup L_2$ is another language, so that union is another operation on languages.

Note that operations can be nested, so that if A and B are languages, we can talk about $(A \circ B) \cup A^*$, for example. Thus arbitrary expressions can be made from languages using these operations repeatedly.

Some identities:

$$L^* = \{\epsilon\} \cup L \cup (L \circ L) \cup (L \circ L \circ L) \cup \dots$$

$$L^* = \{\epsilon\} \cup (L \circ L^*)$$

$$L^+ = L \circ L^*$$

$$L^+ = L \cup (L \circ L) \cup (L \circ L \circ L) \cup \ldots$$

$$L^+ = L \cup (L \circ (L^+))$$

$$L^* = \{\epsilon\} \cup L^+$$

See Handout 2 (Rules of Inference for Operations on Languagse).
Problem 1.7.4 (c) page 46: Show

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