CHAPTER 2

ORDERED SETS

In this chapter we study binary relations which will be used extensively in the remainder of the book.

Ordered sets play a fundamental role, similar to the role of metric spaces, because they allow the comparison of two or more objects. Ordered sets will be used in most of the subsequent chapters (Chapter 3, Chapter 4, Chapter 9, etc.).

We define order relations and ordered sets, mappings between ordered sets and special elements such as minimal and maximal elements, upper and lower bounds. The concept of ideals comes from the general framework in which we can use proofs by induction. Finally, we study complete sets and monotone functions on complete sets, which form the basis of the semantics of programming languages.

The following book is very complete and at the same time quite readable: Garrett Birkhoff, *Lattice theory*, AMS, 3rd edition, Rhode Island (1979).

2.1 Order and preorder relations

2.1.1 Orders and strict orders

Definition 2.1  An order relation or ordering is a reflexive, antisymmetric and transitive relation. A strict ordering is an irreflexive and transitive relation.

Remark 2.2
1. If $\mathcal{R}$ is a strict ordering on a set $E$, then the relation $\mathcal{R} \cup \text{Id}_E$ is an ordering on $E$. Conversely, if $\mathcal{R}'$ is an ordering, $\mathcal{R}' \setminus \text{Id}_E$ is a strict ordering.
2. If $\mathcal{R}$ is an antisymmetric and transitive relation, then the relation $\mathcal{R} \cup \text{Id}_E$ is an ordering and the relation $\mathcal{R} \setminus \text{Id}_E$ is a strict ordering.

Orderings are usually denoted by $\leq$, and strict orderings are denoted by $<$. By the preceding remark, it is easy to derive an ordering from the corresponding strict ordering and the converse, i.e.
2. Ordered sets

- \( z \leq y \) is equivalent to \( z < y \) or \( z = y \),
- \( z < y \) is equivalent to \( z \leq y \) and \( z \neq y \).

2.1.2 Total orderings and partial orderings

If ordering \( \mathcal{R} \) verifies for all \( e, e' \in E \), \( e \neq e' \implies (e \mathcal{R} e' \text{ or } e' \mathcal{R} e) \), then \( \mathcal{R} \) is called a total ordering. Otherwise \( \mathcal{R} \) is called a partial ordering.

**Example 2.3**

(i) The usual ordering on real numbers is a total ordering.
(ii) The divisibility relation on integers is a partial ordering (\( a \leq \text{div} \ b \) if and only if there exists \( c \) such that \( b = ac \)).
(iii) Inclusion on \( \mathcal{P}(E) \) is a partial ordering if \( |E| > 1 \) and a total ordering if \( |E| \leq 1 \).

In the next example, we define common orderings on the free monoid \( A^* \) (see Definition 1.15).

**Example 2.4**

(i) The prefix ordering is a partial ordering on the monoid \( A^* \) that is defined as follows. String \( u = u_1 \ldots u_n \) is a prefix of string \( v = v_1 \ldots v_m \) if \( n \leq m \) and \( \forall i \leq n, u_i = v_i \).
(ii) We assume that alphabet \( A \) has a total ordering \( \leq \). The lexicographic ordering \( \preceq \) is a total ordering on the monoid \( A^* \) that is defined as follows. Let \( u = u_1 \ldots u_n \) and \( v = v_1 \ldots v_m \) be two strings. \( u \preceq v \) if
   - either \( u \) is a prefix of \( v \),
   - or \( u \) and \( v \) coincide up to letter \( k \), \( u_{k+1} \neq v_{k+1} \) and \( u_{k+1} \leq v_{k+1} \), with \( 0 \leq k < \inf(n, m) \).

2.1.3 Preorders

**Definition 2.5** A preorder relation is a transitive relation.

**Example 2.6** Let \( E = \mathcal{P}(\mathbb{N}) \) be the set of finite subsets of \( \mathbb{N} \). Each subset \( X \) contains a least element, denoted by \( \inf(X) \), and a greatest element, denoted by \( \sup(X) \). We define the relation \( \mathcal{R} \) on \( E \) by: \( X \mathcal{R} X' \) if and only if \( \inf(X) \leq \inf(X') \) and \( \sup(X) \leq \sup(X') \). It is easy to see that this relation is transitive and reflexive. But it is not antisymmetric because \( (X \mathcal{R} X' \text{ and } X' \mathcal{R} X) \) implies \( \inf(X) = \inf(X') \) and \( \sup(X) = \sup(X') \), but not necessarily \( X = X' \).
Proposition 2.7 If $\mathcal{R}$ is a preorder relation on $E$ then $Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1})$ is an equivalence relation.

Note that the relation $\equiv_\mathcal{R} = Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1})$ can be translated by $e \equiv_\mathcal{R} e'$ if and only if either $e = e'$ or $(e \mathcal{R} e'$ and $e' \mathcal{R} e$).

Proof. The relation $Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1})$ is obviously reflexive and symmetric. We show that it is transitive, i.e. that

$$(Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1}))^2 \subseteq Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1}).$$

We see that $(Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1}))^2$ is equal to $Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1}) \cup (\mathcal{R} \cap \mathcal{R}^{-1})^2$. Because $\mathcal{R} \cap \mathcal{R}^2$ is the intersection of two transitive relations, it is also transitive (see Exercise 3.5) and thus $(\mathcal{R} \cap \mathcal{R}^{-1})^2 \subseteq (\mathcal{R} \cap \mathcal{R}^{-1})$. \qed

Example 2.8 If we consider the preorder $\mathcal{R}$ of Example 2.6, the equivalence relation defined in Proposition 2.7 is

$$X \ (Id_E \cup (\mathcal{R} \cap \mathcal{R}^{-1})) \ X'$$

if and only if $\inf(X) = \inf(X')$ and $\sup(X) = \sup(X')$.

Let $\mathcal{R}$ be a preorder relation on $E$ and let $\mathcal{E}$ be the associated equivalence. On the factor set $E \mathcal{E}$ of $E$ by $\mathcal{E}$, we can define the relation $\mathcal{R}'$ by $[e]_\mathcal{E} \mathcal{R}' [e']_\mathcal{E}$ if and only if $e \mathcal{R} e'$. This definition does not depend on the choice of the elements $e$ and $e'$ within their equivalence class, because if $e \in \mathcal{E} e_1$ and $e' \in \mathcal{E} e'_1$, then $e \mathcal{R} e'$ if and only if $e_1 \mathcal{R} e'_1$.

Exercise 2.1 Prove that $\mathcal{R}'$ does not depend on the choice of the elements $e$ and $e'$ within their equivalence class. \diamondsuit

Proposition 2.9 The relation $\mathcal{R}'$ is antisymmetric and transitive; $\mathcal{R}'$ is an ordering if $\mathcal{R}$ is reflexive and $\mathcal{R}'$ is a strict ordering if $\mathcal{R}$ is irreflexive.

Proof. $\mathcal{R}'$ is transitive, because $[e]_\mathcal{E} \mathcal{R}' [e']_\mathcal{E}$ and $[e']_\mathcal{E} \mathcal{R}' [e'']_\mathcal{E}$ imply $e \mathcal{R} e'$ and $e' \mathcal{R} e''$, and hence $e \mathcal{R} e''$. It is antisymmetric because $([e]_\mathcal{E} \mathcal{R}' [e']_\mathcal{E}$ and $[e']_\mathcal{E} \mathcal{R}' [e]_\mathcal{E}$) implies $(e \mathcal{R} e'$ and $e' \mathcal{R} e)$. Thus $(e, e') \in (\mathcal{R} \cap \mathcal{R}^{-1}) \subseteq \mathcal{E}$, and hence $[e]_\mathcal{E} = [e']_\mathcal{E}$.

$\mathcal{R}'$ is reflexive (resp. irreflexive) if $\mathcal{R}$ is reflexive (resp. irreflexive). \qed

This ordering $\mathcal{R}'$ will be called the factor ordering of the preorder $\mathcal{R}$.

Example 2.10 Again with the preorder of Example 2.6, the factor set of $E = \mathcal{P}_f(\mathbb{N})$ can be identified with the set of pairs $(a, b)$ of integers such that $a \leq b$. On this set the ordering $\mathcal{R}'$ is defined by $(a, b) \mathcal{R}' (a', b')$ if and only if $a \leq a'$ and $b \leq b'$.

Exercise 2.2 Let $\mathcal{R}$ be a preorder relation. Show that the relation $\mathcal{R}^\dagger$ defined by $x \mathcal{R}^\dagger y$ if and only if $x = y$ or $(x \mathcal{R} y$ and $y \mathcal{R} z)$, where $\mathcal{R}$ denotes the complementary of relation $\mathcal{R}$ (see Section 1.4.3), is an ordering. \diamondsuit
2.2 Ordered sets

Definition 2.11  An ordered set \((E, \leq)\) is a set \(E\) together with an ordering \(\leq\).

The same set \(E\) can be equipped with different orderings. We then have different ordered sets.

Example 2.12  The set of integers \(\mathbb{N}\) can be equipped with the usual ordering or with the divisibility ordering of Example 2.3.

2.2.1 Monotonic mappings

Definition 2.13  Let \((E_1, \leq_1)\) and \((E_2, \leq_2)\) be two ordered sets. A mapping \(f\) from \(E_1\) to \(E_2\) is said to be monotonic, or monotone, if

\[ \forall x, y \in E_1, \quad x \leq_1 y \implies f(x) \leq_2 f(y). \]

\(f\) is also said to be a homomorphism from the ordered set \((E_1, \leq_1)\) to the ordered set \((E_2, \leq_2)\).

\((E_1, \leq_1)\) and \((E_2, \leq_2)\) are said to be isomorphic if there is a bijection \(b\) between \(E_1\) and \(E_2\) with the property that both \(b\) and \(b^{-1}\) are monotone.

Example 2.14  
1. If two ordered sets \((E_1, \leq_1)\) and \((E_2, \leq_2)\) have the same underlying set, namely, if \(E_1 = E_2\), then the inclusion \(\leq_1 \subseteq \leq_2\), i.e. \(\forall x, y, \quad x \leq_1 y \implies x \leq_2 y\), holds if and only if the identity mapping from \(E_1\) to \(E_2\) is monotone.

2. In order for a bijection to be an isomorphism, monotonicity is not sufficient; for instance, the identity mapping from \((\mathbb{N}, \leq_{\text{div}})\) to \((\mathbb{N}, \leq)\) is a monotone bijection but it is not an isomorphism.

2.2.2 Totally ordered sets

An ordered set \((E, \leq)\) is said to be totally ordered if \(\leq\) is a total ordering, i.e. if \(\forall x, y, \quad x \neq y \implies x \leq y \text{ or } y \leq x\). Otherwise, i.e. if \(\exists x, y, \quad x \neq y \text{ and } y \nleq x\), it is said to be a partially ordered set or poset. Let \((E, \leq)\) be a partially ordered set. A linear extension of \((E, \leq)\) is a totally ordered set \((E, \leq_i)\) with the same underlying set such that \(\leq \subseteq \leq_i\).

Theorem 2.15  Let \((E, \leq)\) be an ordered set. It has at least one linear extension, and \(\leq\) is equal to the intersection of all its linear extensions.

This theorem will not be proved in the general case.

Exercise 2.3  Prove the statement of Theorem 2.15 for the case when \(E\) is finite.
2.2.3 Products of ordered sets

Let \((B_1, \leq_1)\) and \((B_2, \leq_2)\) be two ordered sets. The direct product of these two ordered sets is \((B_1 \times B_2, \leq)\) with the ordering \(\leq\) defined by \((x_1, x_2) \leq (y_1, y_2)\) if and only if \(x_1 \leq_1 y_1\) and \(x_2 \leq_2 y_2\).

**Remark 2.16**
1. The ordering on the direct product is also called the product ordering.
2. We can define orderings on \(B_1 \times B_2\) other than the product ordering; for instance, we can define \((x_1, x_2) \leq^I (y_1, y_2)\) if and only if \(y_1 \leq_1 x_1\) and \(x_2 \leq_2 y_2\).

**Exercise 2.4**
1. Show that the projections \(\pi_i\) from \(B_1 \times B_2\) onto \(B_i\) are monotonic.
2. Show that if \(|B_1| \geq 2\) and \(|B_2| \geq 2\), \(B_1 \times B_2\) is not totally ordered even when \(B_1\) and \(B_2\) are.
3. Show that the direct product is associative and commutative up to isomorphism (i.e. the mapping \(b\) from \(B_1 \times B_2\) to \(B_2 \times B_1\) associating \((x_1, x_2)\) with \((x_2, x_1)\) is an isomorphism).

The lexicographic product of \((B_1, \leq_1)\) by \((B_2, \leq_2)\) is \((B_1 \times B_2, \leq)\) with \((x_1, x_2) \leq (y_1, y_2)\) if and only if \(x_1 < y_1\) or \((x_1 = y_1\) and \(x_2 \leq_2 y_2\).

**Exercise 2.5**
1. Show that the lexicographic product of two ordered sets is an ordered set.
2. Show that this product is not commutative, i.e. \((B_1 \times B_2, \leq)\) is not isomorphic to \((B_2 \times B_1, \leq)\).
3. Show that the lexicographic product of total orderings is a total ordering.

2.2.4 Ordered subsets, chains and antichains

Let \((B, \leq)\) be an ordered set. A subordered set of \((B, \leq)\) is an ordered set \((E', \leq')\) such that \(E' \subseteq E\) and \(\leq' = \cap (E' \times E')\), i.e. \(\forall x, y \in E', x \leq' y\) if and only if \(x \leq y\).

A chain of \(E\) is a totally ordered subset of \(E\). A chain is maximal if it is not strictly included in another chain.

An antichain \(E'\) of \(E\) is a subset of \(E\) such that

\[ \leq \cap (E' \times E') = Id_{E'} \, . \]

In other words, any two elements of an antichain are incomparable, because if they are in the ordering then they must be equal. An antichain is maximal if it is not strictly included in any other antichain.

**Exercise 2.6**
1. If \((B, \leq)\) is a totally ordered set then its only antichains are singletons.
2. Show that the intersection of a chain and an antichain has at most one element.
A left segment is a subset $E'$ of $E$ such that

$$y \in E' \text{ and } x \leq y \implies x \in E'.$$

An interval $[x, y]$, with $x \neq y$ and $x \leq y$, is the subset

$$\{ z \mid x \leq z \text{ and } z \leq y \}.$$

An ordered set is locally finite if all its intervals are finite.

**Example 2.17** For the usual ordering on numbers, $\mathbb{N}$ is locally finite and $\mathbb{Q}$ is not.

**Exercise 2.7** Show that the interval $[x, y]$ is empty if and only if $x \not\leq y$.

We say that $x$ is covered by $y$ if interval $[x, y]$ contains only $x$ and $y$. This relation will be denoted by $x \prec y$.

**Exercise 2.8** Show that if interval $[x, y]$ is finite then there exists an element of this interval covering $x$.

**Proposition 2.18** If $(E, \leq)$ is locally finite then $\leq = \prec^*$.

**Exercise 2.9** Prove Proposition 2.18.

### 2.3 Upper and lower bounds

**Definition 2.19** Let $E'$ be a subset of an ordered set $(E, \leq)$. An element $z$ of $E$ is an upper bound of $E'$ (resp. lower bound) if $\forall y \in E', y \leq x$ (resp. $x \leq y$).

We denote by $\text{Maj}(E')$ the set of upper bounds of $E'$ and by $\text{Min}(E')$ the set of lower bounds of $E'$. It is easy to see that $\text{Maj}(\emptyset) = \text{Min}(\emptyset) = E$.

**Proposition 2.20** $\text{Maj}(E') \cap E'$ and $\text{Min}(E') \cap E'$ each have at most one element.

**Proof** Assume that $\text{Maj}(E') \cap E'$ contains two distinct elements $x$ and $y$. We thus have $x \leq y$ and $y \leq x$, a contradiction.

The proof is similar for $\text{Min}$. $\square$

If $\text{Maj}(E') \cap E'$ is non-empty then the unique element of this set is called the greatest element or maximum of $E'$. Similarly, if $\text{Min}(E') \cap E'$ is non-empty then its unique element is called the least element or minimum of $E'$. 
Proposition 2.21 Let \( E' \) be a subset of \( E \) and let \( z \in E \). The following three conditions are equivalent:

(i) \( z \) is the greatest element of \( E' \).
(ii) \( x \in E' \) and \( \forall x \in E', x \leq z \).
(iii) \( z \in E' \) and \( z \) is the least element of \( \text{Maj}(E') \).

The least element of \( E' \) has a similar characterization.

Proof.
(i) \( \Rightarrow \) (ii): If \( z \) is the greatest element of \( E' \), then \( z \in E' \) and \( z \in \text{Maj}(E') \), and thus (ii) is true.
(ii) \( \Rightarrow \) (iii): \( \forall x \in E', \forall y \in \text{Maj}(E') \), \( x \leq y \), and thus \( E' \subseteq \text{Min} \left( \text{Maj}(E') \right) \).
Hence \( z \in E' \cap \text{Maj}(E') \subseteq \text{Min} \left( \text{Maj}(E') \right) \cap \text{Maj}(E') \) and \( z \) is the least element of \( \text{Maj}(E') \).
(iii) \( \Rightarrow \) (i): The least element of \( \text{Maj}(E') \) is in \( \text{Maj}(E') \), and thus \( z \in E' \cap \text{Maj}(E') \).

Let \( E' \) be a subset of \( E \). An element \( z \) of \( E' \) is said to be maximal in \( E' \) if \( \forall y \in E' \), \( y \geq z \) \( \Rightarrow \) \( y = z \) or, equivalently, \( y \neq z \) \( \Rightarrow \) \( y \notin z \). If \( E' \) has a greatest element, this greatest element is its unique maximal element, but the converse is false (see Exercise 2.10).

We define the minimal elements of a subset \( E' \) similarly.

Example 2.22 \( \mathbb{N} \) has a minimal element which is its least element (it is 0), but it has no maximal element.

Exercise 2.10
1. Show that if a subset \( E' \) of \( E \) has a unique maximal element, this element is not necessarily the greatest element of \( E' \).
2. What can you say if \( E \) is totally ordered? 

Definition 2.23 An element \( z \) is the least upper bound of a subset \( E' \) of an ordered set \( E \) if

\[
(\forall y \in E', y \leq x) \quad \text{and} \quad (\forall z \in E, (\forall y \in E', y \leq z) \quad \Rightarrow \quad x \leq z)).
\]

Similarly, an element \( z \) is the greatest lower bound of a subset \( E' \) of an ordered set \( E \) if

\[
(\forall y \in E', z \leq y) \quad \text{and} \quad (\forall z \in E, (\forall y \in E', z \leq y) \quad \Rightarrow \quad z \leq x)).
\]

The terminology 'the' least upper bound (resp. greatest lower bound) is justified because there is at most one least upper bound (resp. greatest lower bound).
Indeed, the definition of the least upper bound (resp. greatest lower bound) of a subset \( E' \) of \( E \) is identical to the definition of the least element of \( \text{Maj}(E') \) (resp. the greatest element of \( \text{Min}(E') \)). The least upper bound of subset \( E' \) is thus an upper bound of \( E' \) that is less than all other upper bounds of \( E' \), i.e. the least upper bound of \( E' \) is the least among the upper bounds of \( E' \). Similarly, the greatest lower bound of \( E' \) is the greatest among the lower bounds of \( E' \).

We denote by \( \sup(E') \) and \( \inf(E') \) the least upper bound and greatest lower bound, respectively, when they exist.

**Proposition 2.24** Let \( E' \) be a subset of \( E \).

(i) If \( z \) is the greatest element of \( E' \), then \( z = \sup(E') \).

(ii) If \( \sup(E') \in E' \), then \( \sup(E') \) is the greatest element of \( E' \).

We have a similar result for the least element and the greatest lower bound of \( E' \).

**Proof.** This result is a consequence of Proposition 2.21:

(i) If \( z \) is the greatest element of \( E' \), then \( z \) is the least element of \( \text{Maj}(E') \). It is thus the least upper bound of \( E' \).

(ii) The least upper bound of \( E' \) is the least element of \( \text{Maj}(E') \). If it belongs to \( E' \) it is thus the greatest element of \( E' \). \( \square \)

**Example 2.25**

1. Let \( \mathbb{N} \) be ordered by the divisibility relation (see Example 2.3 and Exercise 2.17). For this ordering, the greatest lower bound of a set of two integers always exists and is the greatest common divisor of these two integers. The least upper bound also always exists and is their least common multiple.

2. The least upper bound and the greatest lower bound do not always exist. Consider the set \( E = \{a, b, c, d\} \) ordered by : \( a \leq c, a \leq d, b \leq c, b \leq d \). See Figure 2.1.

![Figure 2.1](image)

Then \( \{a, b\} \) has neither a least upper bound nor a greatest lower bound; the same holds for \( \{c, d\} \).
EXAMPLE 2.26 Let $\mathcal{P}(E)$ be the set of subsets of $E$, ordered by inclusion. Let $E_i$ for $i \in I$ be a family of subsets of $E$. The least upper bound of this family is $\bigcup_{i \in I} E_i$ and its greatest lower bound is $\bigcap_{i \in I} E_i$.

Proposition 2.27 Let $E_i$, for $i \in I$, be a family of subsets of an ordered set and let $E' = \bigcup_{i \in I} E_i$ be its union. If each set $E_i$ has a least upper bound (resp. greatest lower bound) $e_i$, and if the set $\{e_i / i \in I\}$ has a least upper bound (resp. greatest lower bound) $e$, then $e$ is the least upper bound (resp. greatest lower bound) of $E'$.

Proof. We show this result only in the case of the least upper bound; the other case is completely similar.

First, we show that $e$ is an upper bound of $E'$. Let $x$ be any element of $E'$. It thus belongs to some $E_i$, and so $x \leq e_i \leq e$.

Now we let $z$ be any upper bound of $E'$ and show that $e \leq z$. Because $z$ is an upper bound of $E'$, it also is an upper bound of $E_i$, for any $i \in I$, and we have: $\forall i \in I, e_i \leq z$. Therefore, because $e$ is the least upper bound of $\{e_i / i \in I\}$, we have that $e \leq z$.

2.4 Well-ordered sets and induction

Well-founded sets form the general framework in which we can use proofs by induction. All induction principles stated in Chapter 3 are thus justified by the present section.

Definition 2.28 An ordered set $(E, \leq)$ is said to be well founded if there is no infinite strictly decreasing sequence of elements of $E$; $\leq$ is then said to have the well-founded ordering property or to be a well founded ordering. A total ordering $\leq$ having the well-founded ordering property is called a well ordering.

We now prove an important characterization of well-founded ordered sets.

Proposition 2.29 An ordered set $(E, \leq)$ is well founded if and only if any non-empty subset of $E$ has at least one minimal element.

Proof. It is equivalent to show the contrapositive of this result, namely, that $(E, \leq)$ has an infinite decreasing sequence if and only if there exists a non-empty subset having no minimal element. Assume that there exists a strictly decreasing infinite sequence $(x_n)_{n \in \mathbb{N}}$ in $E$. The set $X = \{x_n / n \in \mathbb{N}\}$ is a non-empty subset having no minimal element.

Conversely, assume that there exists a non-empty subset having no minimal element. Because $X$ has no minimal element, any element $x$ of $X$ is strictly larger
than at least one other element $y$ of $X$. Thus there exists a function $f$ from $X$ to $X$ verifying $\forall x \in X, f(x) < x$. (It suffices to choose one among the elements $y < x$ and to let $f(x) = y$.) Let $x_0 \in X$ (where $X$ is non-empty by hypothesis). For any integer $n$, we define $x_n = f^n(x_0)$. The sequence $(x_n)_{n \in \mathbb{N}}$ is strictly decreasing because $\forall n \in \mathbb{N}, x_n = f(x_{n-1}) < x_{n-1}$.

**Example 2.30**

1. The usual ordering is a well ordering on $\mathbb{N}$ but not on $\mathbb{Z}$.
2. $\mathbb{N}^2$ equipped with the product order $\leq$ (see Section 2.2.3) is well founded. Indeed, any element of $\mathbb{N}^2$ has a finite number of lower bounds. Consequently, there can exist no strictly decreasing infinite sequence. More generally, it is easy to see that the product of two well-founded sets is also well founded.
3. The lexicographic ordering $\preceq$ on $\mathbb{N}^2$ is defined by $(n,m) \preceq (n',m')$ if and only if $(n < n')$ or $(n = n'$ and $m < m')$. We note that if $n > 0$ then $(n,m)$ has infinitely many lower bounds. For instance, $\forall p \in \mathbb{N}$, $(n-1,p) \preceq (n,m)$. Nevertheless, the lexicographic ordering is a well ordering on $\mathbb{N}^2$. Indeed, let $X$ be a non-empty subset of $\mathbb{N}^2$, and let $n = \min\{p \in \mathbb{N} / \exists q \in \mathbb{N}, (p,q) \in X\}$ and $m = \min\{q \in \mathbb{N} / (n,q) \in X\}$. We easily verify that $(n,m)$ is the least element of $X$.

**Exercise 2.11** Let $<_1$ be a well ordering on $E_1$ and let $<_2$ be a well ordering on $E_2$; we define the **lexicographic product** $\preceq'$ of $<_1$ and $<_2$ on $E_1 \times E_2$ by $(n,m) \preceq' (n',m')$ if and only if $(n <_{1} n')$ or $(n = n'$ and $m <_{2} m')$. Verify that $\preceq'$ is a well ordering on $E_1 \times E_2$.

The induction principle for well-founded sets is stated in the following theorem.

**Theorem 2.31** Let $(E, \leq)$ be a well-founded set and let $P$ be an assertion depending on an element $x$ of $E$. ($P$ is called a predicate, see Chapter 5.) If the following property is verified:

\[(1) \quad \forall x \in E, \quad (\forall y < x, P(y)) \implies P(x),\]

then $\forall x \in E, P(x)$.

**Proof.** Let $X = \{x \in E / P(x) \text{ is false}\}$. If $X$ is non-empty, $X$ has a minimal element $x_0$. $\forall y < x_0, y \notin X$ and thus $P(y)$ is true. Using (1) we deduce that $P(x_0)$ is true, which contradicts $x_0 \in X$. Thus $X = \emptyset$, which means that $\forall x \in E, P(x)$ is true.

Unfortunately, sets equipped with their natural orderings are not always well founded. We have already seen that $\mathbb{Z}$ with the usual ordering is not well founded. It is of course possible to define well-founded orderings and even well orderings on $\mathbb{Z}$, but these orderings are not very natural. For instance, a well ordering $\preceq$ is defined on $\mathbb{Z}$ by using the usual ordering $\leq$ as follows:
**Complete sets and lattices**

- $\forall n > 0, \forall m > 0, n < m \iff n \leq m$ (positive integers are less than on $\mathbb{N}$).
- $\forall n < 0, \forall m \geq 0, n < m$ (negative integers are less than positive ones).
- $\forall n < 0, \forall m < 0, n < m \iff m < n$ (the inverse ordering on negative integers).

Below we give yet another example where the usual ordering is not a well-founded ordering.

**Example 2.32** Let $A$ be an alphabet with at least two letters $a$ and $b$. The free monoid $A^*$, together with the lexicographic ordering (see Example 2.4), is not well founded. Indeed, $(a^n b)_{n \in \mathbb{N}}$ is a strictly decreasing infinite sequence. Thus, proofs by induction on $A^*$ equipped with the lexicographic ordering will not be valid.

On the other hand, $A^*$ equipped with the prefix ordering (Example 2.4) is well founded. Finally, a well ordering on $A^*$ is defined by: $x < y$ if and only if

$$ (|x| < |y|) \quad \text{or} \quad (|x| = |y| \text{ and } x < y \text{ in the lexicographic ordering}). $$

Hence proofs by induction on $A^*$ using either the prefix ordering or the ordering $<$ will be valid.

**2.5 Complete sets and lattices**

**2.5.1 Complete sets and continuous functions**

**Definition 2.33** An ordered set $(E, \leq)$ is said to be a lattice (resp. complete lattice) if any finite subset (resp. any subset) of $E$ has a least upper bound and a greatest lower bound.

If $E$ is a lattice, then the greatest lower bound of $E$ is less than any element of $E$; hence a lattice has a least element that is denoted by $\bot$ and pronounced 'bottom'. Similarly, a lattice has a greatest element that is denoted by $\top$ and pronounced 'top'.

**Example 2.34** $\mathcal{P}(E)$ ordered by inclusion is a complete lattice.

**Exercise 2.12**

1. Show that an ordered set $(E, \leq)$ is a lattice if and only if any two-element subset of $E$ has a least upper bound and a greatest lower bound.
2. Show that an ordered set $(E, \leq)$ is a complete lattice if and only if any subset of $E$ has a least upper bound.

If an ordered set is a lattice, its least element $\bot$ is also the least upper bound of the empty set. Because the set $\text{Maj}(\emptyset)$ of upper bounds of the empty set
is the whole of $E$, $\bot$ is the least element of $E$.

Similarly, the greatest element $\top$ is also the greatest lower bound of the empty set.

**Definition 2.35** A mapping $f$ from an ordered set $(E_1, \leq_1)$ to an ordered set $(E_2, \leq_2)$ is said to be continuous (or, more precisely, sup-continuous) if it preserves the least upper bounds of non-empty subsets. In other words, if the subset $E' \neq \emptyset$ has a least upper bound $c = \sup(E')$, then $f(E') = \{ f(x) / x \in E' \}$ also has a least upper bound equal to $f(c)$.

**Remark 2.36** Since the least upper bound of the empty set is $\bot$, the condition "$f$ preserves the least upper bound of the empty set" is simply $f(\bot) = \bot$. This is a very exacting requirement that we will not demand for a continuous function.

Since in a complete lattice least upper bounds always exist, the continuity of a mapping between two complete lattices is then simply expressed by:

$$f(\sup(E)) = \sup(f(E)).$$

**Exercise 2.13** Show that any continuous function is monotonic.

Let $C(E)$ be the set of left segments of $E$ ordered by inclusion. Let $i$ be the mapping from $E$ to $C(E)$ defined by $i(x) = \{ y \in E / y \leq x \}$, and let $i(E)$ be the image of $E$ by $i$.

**Proposition 2.37** $C(E)$ is a complete set. The mapping $i$ is monotonic and is an isomorphism between $E$ and $i(E)$.

**Proof.** In order for $C(E)$ to be complete for inclusion, it suffices that any union of left segments is a left segment, and this clearly holds.

If $x \leq y$, it is clear that $i(x) \subseteq i(y)$ and thus $i$ is monotonic.

Conversely, if $i(x) \subseteq i(y)$, then because $x \in i(x)$ we have that $x \in i(y)$ and thus $x \leq y$. Hence $i(x) = i(y)$ implies $x \leq y$ and $y \leq x$, and thus $x = y$. □

However, $i$ is not always continuous, as shown by the next example.

**Example 2.38** Let $E = \mathbb{N}$, together with the usual ordering. For $n \in \mathbb{N}$, $i(n) = \{0, 1, \ldots, n\}$, and the only left segment that is not of this form is the whole of $\mathbb{N}$. We can thus identify $C(\mathbb{N})$ with the complete ordered set $\mathbb{N} = \mathbb{N} \cup \{ \omega \}$, where $\forall n \in \mathbb{N}$, $n < \omega$.

We may again consider the set $C(\mathbb{N})$ of left segments of $\mathbb{N}$ which is equal to $\{ i(n) / n \in \mathbb{N} \} \cup \{ \mathbb{N}, \mathbb{N} \}$. The mapping $i'$ from $\mathbb{N}$ to $C(\mathbb{N})$ is defined by $\forall n \in \mathbb{N}$, $i'(n) = \{0, 1, \ldots, n\}$ and $i'(\omega) = \mathbb{N}$. This mapping is not continuous. Indeed, in $\mathbb{N}$ the least upper bound of $\mathbb{N}$ is $\omega$, whilst in $C(\mathbb{N})$ the least upper bound of the set $\{ i'(n) / n \in \mathbb{N} \}$ is $\mathbb{N}$. 
2.5.2 Fixed points of monotone functions

Let \( f \) be a mapping from a set \( E \) to itself. A fixed point of \( f \) is an element \( x \) of \( E \) such that \( f(x) = x \).

If \( E \) is an ordered set, the set of fixed points of \( f \) is a subordered set of \( E \), possibly empty. If this subset has a least element, this least element is called the least fixed point of \( f \), and if it has a greatest element, this greatest element is called the greatest fixed point of \( f \).

**Theorem 2.39** If \( f \) is a monotone mapping from a complete ordered set to itself, then \( f \) has a greatest fixed point.

**Proof.** We verify that \( f \) has a greatest fixed point. Let

\[
X = \{ x \in E \mid x \leq f(x) \}
\]

and let \( z = \sup(X) \). By the definition of \( z \), we have \( \forall x \in X, x \leq z \) and hence, since \( f \) is monotonic, \( f(z) \leq f(x) \). As \( x \leq f(x) \), \( f(z) \) is an upper bound of \( X \), and hence \( z \leq f(z) \). We deduce that \( f(z) \leq f(f(z)) \), and hence \( f(z) \in X \) and thus \( f(z) \leq x \). It follows that \( z \) is a fixed point of \( f \). If \( z' \) is another fixed point, then \( z' \in X \) and thus \( z' \leq z \). \( \square \)

**Theorem 2.40** If \( f \) is a continuous mapping from a complete ordered set to itself, then \( f \) has a least fixed point. This least fixed point is equal to

\[
\sup(\{ f^n(\bot) \mid n \in \mathbb{N} \}).
\]

**Proof.** Let \( z = \sup(\{ f^n(\bot) \mid n \in \mathbb{N} \}) \). Because \( f \) is continuous,

\[
f(z) = \sup(\{ f^{n+1}(\bot) \mid n \in \mathbb{N} \}),
\]

and because \( \bot = f^0(\bot) \) is the least element of \( E \),

\[
\sup(\{ f^{n+1}(\bot) \mid n \in \mathbb{N} \}) = \sup(\{ f^n(\bot) \mid n \in \mathbb{N} \}) = z,
\]

which is thus a fixed point of \( f \). If \( y \) is another fixed point of \( f \), we first show by induction that \( \forall n \in \mathbb{N}, f^n(\bot) \leq y \); because \( \bot \) is the least element of \( E \), \( \bot = f^0(\bot) \leq y \); if \( f^n(\bot) \leq y \) then \( f^{n+1}(\bot) \leq f(y) = y \). Hence, \( z = \sup(\{ f^n(\bot) \mid n \in \mathbb{N} \}) \leq y \). \( \square \)
Theorem 2.41  If $E$ is a finite ordered set having a least element $\bot$, then for any monotone function $f$ from $E$ to itself there exists $k \leq \text{card}(E)$ such that the least fixed point of $f$ is $f^k(\bot)$.

Proof. Consider the sequence

$$\bot, f(\bot), f^2(\bot), \ldots, f^n(\bot), \ldots,$$

which is increasing because $f$ is monotone. If the sequence has two consecutive equal elements then it is stationary:

$$f^i(\bot) = f^{i+1}(\bot) \implies f^{i+1}(\bot) = f^{i+2}(\bot)$$

and thus, by induction, $\forall j \geq i, f^j(\bot) = f^i(\bot)$. In the $|E| + 1$ first elements of this sequence, there must be two consecutive elements that are equal. We thus have $f^k(\bot) = f^{k+1}(\bot)$ for $k \leq |E|$. The fact that $f^k(\bot)$ is less than any other fixed point of $f$ is proved as in the preceding theorem. \qed

Exercise 2.14
1. What is the value of $f(\bot)$ if $f$ preserves all least upper bounds?
2. If $f(\bot) = \bot$, what is the least fixed point of $f$? \diamond

Exercise 2.15  Consider $\mathcal{P}(E \times E)$ ordered by inclusion. Let $\mathcal{R}$ be a binary relation on $E$ and let $f$ be the mapping of $\mathcal{P}(E \times E)$ to itself defined by $f(X) = Id_E \cup \mathcal{R} \cdot X$.

Show that this mapping is continuous and that its least fixed point is $\mathcal{R}^*$. \diamond

2.5.3 Lattices

Definition 2.42  An ordered set is a lattice (see Definition 2.33 and Exercise 2.12) if any pair of elements has a least upper bound and a greatest lower bound.

We will sometimes denote by $x \lor y$, instead of sup($\{x, y\}$), the least upper bound of $x$ and $y$, and by $x \land y$ their greatest lower bound.

Example 2.43  $\mathbb{N}$ equipped with the divisibility ordering is a lattice. The binary operations $\lor$ and $\land$ are, respectively, the lcm (least common multiple) and the gcd (greatest common divisor). $\bot$ is $1$, and $\top$ is $0$. Indeed, $1$ divides any number $n$ because $1 \cdot n = n$, and any number $n$ divides $0$ because $n \cdot 0 = 0$.

Example 2.44  $\mathcal{P}(E)$ together with inclusion is a lattice. The binary operations $\lor$ and $\land$ are, respectively, $\cup$ and $\cap$.

If $E$ is a lattice, we may thus consider that $E$ is a set equipped with two binary operations $\lor$ and $\land$. 
Proposition 2.45 The operations \( \cup \) and \( \cap \) have the following properties:

- **Idempotence:** \( x \cup x = x \) and \( x \cap x = x \).
- **Commutativity:** \( x \cup y = y \cup x \) and \( x \cap y = y \cap x \).
- **Associativity:** \( (x \cup y) \cup z = x \cup (y \cup z) \) and \( (x \cap y) \cap z = x \cap (y \cap z) \).
- **Absorption:** \( x \cap (x \cup y) = x = (y \cap x) \cup x \).

Conversely, assume that on a set \( E \) there exist two binary operations \( \cup \) and \( \cap \) that have the four properties mentioned in the above proposition. Then we can order \( E \) in such a way that \( x \cup y \) and \( x \cap y \) are, respectively, the least upper bound and the greatest lower bound of \( x \) and \( y \). It suffices to let \( x \leq y \) if and only if \( x \cup y = y \), which, because of the absorption property, is equivalent to \( x \cap y = x \):

\[
x \cup y = y \implies x \cap (x \cup y) = x \cap y \implies x = x \cap y.
\]

Since \( \cup \) is idempotent, \( \leq \) is reflexive: from \( x \cup x = x \) we deduce that \( x \leq x \). From the commutativity of \( \cup \), we easily deduce that \( \leq \) is antisymmetric: \( x \leq y \) implies \( x \cup y = y \); \( y \leq x \) implies \( y \cup x = x \); as \( x \cup y = y \cup x \) we have \( x = y \). Finally, the transitivity of \( \leq \) is an immediate consequence of the associativity of \( \cup \):

\[
x \leq y \implies x \cup y = y; \quad y \leq z \implies y \cup z = z; \quad\text{hence } x = (x \cup y) \cup z = x \cup (y \cup z) = x \cup z\text{ and thus } x \leq z.
\]

Moreover, the least upper bound of \( x \) and \( y \) is indeed \( x \cup y \): as \( x \cap (x \cup y) = x \), we have \( x \leq (x \cup y) \), and for the same reasons, \( y \leq (x \cup y) \); if \( x \leq z \) and \( y \leq z \), we have \( z = x \cup z = y \cup z \) and thus \( z = x \cup z = z \cup y \), hence \( x \cup y \leq z \). The fact that the greatest lower bound of \( x \) and \( y \) is \( x \cap y \) is proved similarly.

From the associativity and the commutativity of \( \cup \) and \( \cap \), it immediately follows that in a lattice, any finite non-empty subset has a least upper bound and a greatest lower bound. This can be proved by induction on the number of elements of the finite subset by writing \( \{e_1, e_2, \ldots, e_n, e_{n+1}\} \) as the union of two sets \( \{e_1, e_2, \ldots, e_n\} \) and \( \{e_{n+1}\} \) and by applying Proposition 2.27.

**Exercise 2.16** Show that both operations \( \cap \) and \( \cup \) are monotone, i.e. if \( x \leq x' \) and \( y \leq y' \) then \( x \cap y \leq x' \cap y' \) and \( x \cup y \leq x' \cup y' \).

**Definition 2.46** A lattice is said to be **distributive** if \( \cap \) and \( \cup \) distribute over each other, i.e. if

(i) \( \forall x, y, z \), \( x \cup (y \cap z) = (x \cup y) \cap (x \cup z) \) and

(ii) \( \forall x, y, z \), \( x \cap (y \cup z) = (x \cap y) \cup (x \cap z) \).

These two conditions are indeed equivalent. If (i) is true, then (ii) is true. To show this, let us compute

\[
(x \cap y) \cup (x \cap z).
\]
By (i), this can be written
\[(x \cap y) \cup (x \cap y) \cup x\).

By using the absorption property, we obtain
\[x \cap ((x \cap y) \cup x)\]
and, by again applying (i),
\[x \cap ((x \cup z) \cap (y \cup z)).\]
By the associativity of \(\cap\), this is equal to
\[(x \cap (x \cup z)) \cap (y \cup z)\]
and, by again using the absorption property, this is equal to
\[x \cap (y \cup z)\).

The converse implication is proved similarly.

**Example 2.47** \(P(E)\) is a distributive lattice.

**Example 2.48** Assume \(E\) contains three elements \(a, b\) and \(c\) pairwise incomparable, a least element \(\bot\) and a greatest element \(\top\), see Figure 2.2.

![Figure 2.2](image)

It is a lattice because
\[\forall x, y \in \{a, b, c\}, x \not= y, x \cap y = \bot \text{ and } x \cup y = \top.\]
It is not distributive, because
\[a \cap (b \cup c) = a \cap \top = a,\]
while
\[(a \cup b) \cap (a \cup c) = \top \cap \top = \top.\]
Definition 2.49 A lattice $E$ is said to be complemented if

(i) it has a least element $\bot$ and a greatest element $\top$, with $\bot \neq \top$ and
(ii) there exists a mapping $\nu$ from $E$ to $E$ such that

- $\forall x \in E, \ x \cap \nu(x) = \bot$ and
- $\forall x \in E, \ x \cup \nu(x) = \top$.

Example 2.50
1. The lattice $P(E)$ is complemented. Its least element is the empty set, its greatest element is $E$ and the mapping $\nu$ is the usual complement operation.
2. The lattice of Example 2.48 is complemented. Let $\nu(\bot) = \top, \nu(\top) = \bot, \nu(a) = b = c$ and $\nu(c) = a$.

Exercise 2.17 The set $\mathbb{N}$ equipped with the divisibility ordering is a lattice.
1. Is it distributive?
2. Is it complemented?

Exercise 2.18
1. Show that the set of equivalence relations on a set $E$ is a lattice for inclusion.
2. Is it distributive?
3. Is it complemented?

Exercise 2.19 Show that, in a complemented lattice,

- $\nu(\top) = \bot$ and $\nu(\bot) = \top$.

Exercise 2.20 Show that, in a complemented lattice, $\forall x, \nu(x) \neq x$.

Proposition 2.51 If a complemented lattice is distributive, there exists exactly one operation of complement $\nu$. This operation, moreover, verifies

(i) *involution:* $\forall x, \ \nu(\nu(x)) = x$,
(ii) *De Morgan's laws:* $\forall x, y, \ \nu(x \cup y) = \nu(x) \cap \nu(y)$ and $\nu(x \cap y) = \nu(x) \cup \nu(y)$
(iii) *antimonicity:* $x \leq y \iff \nu(y) \leq \nu(x)$.

Proof.
(i) We first show that in a distributive lattice with a least element and a greatest element, we have the property

$\forall x, y, z, \ x \cap y = \bot$ and $x \cup z = \top \implies y \leq z$.

Indeed, $z = z \cup \bot = z \cup (x \cap y) = (z \cup x) \cap (z \cup y) = \top \cap (z \cup y) = z \cup y$, and thus $y \leq z$. 
Assume now that there exist two mappings $\nu$ and $\mu$ verifying

$$\forall x \in E, \quad x \cap \nu(x) = \bot,$$
$$\forall x \in E, \quad x \cup \nu(x) = T,$$
$$\forall x \in E, \quad x \cap \mu(x) = \bot,$$
$$\forall x \in E, \quad x \cup \mu(x) = T.$$ 

Because $x \cap \nu(x) = \bot$ and $x \cup \mu(x) = T$, we have that $\nu(x) \leq \mu(x)$. Similarly, $x \cap \mu(x) = \bot$ and $x \cup \nu(x) = T$, and hence $\mu(x) \leq \nu(x)$ and thus $\mu(x) = \nu(x)$. Because $\nu(x) \cap x = \bot$ and $\nu(x) \cup \nu(\nu(x)) = T$, we have that $x \leq \nu(\nu(x))$, and, for similar reasons, $\nu(\nu(x)) \leq x$.

(ii) In order to show the De Morgan’s laws, it suffices to show, taking into consideration the uniqueness of the complement, that

1. \( (x \cup y) \cap (\nu(x) \cap \nu(y)) = \bot \) and \( (x \cup y) \cup (\nu(x) \cap \nu(y)) = T \) and
2. \( (x \cap y) \cap (\nu(x) \cup \nu(y)) = \bot \) and \( (x \cap y) \cup (\nu(x) \cup \nu(y)) = T \).

We show only the first identity; the second one can be proved similarly:

\[
(x \cup y) \cap (\nu(x) \cap \nu(y)) = (x \cap \nu(x) \cap \nu(y)) \cup (y \cap \nu(x) \cap \nu(y)) = \bot \cup \bot = \bot.
\]

\[
(x \cup y) \cup (\nu(x) \cap \nu(y)) = (x \cup y \cup \nu(x)) \cap (x \cup y \cup \nu(y)) = T \cap T = T.
\]

(iii) To show the last equivalence, notice that

\[
x \leq y \quad \iff \quad x = x \cap y \quad \iff \quad \nu(x) = \nu(x \cap y) = \nu(x) \cup \nu(y)
\]

\[
\iff \quad \nu(y) \leq \nu(x). \quad \Box
\]

**Example 2.52** The lattice of Example 2.48 is not distributive, and there indeed exist at least two operations of complement. For instance $\mu(\bot) = T$, $\mu(T) = \bot$, $\mu(a) = a$, $\mu(b) = a$, $\mu(c) = b$. 
CHAPTER 3

RECURSION AND INDUCTION

Inductive and recursive definitions are the construction of finite objects from other finite objects, according to some given rules. Inductive definitions also provide us with a way of grasping infinite objects defined by recursive definitions: indeed, since only finite objects can be handled by computer science, such infinite objects are studied via sequences of finite approximations; usually, the finite approximations are also defined by an inductive definition.

Inductive proofs enable one to reason about inductively defined objects. Because computer science makes extensive use of such objects, this chapter is essential. For instance, recursive definitions constantly occur in data structures and in the conception of recursive programs (in functional languages such as LISP, but also in logic programming and PROLOG). The proofs of such recursive programs are then inductive proofs, as are the proofs of termination of iterative programs (sometimes called top-down induction).

However, the various induction principles are not stated in detail in textbooks (to our knowledge); this is why we cannot recommend any handbook for the present chapter.

In this chapter we review the two basic induction principles on the integers: the induction principle and the complete induction principle. We introduce the notion of definition of a set by induction and we show how to prove properties of sets defined by induction. As a special case, we introduce the concept of 'set of terms' which is a major tool in computer science. Finally, we present the concept of closure, which is a general way of looking at inductive definitions.
3.1 Reasoning by induction in \( \mathbb{N} \)

3.1.1 First induction principle

In \( \mathbb{N} \), the first induction principle, also called the mathematical induction principle, is a most useful way of reasoning. We will use both terminologies 'proof by induction' and 'proof by mathematical induction' for proofs using this first induction principle.

**Theorem 3.1** Let \( P(n) \) be a predicate (a property) depending on the integer \( n \). If both the following conditions hold:

1. \( P(0) \) is true, and
2. \( \forall n \in \mathbb{N}, \text{the implication} \quad (P(n) \implies P(n + 1)) \text{ is true,} \)

then \( \forall n \in \mathbb{N}, P(n) \) is true.

(B) is called the basis step of the induction and (I) is called the inductive step (or sometimes 'going from \( n \) to \( n + 1 \')).

Here we give a direct proof of this result, but it is worth while noting that it can also be justified by using Proposition 3.1.1 and the inductive definition of \( \mathbb{N} \) given in Example 3.9.

**Proof.** By contradiction. We consider the set

\[ X = \{ k \in \mathbb{N} / P(k) \text{ is false} \}. \]

If \( X \) is non-empty, it has a least element \( n \). By condition (B), \( n \neq 0 \).

Thus \( n - 1 \) is an integer and \( n - 1 \not\in X \), namely, \( P(n - 1) \) is true.

Using (I), we then obtain: \( P(n) \) is true, which contradicts \( n \in X \). Therefore, \( X \) is empty, and this proves the theorem.

\( \square \)

(I) does not assert that \( P(n + 1) \) or \( P(n) \) hold, but only that if \( P(n) \) is true, then \( P(n + 1) \) must be true. Only after proving (I) and (B) can we conclude that, for all \( n \geq 0 \), \( P(n) \) is true. Usually the basis (B) is easy to prove, and the difficult part is the inductive step (I). However, one should not forget to prove the basis (B), otherwise one will obtain false results; for instance, we verify immediately that \( \forall n \geq 0, (n > 10 \implies n + 1 > 10) \). It is none the less false that \( \forall n \geq 0, n > 10 \). (See also Exercise 3.6.)

**Remark 3.2** We can prove a slightly more general form of Theorem 3.1 similarly. Let \( n_0 \) be an integer greater than or equal to 0, if both following conditions hold:

1. \( P(n_0) \) is true, and
2. \( \forall n \geq n_0, \text{the implication} \quad (P(n) \implies P(n + 1)) \text{ is true,} \)

then \( \forall n \geq n_0, P(n) \) is true.
EXAMPLE 3.3  We wish to compute the sum \( S_n = 1 + 2 + \cdots + n \). We note that \( 2S_1 = 2 = 1 \times 2, \) \( 2S_2 = 6 = 2 \times 3, \) \( 2S_3 = 12 = 3 \times 4. \) We then conjecture that \( \forall n > 0, \) \( 2S_n = n(n + 1) \). We prove this by induction. Let \( P(n) \) be the property \( '2S_n = n(n + 1)' \), we verify that

(B) \( 2S_1 = 1 \times 2, \)

(i) Let \( n \geq 1. \) We assume \( P(n) \). We have

\[
2S_{n+1} = 2S_n + 2(n + 1) = n(n + 1) + 2(n + 1) = (n + 1)(n + 2),
\]

hence \( P(n + 1) \) is true.

We can then conclude that \( \forall n \geq 1, \) \( P(n) \).

EXERCISE 3.1  Adopting the convention that \( \forall r \in \mathbb{R}, r^0 = 1, \) prove by induction that:

1. \( \forall r \in \mathbb{R}, \forall n \in \mathbb{N}, \) \( S_n = \sum_{i=0}^{n} r^i = \begin{cases} \frac{n+1}{r-1} & \text{if } r \neq 1, \\ \frac{r^{n+1} - 1}{r - 1} & \text{if } r = 1. \end{cases} \)

2. \( \forall r \in \mathbb{R}, \forall n \in \mathbb{N}, \) \( T_n = \sum_{i=0}^{n} ir^i = \begin{cases} \frac{n(n+1)/2}{r} & \text{if } r = 1, \\ \frac{nr^{n+2} - (n+1)r^{n+1} + r}{(r-1)^2} & \text{if } r \neq 1. \end{cases} \)

EXERCISE 3.2  
1. Show that \( \forall n \geq 1, \) \( S_n = 1^3 + 3^3 + \cdots + (2n - 1)^3 = 2n^4 - n^2. \)

2. Compute \( T_n = \sum_{k=1}^{n} \frac{1}{4k^2 - 1} \) for all \( n \geq 1. \)

EXERCISE 3.3  We consider the polynomial with real-valued coefficients

\[ P(x) = \frac{1}{3} x^3 + ax^2 + bx. \]

1. Find \( a \) and \( b \) such that \( \forall x \in \mathbb{R}, P(x + 1) - P(x) = x^2. \) We assume that this property holds in the remainder of the exercise.

2. Show that \( \forall n \in \mathbb{N}, P(n) \) is an integer.

3. \( \forall n \geq 0, \) \( S_n = \sum_{k=0}^{n} k^2. \) Show that \( \forall n \geq 0, \) \( S_n = P(n + 1) = \frac{n(n+1)(2n+1)}{6}. \)

NOTATION We will write \( p \mid n \) to denote the fact that \( p \) divides \( n, \) where \( p \) and \( n \) are integers.

EXERCISE 3.4  Let \( n \geq 1 \) and let \( A \subseteq \{1, 2, \ldots, 2n\} \) be such that \( |A| \geq n + 1. \) Show that there exist two distinct integers \( a \) and \( b \) in \( A \) such that \( a \mid b. \)

EXERCISE 3.5  Let \( \mathcal{R} \) be a binary relation on a set \( E. \) Let \( \mathcal{R}^0 = \text{Id}_E, \) \( \mathcal{R}^{i+1} = \mathcal{R} \mathcal{R}^i. \)
Show that \( \forall i, j \geq 0, \mathcal{R}^{i,j} = \mathcal{R}^j \mathcal{R}^i \). ☐

**Exercise 3.6** We consider the properties \( P(n) \): \( '9 | 10^n - 1' \) and \( Q(n) \): \( '9 | 10^n + 1' \).
1. Show that \( \forall n \in \mathbb{N}, P(n) \implies P(n + 1) \) and \( Q(n) \implies Q(n + 1) \).
2. Find the values of \( n \) for which \( P(n) \) (resp. \( Q(n) \)) is true. ☐

**Exercise 3.7** Find the error in the following proof by induction. Let \( P(n) \) be the property 'in any group consisting of \( n \) individuals, all the people are of the same age'.

\( \text{(B)} \) \( P(1) \) is clearly true.
\( \text{(I)} \) Let \( n \) be such that \( P(n) \) is true. Let \( G \) be a group of \( n + 1 \) individuals numbered from 1 to \( n + 1 \). Let \( G_1 \) (resp. \( G_2 \)) be the group consisting of the \( n \) first (resp. last) individuals in \( G \). Since \( P(n) \) is true, all the people of \( G_1 \) (resp. \( G_2 \)) are of the same age. Moreover, individual number \( n \) is a member of both \( G_1 \) and \( G_2 \). Thus all the people of \( G \) are of the same age as individual number \( n \), and this proves \( P(n + 1) \).

We hence deduce that \( \forall n \geq 1, P(n) \). ☐

### 3.1.2 Second induction principle

In the first induction principle (see Theorem 3.1), the truth of \( P(n + 1) \) depends only upon that of \( P(n) \), i.e. if proposition \( P \) is true at step \( n \) it is also true at step \( n + 1 \). More complex cases may occur, where in order to establish that \( P \) is true at step \( n + 1 \) we have to explicitly use the fact that \( P \) is true at steps \( 0, 1, \ldots, n - 1, n \). In such a case, it is more convenient to use the second induction principle, which is stated as follows:

**Theorem 3.4** Let \( P(n) \) be a property depending on the integer \( n \). If the following proposition is verified:

\[
(\forall') \quad \forall n \in \mathbb{N}, \quad \left( \forall k < n, P(k) \right) \implies P(n)
\]

then \( \forall n \in \mathbb{N}, P(n) \) is true.

This second induction principle is a consequence of Theorem 2.31 because the usual ordering on \( \mathbb{N} \) is a well ordering (see Section 2.4).

**Remark 3.5**

1. The fact that the second induction principle has no basis step may seem suspicious; in fact, the basis step is 'hidden' in \( (\forall') \). Indeed, verifying \( (\forall') \) implies proving that for \( n = 0 \) \( (\forall k < 0, P(k)) \implies P(0) \). But \( (\forall k < 0, P(k)) \) is true because there is no negative integer \( k < 0 \), hence we must prove that \( P(0) \) is true. Here we see a typical instance of reasoning with the empty set: \( (\forall k < 0, P(k)) \) can be rewritten as \( (k < 0 \implies P(k)) \), or, since there is no negative integer \( k < 0 \) in \( \mathbb{N} \), \( (k \in \emptyset \implies P(k)) \), which is true because \( k \in \emptyset \) is always false; more generally, any 'empty' statement of the form \( (\forall x \in \emptyset, P(x)) \) always holds.
As for the first induction principle, we may start from any integer \( n_0 \). We must then check that:

\[
(P_{n_0}^\prime) \quad \forall n \geq n_0, \quad \left( \forall k \in \{n_0, \ldots, n-1\}, P(k) \right) \implies P(n)
\]

and deduce \( \forall n \geq n_0, P(n) \).

3. On \( \mathbb{N} \), the two induction principles are equivalent (i.e. each can be shown to hold from the other), but (see Section 2.4) only the second induction principle generalizes to more general ordered sets.

Exercise 3.8 Verify that the two induction principles entail the same properties on \( \mathbb{N} \), i.e. that, if \( P(n) \) is a property depending on the integer \( n \), \( P \) verifies \((P')\) if and only if \( P \) verifies \((B)\) and \((I)\). The two induction principles thus have the same power on \( \mathbb{N} \); we will say that they are equivalent on \( \mathbb{N} \).

\( \diamond \)

Example 3.6 The second principle is simpler to use when the property of the elements at step \( n \) involves simultaneously the property of the elements at steps \( n-1 \), \( n-2 \), \ldots, etc. For instance, we can quite easily show that any integer \( n \geq 2 \) can be written as a product of primes. Denote by \( P(n) \) the property ' \( n \) can be written as a product of primes'; it suffices to verify \((I_1')\), see Remark 3.5 (2). Let \( n \geq 2 \). Assume \( \forall k \in \{2, \ldots, n-1\}, P(k) \). Two cases can occur:

- \( n \) is a prime. Then \( n \) can clearly by written as a product of primes (a single prime is also considered as a product).
- \( n \) is not a prime. Then we can write \( n = ab \), where \( a \) and \( b \) are two integers between 2 and \( n-1 \). \( P(a) \) and \( P(b) \) are true by hypothesis, and so we deduce that \( n \) can also be written as the product of the decompositions of \( a \) and \( b \).

Exercise 3.9
1. Show that \( \forall n \in \mathbb{N} \), \( (n+1)^2 - (n+2)^2 - (n+3)^2 + (n+4)^2 = 4 \).
2. Deduce that any integer \( m \) can be written as sums and differences of squares \( 1^2, 2^2, \ldots, n^2 \) for an \( n \), i.e.

\[
\forall m \in \mathbb{N}, \exists n \in \mathbb{N}, \exists \varepsilon_1, \ldots, \varepsilon_n \in \{-1, 1\}, \quad m = \varepsilon_1 1^2 + \varepsilon_2 2^2 + \cdots + \varepsilon_n n^2.
\]

(Hint: first show the result for \( m \in \{0, 1, 2, 3\} \).)

\( \diamond \)

Exercise 3.10 Let \( A^* \) be the free monoid on the alphabet \( A \) (see Definition 1.15). Show that \( \forall u, v \in A^* \), \( u \cdot v = v \cdot u \iff \exists w \in A^*, \exists p, q \in \mathbb{N}: u = w^p \text{ and } v = w^q \).

\( \diamond \)

Exercise 3.11 Let \( A^* \) be the free monoid on the alphabet \( A \) (see Definition 1.15 and Example 2.4). A language is a subset of \( A^* \). If \( L_1 \) and \( L_2 \) are two languages of \( A^* \), we define their concatenation by:

\[
L_1 \cdot L_2 = \{ u \cdot v \mid u \in L_1, v \in L_2 \}.
\]
Language concatenation is an associative operation with unit \{e\}. We can then define the powers of language \(L\) as follows:

\[ L^0 = \{e\} \quad \text{and} \quad \forall n \in \mathbb{N}, L^{n+1} = L^n \cdot L = L \cdot L^n. \]

Finally, the star of language \(L\) is the submonoid of \(A^*\) generated by \(L\), i.e.

\[ L^* = \bigcup_{n \in \mathbb{N}} L^n. \]

Let \(L\) and \(M\) be two languages on \(A^*\) such that \(e \notin L\). Show that in \(\mathcal{P}(A^*)\), the equation \(X = L \cdot X \cup M\) has as its unique solution the language \(L^* \cdot M\). \(\diamondsuit\)

### 3.2 Inductive definitions and proofs by structural induction

In the present section, we introduce inductive definitions of sets and functions and proofs by induction on inductively defined structures.

#### 3.2.1 Inductively defined sets

Quite often in computer science subsets are inductively (recursively) defined. In particular, many data structures may be so defined. Intuitively, the inductive definition of a subset \(X\) of a set explicitly gives some elements of the set \(X\) together with ways of constructing new elements of \(X\) from already known elements. Such a definition will hence have the following intuitive generic form:

(B) Some elements of the set \(X\) are explicitly given (basis of the recursive definition).

(I) The other elements of the set \(X\) are defined in terms of elements already in the set \(X\) (inductive steps of the recursive definition).

Formally, we have the following definition.

**Definition 3.7** Let \(E\) be a set. An inductive definition of a subset \(X\) of \(E\) consists of giving:

- a subset \(B\) of \(E\) and
- a set \(K\) of operations \(\Phi: E^{a(\Phi)} \rightarrow E\), where \(a(\Phi) \in \mathbb{N}\) is the arity (or rank) of \(\Phi\).

\(X\) is defined as the least set verifying the following assertions (B) and (I):

(B) \(B \subseteq X\).

(I) \(\forall \Phi \in K, \forall x_1, \ldots, x_{a(\Phi)} \in X, \Phi(x_1, \ldots, x_{a(\Phi)}) \in X.\)
The set thus defined is

\[ X = \bigcap_{Y \in \mathcal{F}} Y, \]

where \( \mathcal{F} = \{ Y \subseteq E \mid B \subseteq Y, \text{ and } Y \text{ verifies (I) with } X \text{ replaced by } Y \} \).

Henceforth, we modify assertions (B) and (I) slightly, and we denote an inductive definition by the form

(B) \( x \in X \quad (\forall z \in B), \)

(I) \( x_1, \ldots, x_{a(\Phi)} \in X \implies \Phi(x_1, \ldots, x_{a(\Phi)}) \in X \quad (\forall \Phi \in K). \)

Remark 3.8. The set \( \mathcal{F} \) is non-empty because it contains \( E \); indeed, \( E \) clearly verifies (B) \( (B \subseteq E) \) and (I). Moreover, if subsets of a set verify a condition then their intersection also verifies that condition. Indeed, let \( \mathcal{Y} \) be a set of subsets \( Y \) of \( E \) verifying (B) and (I), and let \( Z = \bigcap_{Y \in \mathcal{Y}} Y \). Since \( B \) is included in any set \( Y \) of \( \mathcal{Y} \), \( B \) is also included in \( Z = \bigcap_{Y \in \mathcal{Y}} Y \) and hence \( Z \) verifies (B); if \( x_1, \ldots, x_{a(\Phi)} \in Z \), then for any \( Y \in \mathcal{Y} \), \( x_1, \ldots, x_{a(\Phi)} \in Y \), whence \( \Phi(x_1, \ldots, x_{a(\Phi)}) \in Y \), and hence \( \Phi(x_1, \ldots, x_{a(\Phi)}) \in Z \) and \( Z \) verifies (I). Thus \( \bigcap_{Y \in \mathcal{F}} Y \), where \( \mathcal{F} \) is the above-defined set of subsets of \( E \), is indeed the least subset of \( E \) verifying the conditions (B) and (I).

Note that, in general, many sets verify these conditions. Consider, for instance, the conditions:

(B) \( 0 \in P, \)

(I) \( n \in P \implies n + 2 \in P. \)

There are infinitely many subsets of \( \mathbb{N} \) verifying these properties: \( \mathbb{N}, \mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1, 3\}, \mathbb{N} \setminus \{1, 3, 5\} \), etc., are such subsets. The subset \( P \) defined by (B) and (I) is not among them because it consists of the set of even integers.

We consider now examples of inductive definitions.

Example 3.9

1. The subset \( X \) of \( \mathbb{N} \) inductively defined by

\( (B) \quad 0 \in X, \)

\( (I) \quad n \in X \implies n + 1 \in X, \)

is identical to \( \mathbb{N} \). (B) and (I) thus constitute an inductive definition of \( \mathbb{N} \).

2. The subset \( X \) of the free monoid \( A^* \) (see Definition 1.15) inductively defined by

\( (B) \quad e \in X, \)

\( (I) \quad u \in X \implies \forall a \in A, u \cdot a \in X, \)

is identical to \( A^* \). (B) and (I) thus constitute an inductive definition of \( A^* \).
3. Let $A = \{(),\}$ be the alphabet consisting of two parentheses (left and right). The set $D \subseteq A^* \equiv$ strings of balanced parentheses, the so-called Dyck language, is defined by

(B) $\varepsilon \in D,$
(I) if $x$ and $y$ belong to $D,$ then $(x)$ and $xy$ also belong to $D.$

4. Let $E$ be the set of expressions all of whose subexpressions are included in parentheses and which are formed from identifiers in a set $A$ and the two operators $+$ and $\times.$ $E$ is the subset of $\left( A \cup \{+, \times, (, )\} \right)^*$ inductively defined by

\begin{itemize}
\item[(B)] $A \subseteq E,$
\item[(I)] if $e$ and $f$ are in $E$ then $(e + f)$ and $(e \times f)$ are also in $E.$
\end{itemize}

We note that in computer science syntactic definitions are almost always inductive. We often use the BNF (Backus–Naur Form) notation for describing them. For instance, the set $E$ is defined by

$$E ::= A \mid (E + E) \mid (E \times E),$$

where the symbol $\mid$ is read 'or'.

5. The set $BT$ of labelled binary trees on the alphabet $A$ is the subset of $\left( A \cup \{\emptyset, (, ), ,\} \right)^*$ inductively defined by

\begin{itemize}
\item[(B)] $\emptyset \in BT$ (the empty tree),
\item[(I)] if $l, r \in BT$ then $\forall a \in A, (a, l, r) \in BT$ (the tree with root $a$, left child $l$ and right child $r$).
\end{itemize}

The set $BT$ thus defined is a language on the alphabet $A \cup \{\emptyset\} \cup \{(\) \cup \{\} \cup \{,\}.$ In general, we use a very intuitive graphical representation of trees. To simplify, tree $(a, \emptyset, \emptyset)$ will simply be denoted by $a$. For instance, the trees $a, (a, a, b), (a, \emptyset, (b, c, \emptyset))$ and $(a, (a, b, c), d)$ can be drawn as in Figure 3.1

![Figure 3.1](image)

Binary trees are extensively used in algorithmics.

**Exercise 3.12** Let $A$ be an alphabet. We recursively define the sets $(BT_n)_{n \in \mathbb{N}}$ by

- $BT_0 = \{\emptyset\},$
- $BT_{n+1} = BT_n \cup \{(a, l, r) / a \in A, l, r \in BT_n\}.$

Show that $X = \bigcup_{n \in \mathbb{N}} BT_n$ is the set $BT$ of binary trees on alphabet $A.$

$\Diamond$
Inductive definitions and proofs by structural induction

The preceding example illustrates a more general phenomenon. Indeed, in most cases, the elements of an inductively defined set can be obtained from the basis by applying finitely many inductive steps. We have the following theorem.

**Theorem 3.10** If \( X \) is defined by the conditions (B) and (I), any element of \( X \) can be obtained from the basis by applying finitely many inductive steps.

**Proof.** We define the sets

- \( X_0 = B \),
- \( X_{n+1} = X_n \cup \{ \Phi(x_1, \ldots, x_n(\Phi)) \mid x_1, \ldots, x_n(\Phi) \in X_n \text{ and } \Phi \in K \} \).

As in Exercise 3.12, we show by induction that \( \forall n \in \mathbb{N}, X_n \subseteq X \), and we deduce that \( X = \bigcup_{n \in \mathbb{N}} X_n \subseteq X \). The set of elements obtainable from the basis by applying finitely many inductive steps is exactly \( X \). We must now show that \( X \) verifies (B) and (I). As \( B = X_0 \subseteq X \), \( X \) verifies (B). Let \( \Phi \in K \) and let \( x_1, \ldots, x_n(\Phi) \in X \). Each \( x_i \) belongs to a set \( X_{n_i} \subseteq X \). Let \( n = \sup\{n_1, \ldots, n(\Phi)\} \). Then \( x_1 \in X_n \), thus \( \Phi(x_1, \ldots, x_n(\Phi)) \in X_{n+1} \subseteq X \), and \( X \) verifies (I).

### 3.2.2 Inductive proofs

The induction principle is a generalization of the induction principle on the integers and is designed to prove the properties of inductively defined sets. The proof by induction exactly follows the inductive definition of the set; this is why it is also called **structural induction**.

**Proposition 3.11** Let \( X \) be an inductively defined set (see Definition 3.7), and let \( P(x) \) be a predicate expressing a property of the elements \( x \) of \( X \). If the following conditions hold:

\[(B^\prime) \quad P(x) \text{ is true for each } x \in B, \quad \text{and} \]
\[(I^\prime) \quad (P(x_1), \ldots, P(x_n(\Phi))) \quad \Rightarrow \quad P(\Phi(x_1, \ldots, x_n(\Phi))) \quad \text{for each } \Phi \in K, \]

then \( P(x) \) is true for any \( x \) in \( X \).

Verifying (B') and (I') constitutes a proof by induction of property \( P \) on \( X \).

**Proof.** Let \( Y \) be the set of \( x \)s such that \( P(x) \) is true. We have that \( B \subseteq Y \) (by (B')), and that \( Y \) verifies the inductive clauses (I) of the definition of \( X \) (by (I')); hence \( Y \supseteq X \) (see Definition 3.7).

**Remark 3.12** If we consider that the non-negative integers are defined as in Example 3.9, the first induction principle on the integers corresponds to the above definition. All the proofs by mathematical induction seen in Section 3.1 are hence examples of proofs by induction according to the present definition.
Example 3.13 We show by induction that any string of the Dyck language has as many left parentheses as right parentheses (see Example 3.9). For \( x \) in \( D \), we denote by \( l(x) \) (resp. \( r(x) \)) the number of left (resp. right) parentheses in \( x \). (The inductive definition of these functions is left to the reader.) Finally, let \( P(x) \) be the property \( 'r(x) = l(x)' \). We prove by induction that \( P(x) \) holds for any \( x \) in \( D \).

(B) The only element of the basis is \( \varepsilon \), and it satisfies \( P \) because

\[
r(\varepsilon) = l(\varepsilon) = 0.
\]

(1) Let \( x, y \in D \) be such that \( r(x) = l(x) \) and \( r(y) = l(y) \) and let \( z = xy \). We have that \( r(z) = r(x) + r(y) = l(x) + l(y) = l(x) \), and so \( P(z) \) is thus verified. The case where \( z = (x) \) can be verified in the same way: \( r(x) = r(x) + 1 = l(x) + 1 = l(x) \).

We deduce that \( \forall x \in D, l(x) = r(x) \).

Exercise 3.13 Characterization of the Dyck language. We use the notations of Examples 3.9 and 3.13. Show that \( D = L \), where

\[
L = \{ x \in A^* / l(x) = r(x) \text{ and } l(y) \geq r(y) \text{ for any prefix } y \text{ of } x \}.
\]

Exercise 3.14 Let \( BT \) be the set of binary trees and let \( h, n, f \) be the functions that give the height (see Example 3.24), the number of nodes (nodes are also called vertices) and the number of leaves of a tree respectively. Show that

1. \( \forall x \in BT, n(x) \leq 2^{h(x)} - 1 \),
2. \( \forall x \in BT, f(x) \leq 2^{h(x)} - 1 \).

Exercise 3.15 A binary tree is strict if it is non-empty and if it has no node with a single non-empty child. For instance, the trees of the Figure 3.5 (page 51) are strict, while the tree of the Figure 3.2 is non-strict.

```
        a
       / \  
      b   c
     |
    d
```

Figure 3.2

1. Give a definition of the set \( SBT \) of strict trees.
2. Show that \( \forall x \in SBT, n(x) = 2f(x) - 1 \).

Exercise 3.16 A binary tree is said to be balanced if for each node in the tree, the difference between the heights of its left and right subtrees is at most \( 1 \). For instance, Figure 3.3 represents balanced trees of height 3, 4, and 5. (The labels of nodes are not represented.)
1. Give a definition of the set $BBT$ of balanced binary trees.

   We define $(u_n)_{n \in \mathbb{N}}$ by: $u_0 = 0, u_1 = 1$ and
   
   $$v_n \geq 0, \quad u_{n+2} = u_{n+1} + u_n + 1.$$ 

   Show that $\forall x \in BBT, \, n(x) \geq u_{h(x)}$, where $h$ and $n$ are the functions that give the height and the number of nodes of a tree respectively.

**Exercise 3.17** Let $A^*$ be the free monoid on alphabet $A$ (see Definition 1.15, Example 2.4 and Exercise 3.11). The set $\text{Rat}$ of rational languages is defined inductively by:

1. $\emptyset \in \text{Rat}$ and $\forall a \in A, \{a\} \in \text{Rat}$,
2. $L, M \in \text{Rat} \implies \overline{L \cup M} \in \text{Rat}$,
3. $L, M \in \text{Rat} \implies \overline{L \cdot M} \in \text{Rat}$,
4. $L \in \text{Rat} \implies \overline{L^*} \in \text{Rat}$.

   The mirror image (or reverse) of language $L$ is the set $\overline{L} = \{\overline{u} \mid u \in L\}$, where, if $u = a_1a_2\cdots a_n$, then $\overline{u} = a_n\cdots a_2a_1$, see Exercise 3.18.

   Show that $L \in \text{Rat} \implies \overline{L} \in \text{Rat}$.

2. We denote by $LP(L)$ the set of prefixes (left factors) of strings in the language $L$, i.e. $LP(L) = \{u \in A^* \mid \exists u \in L \text{ such that } u \text{ is a prefix of } v\}$. Show that $L \in \text{Rat} \implies LP(L) \in \text{Rat}$.

### 3.3 Terms

In the present section we study a particular instance of definition by structural induction that is quite useful in computer science: the definition of terms. Many structures use terms in their representation.

#### 3.3.1 Definition

Let $F = \{f_0, \ldots, f_n, \ldots\}$ be a set of operation symbols. With each symbol $f$ is associated a finite arity (or rank) $a(f) \in \mathbb{N}$ representing the number of arguments of $f$. $F_n$ denotes the set of arity $n$ operation symbols.

Let $U$ be the set of all strings of symbols in $F \cup \{('\cdot'), ',', '\}_i$. Let $F_i$ be the set of symbols of arity $i$. 

Definition 3.14 The set $T$ of terms built on $P$ is inductively defined by:
(B) $B = F_0 \subseteq T$,
(I) $\forall f \in F_n$, $\Phi_f(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$ for $t_1, \ldots, t_n$ in $T$.

$\Phi_f(t_1, \ldots, t_n)$ represents the result of operation $\Phi$ applied to the $n$-tuple of terms $(t_1, \ldots, t_n)$, i.e. a semantic object, whilst $f(t_1, \ldots, t_n)$ represents a string of formal symbols constituting a term, i.e. a syntactic object.

A term may be represented as a tree; for instance, $f(t_1, \ldots, t_n)$ may be pictured as in Figure 3.4.

![Figure 3.4](image)

3.3.2 Interpretations of terms

Let $V$ be an arbitrary set. With each element $f$ of $F_0$ we associate an element $h(f)$ of $V$. With each element $f$ of $F_i$ with $i > 0$ we associate a mapping $h_f: V^i \rightarrow V$.

Proposition 3.15 There exists a unique function $h^*$ from $T$ to $V$ such that:

(B') If $t \in F_0$, $h^*(t) = h(t)$.

(I') If $t = f(t_1, \ldots, t_n)$, $h^*(t) = h_f(h^*(t_1), \ldots, h^*(t_n))$.

If $t$ is a term, the element $h^*(t)$ of $V$ will be called the interpretation of $t$ by $h^*$.

Proof. By structural induction (or induction on the construction of terms). Let $P(t)$ be the property: 'there exists a unique $y = h^*(t)$ verifying (B') and (I')'.

(B) $P(t)$ is true if $t = f \in F_0$ because $y = h(t)$ by (B').

(I) If $P(t_1), \ldots, P(t_n)$ are true, and if $t = f(t_1, \ldots, t_n)$, then $P(t)$ is true because

- on the one hand, there is a unique way of decomposing $t$ in the form $f(t_1, \ldots, t_n)$: if $f(t_1, \ldots, t_n) = g(t'_1, \ldots, t'_p)$ then $f = g$, $n = p$, and $t_i = t'_i$, $\forall i = 1, \ldots, n$,
- on the other hand, by (I') if $P(t_1), \ldots, P(t_n)$ are true then $P(t)$ must be true because $h^*(t)$ is entirely defined by $h^*(t) = h_f(h^*(t_1), \ldots, h^*(t_n))$.

Another proof will be given in Section 3.4.
EXAMPLE 3.16 Let $F_0 = \{a\}$, $F_1 = \{s\}$, $F = F_0 \cup F_1$. We have

$$T = \{a, s(a), s(s(a)), \ldots\}.$$ 

Let $V = \mathbb{N}$.

If $h_1(a) = 0$ and $h_1(s(n)) = n + 1$, then

$$h_1^*(s^n(a)) = h_1^*(s\underbrace{s\ldots (s(a)) \ldots}_{n \text{ times}}) = n.$$ 

If $h_2(a) = 1$ and $h_2(s(n)) = 2n$, then

$$h_2^*(s^n(a)) = h_2^*(s\underbrace{s\ldots (s(a)) \ldots}_{n \text{ times}}) = 2^n.$$ 

If $h_3(a) = 1$ and $h_3(s(n)) = n + 2$, then

$$h_3^*(s^n(a)) = h_3^*(s\underbrace{s\ldots (s(a)) \ldots}_{n \text{ times}}) = 2n + 1.$$ 

Indeed, we verify by induction that:

1. $h_1(a) = 0$ and $h_1^*(s^{n+1}(a)) = h_1^*(s(s^n(a))) = h_1^*(h_1^*(s^n(a))) = h_1(s(n)) = n + 1$,
2. $h_2(a) = 1$ and $h_2^*(s^{n+1}(a)) = h_2^*(s(s^n(a))) = 2 \times h_2^*(s^n(a)) = 2 \times 2^n = 2^{n+1}$,
3. $h_3(a) = 1$ and $h_3^*(s^{n+1}(a)) = h_3^*(s(s^n(a))) = h_3^*(s^n(a)) + 2 = 2n + 1 + 2 = 2(n + 1) + 1$.

Let $E$ be an arbitrary set, and let $X$ be the subset of $E$ inductively defined by the conditions (B) and (I). Theorem 3.10 asserts that each element of $X$ is obtained from the basis by applying a finite number of inductive steps. We refine this result by describing it by an element $x$ how the element $x$ is obtained.

With each element $b$ of the basis $B$, we associate a nullary symbol denoted by $b$.

With each function $\Phi$ of $K$, we associate the arity $\alpha(\Phi)$ symbol $\Phi$. Let $T$ be the set of all terms constructed with these symbols.

We consider the interpretation $h^* : T \to E$ defined by

- $h^*(b) = b$,
- $h^*(x_1, \ldots, x_{\alpha(\Phi)}) = \Phi(x_1, \ldots, x_{\alpha(\Phi)})$.

Proposition 3.17 $X = \{h^*(t) \mid t \in T\}$.

Proof. For an element $x$ of $E$, let $P(x)$ be the property: "there exists a term $t$ such that $x = h^*(t)$." It is easy to see that $P$ has properties (B') and (I') of Proposition 3.11, and thus $X \subseteq h^*(T)$.

For a term $t$ of $T$, let $Q(t)$ be the property: "$h^*(t) \in X$". Here also Proposition 3.11 enables us to conclude that $h^*(T) \subseteq X$. 

$\square$
3.3.3 Unambiguous definitions

Definition 3.18 An inductive definition of a set \( X \) is said to be unambiguous if the mapping \( h^* \) of Proposition 3.17 is injective, i.e. for any \( x \in X \) there exists a unique term \( t \) such that \( x = h^*(t) \).

More intuitively, this means that there is a unique way of building up an element \( x \) of \( X \).

Example 3.19 The following definition of \( \mathbb{N}^2 \) is ambiguous:

\[ \begin{align*}
(B) & \quad (0,0) \in \mathbb{N}^2, \\
(I_1) & \quad (n,m) \in \mathbb{N}^2 \implies (n+1,m) \in \mathbb{N}^2, \\
(I_2) & \quad (n,m) \in \mathbb{N}^2 \implies (n,m+1) \in \mathbb{N}^2.
\end{align*} \]

Indeed, the pair \((1,1)\) can be obtained from \((0,0)\) by using the rule \((I_1)\) first then the rule \((I_2)\), or by using the rule \((I_2)\) first then the rule \((I_1)\).

More formally, we consider the terms built up from

- the arity 0 symbol \( \overline{b} \) whose interpretation \( h(\overline{b}) \) is \((0,0)\),
- the unary symbols \( \overline{f} \) and \( \overline{g} \) whose interpretations are defined by
  \[ \begin{align*}
  (i) & \quad h_\overline{f}(n,m) = (n+1,m), \\
  (ii) & \quad h_\overline{g}(n,m) = (n,m+1).
  \end{align*} \]

Then \((1,1) = h^*(\overline{f}(\overline{g}(\overline{b}))) = h^*(\overline{g}(\overline{f}(\overline{b})))\).

3.3.4 Inductively defined functions

In order to define a function on an inductively defined set unambiguously, it is convenient to use an inductive definition. Intuitively, we define the function on the elements of the basis directly, and then define new elements inductively, building them up from elements already defined.

Definition 3.20 Let \( X \subseteq E \) be an unambiguous inductively defined set (see Definitions 3.7 and 3.18), and let \( F \) be any set. The inductive definition of mapping \( \psi \) from \( X \) to \( F \) consists of

\[ \begin{align*}
(B) & \quad \text{specifying } \psi(x) \in F \text{ for each element } x \in B, \\
(I) & \quad \text{specifying the expression of } \psi(\Phi(x_1, \ldots, x_\alpha(\Phi))) \text{ in terms of } x_1, \ldots, x_\alpha(\Phi) \text{ and of } \psi(x_1), \ldots, \psi(x_\alpha(\Phi)) \text{ for each } \Phi \in K. \text{ We will write}
\end{align*} \]

\[ \psi(\Phi(x_1, \ldots, x_\alpha(\Phi))) = \psi_\Phi(x_1, \ldots, x_\alpha(\Phi), \psi(x_1), \ldots, \psi(x_\alpha(\Phi))), \]

where \( \psi_\Phi \) is a mapping from \( E^{\alpha(\Phi)} \times F^{\alpha(\Phi)} \) to \( F \).

The definition is illustrated by the following examples.
EXAMPLE 3.21 The factorial function from \( \mathbb{N} \) to \( \mathbb{N} \) is defined inductively by

(B) \( \text{Fact}(0) = 1 \),

(I) \( \text{Fact}(n + 1) = (n + 1) \times \text{Fact}(n) \).

Here we use the inductive definition of \( \mathbb{N} \) given in Example 3.9. First, the factorial function for the unique element of the basis \( \langle 0 \rangle \) is defined directly, and then the factorial applied to the new element \( n + 1 \) is expressed in terms of \( n \) and \( \text{Fact}(n) \).

Henceforth, we will also write inductive definitions of functions as follows:

\[
\text{Fact}(n) = \begin{cases} 
1 & \text{if } n = 0, \\
1 \times \text{Fact}(n - 1) & \text{otherwise.}
\end{cases}
\]

EXERCISE 3.18 Let \( A^* \) be the free monoid on the alphabet \( A \) (see Definition 1.15). The mirror image (or reverse) of a string \( u = a_1 a_2 \cdots a_n \) is the string \( \overline{u} = a_n \cdots a_2 a_1 \). Give an inductive definition of the mirror image.

EXERCISE 3.19 Let the lists \( L \) of letters from the alphabet \( A \) be defined inductively by:

(B) \( \varepsilon \in L \),

(I) \( \forall l \in L, \forall a \in A, (al) \in L \).

We define \( g(x, y) \) on \( L \times L \) by, \( \forall a \in A, \forall l \in L, \forall y \in L \),

\[
\begin{align*}
g(\varepsilon, y) &= y, \\
g(al, y) &= g(l, ay).
\end{align*}
\]

1. Let \( Q(x) \) be the predicate \( \langle x, y \rangle \) is defined. Prove by induction on \( x \) that \( Q(x) \) holds on \( L \).
2. Compute \( g((a_1), y) \), for \( a_1 \in A, y \in L \).
3. Prove by induction on \( n \) (for \( n \geq 1 \)) that \( g((a_1(a_2 \cdots (a_n \cdots) \cdots), y) = g(\varepsilon,\)

\( a_1(a_2 \cdots (a_n \cdots) \cdots)) \).
4. Let \( \text{rev}(x) = g(x, \varepsilon) \). Deduce from 3 that, for \( a_1, \ldots, a_n \in A, \)

\( \text{rev}(a_1(a_2 \cdots (a_n \cdots)) = (a_n \cdots (a_2 \cdots a_1)) \).

We now justify Definition 3.20 and explain why we have assumed the definition of the set \( X \) to be unambiguous.

Instead of defining a function \( \psi \) from \( X \) to \( F \), we will define a function \( \psi' \) from \( T \) to \( F \), where \( T \) is the set of terms whose interpretation is in \( X \) (see Proposition 3.17). \( \psi' \) is defined as follows:

- \( \psi'(b) = \psi(b) \),
- \( \psi'(\Psi(t_1, \ldots, t_n)) = \psi_h(h_1(t_1), \ldots, h_n(t_n)) \).

As in the proof of Proposition 3.15 we show that such a function exists and is unique.
If the inductive definition of $X$ is unambiguous, then for each element $x$ of $X$ there exists a unique term $t$ such that $h^*(t) = x$. Then let $\psi(x) = \psi'(t)$. $\psi$ is thus indeed a mapping from $X$ to $P$ and it is easy to prove that $\psi$ verifies the conditions (B) and (I) of Definition 3.20.

If the definition of $X$ is ambiguous, then there exist several terms $t_1, \ldots, t_n$ whose interpretation is the same element $x$ of $X$ and, according to the chosen term, Definition 3.20 will give different values $\psi(t_1), \ldots, \psi(t_n)$ to $\psi(x)$. This is illustrated by the following example.

**Example 3.22** Let us consider the following inductive definition of $\psi$ from $\mathbb{N}^2$ to $\mathbb{N}$, where the inductive definition of $\mathbb{N}^2$ is given in Example 3.19:

(B) $\psi(0,0) = 1$,

(I') $\psi(n+1,m) = \psi(n,m)^2$,

(I'') $\psi(n,m+1) = 3 \times \psi(n,m)$.

The thus defined $\psi$ is not a mapping because by using the rule (I') first and then the rule (I''), we obtain $\psi(1,1) = \psi(0,1)^2 = (3 \times \psi(0,0))^2 = 9^2 = 9$, whilst by using the rule (I'') first and then the rule (I'), we obtain

$$\psi(1,1) = 3 \times \psi(1,0) = 3 \times \psi(0,0)^2 = 3.$$

More generally, we can consider that Definition 3.20 in fact defines a relation $\mathcal{R}$ from $X$ to $P$ by: $x \mathcal{R} y$ if and only if there exists a term $t$ such that $x = h^*(t)$ and $y = \psi(t)$. If $h^*$ is injective then this relation is functional, as we just saw. We should, however, note that this is not the only case when $\mathcal{R}$ is functional. In fact $\mathcal{R}$ is functional if and only if

$$\forall t, t' \in T, \quad h^*(t) = h^*(t') \implies \psi(t) = \psi(t').$$

**Example 3.23** We consider again the ambiguous definition of $\mathbb{N}^2$ given in Example 3.19 and we consider the inductive definition

(B) $g(0,0) = 1$,

(I_1) $g(n+1,m) = 2 \times g(n,m)$,

(I_2) $g(n,m+1) = 3 \times g(n,m)$.

Using Proposition 3.11, we easily show by induction that there exists a unique mapping $g$ verifying these conditions and that this unique mapping is defined by $\forall (n,m) \in \mathbb{N}^2$, $g(n,m) = 2^n 3^m$.

**Exercise 3.20** Let $\mathbb{N}^* = \mathbb{N} \setminus \{0\}$. We give an inductive definition of the function 'modulo' defined on $\mathbb{N} \times \mathbb{N}^*$, that, when applied to the pair $(n,m)$, gives the remainder of the Euclidean division of $n$ by $m$

$$n \mod m = \begin{cases} n & \text{if } n < m, \\ (n - m) \mod m & \text{otherwise.} \end{cases}$$
51

the corresponding unambiguous inductive definition of \( \mathbb{N} \times \mathbb{N}^* \).

\[ \diamond \]

**Example 3.24**

1. The expressions of the set \( B \) (see Example 3.9) use an infix notation (in which the operator is placed between its arguments). We can also use a postfix notation without parentheses (in which the operator is placed after both its arguments). For instance, the postfix notation of expression

\[ \left( a \times (b + c) + d \right) \]

is \( abc + d \). The transformation from the infix notation to the postfix notation is inductively defined by

- \( B \) : \( \forall a \in A, \text{Post}(a) = a \),
- \( I \) : \( \forall c, f \in B, \text{Post}(c + f) = \text{Post}(c) \text{Post}(f) + \) and \( \text{Post}(c \times f) = \text{Post}(c) \text{Post}(f) \times \).

2. The height of a binary tree is inductively defined by

- \( B \) : \( h(\emptyset) = 0 \),
- \( I \) : \( \forall t, r \in BT, \forall a \in A, h((a, l, r)) = 1 + \max(h(l), h(r)) \).

A more elegant definition of this function is

\[ h(x) = \begin{cases} 0 & \text{if } x = \emptyset, \\ 1 + \max(h(l), h(r)) & \text{if } x = (a, l, r). \end{cases} \]

3. The inorder traversal of a tree is the list of the labels of its nodes from left to right. We can notice that several trees may have the same inorder traversal. For instance, the two trees of Figure 3.5 have the same inorder traversal \( bacad \). The inductive definition of the inorder traversal is

\[ \text{Inf}(x) = \begin{cases} e & \text{if } x = \emptyset, \\ \text{Inf}(l) \cdot a \cdot \text{Inf}(r) & \text{if } x = (a, l, r). \end{cases} \]

**Figure 3.5**
EXERCISE 3.22 Give inductive definitions of the functions $n$ and $l$ from $BT'$ to $\mathbb{N}$, defining respectively, the number of nodes and the number of leaves of a binary tree. For instance, if $x$ is either tree in Figure 3.5, we have $n(x) = 5$ and $l(x) = 3$.

EXERCISE 3.23 Define the preorder traversal of a binary tree. The preorder traversals of the trees of Figure 3.5 are abcd and abcd.

Note that inductive definitions are appropriate as definitions of certain algorithms: sorting algorithms, algorithms on trees such as binary search, insertion, traversal, etc.

3.4 Closure operations

In the proof of Theorem 3.10, we showed that the set $X$ inductively defined by

(2) $B \subseteq X$, and

(1) $\forall \Phi \in K, \forall x_1, \ldots, x_{a(\Phi)} \in X, \Phi(x_1, \ldots, x_{a(\Phi)}) \in X$,

is the union of the sets $X_n$ with $X_0 = B$ and $X_{n+1} = X_n \cup \{ \Phi(x_1, \ldots, x_{a(\Phi)}) : x_1, \ldots, x_{a(\Phi)} \in X_n \text{ and } \Phi \in K \}$. We see that the subset (1) of the inductive definition of $X$ is used in order to build a new set $X_{n+1}$ from an already known set $X_n$. Indeed, it suffices to define $X_{n+1}$ (or even $X_{n+1} - X_n$) from $X_n$ and, when no new element can be added, the inductive definition is completed.

More generally, we will assume that from any given set $E$ we can build a new set $C(E)$. We will study the properties that $C$ should have in order to give an inductive definition which will be completed whenever $C$ can add no new element to $E$. With this standpoint we will generalize the results of Section 3.2.

Let $U$ be any set and let $C : \mathcal{P}(U) \rightarrow \mathcal{P}(U)$ be a monotone mapping, i.e., a mapping verifying $\forall E, E' \subseteq U, E \subseteq E' \rightarrow C(E) \subseteq C(E')$.

A subset $E$ of $U$ is said to be $C$-closed if $C(E) \subseteq E$.

Proposition 3.25 Let $I$ be any set of indices. Let $E_i$ be a $C$-closed set, for any $i \in I$. Then $\bigcap_{i \in I} E_i$ is $C$-closed.

Proof. Let $E = \bigcap_{i \in I} E_i$. Since $E \subseteq E_i$ and $E_i$ is $C$-closed, $C(E) \subseteq C(E_i) \subseteq E_i$; hence $C(E) \subseteq \bigcap_{i \in I} E_i = E$.

If $E$ is any subset of $U$, the intersection of all the $C$-closed subsets of $U$ containing $E$ is a $C$-closed subset of $U$ containing $E$, denoted by $\hat{C}(E)$.
Closure operations

Proposition 3.26
- If $E'$ is a $C$-closed subset containing $E$, then $\hat{C}(E) \subseteq E'$,
- $E \subseteq \hat{C}(E)$,
- $\hat{C}(\hat{C}(E)) = \hat{C}(E)$,
- $E \subseteq E' \implies \hat{C}(E) \subseteq \hat{C}(E')$.

Proof. The first two points are clear by the definition of $\hat{C}(E)$.

- On the one hand, we have that $E \subseteq \hat{C}(E) \subseteq \hat{C}(\hat{C}(E))$. And, on the other hand, since $\hat{C}(E)$ is a $C$-closed subset containing $\hat{C}(E)$, then $\hat{C}(\hat{C}(E)) \subseteq \hat{C}(E)$.
- If $E \subseteq E'$, then $E \subseteq \hat{C}(E')$ which is a $C$-closed subset containing $E$. We thus have that $\hat{C}(E) \subseteq \hat{C}(E')$.

The next proposition is a generalization of the induction principles and may be called the universal induction principle.

Proposition 3.27 Let $P \subseteq U$ be such that $C(P) \subseteq P$. Then

$$\forall E, \quad E \subseteq P \implies \hat{C}(E) \subseteq P.$$ 

Proof. If $C(P) \subseteq P$, then $P$ is $C$-closed. So if $E \subseteq P$, then $\hat{C}(E) \subseteq P$.

Example 3.28 Let $X$ be a subset of a set $U$; assume that $X$ is inductively defined by (B) and (I) (see Definition 3.7). Define $C: P(U) \rightarrow P(U)$ by

$$C(Y) = \{ \Phi(y_1, \ldots, y_{\Phi}) \mid \Phi \in K, y_1, \ldots, y_{\Phi} \in Y \}.$$ 

Then $X = \hat{C}(B)$.

Example 3.29 Let $U = \mathbb{N}$, $C(E) = \{n + 1 \mid n \in E\}$. Then

$$E' = \hat{C}(E) = \{n + m \mid n \in E, m \in \mathbb{N}\}.$$ 

Indeed, $C(E') = \{n + m + 1 \mid n \in E', m \in \mathbb{N}\} \subseteq E'$. Assume there is a $C$-closed subset $E''$ containing $E$ and strictly included in $E'$. Let $k$ be the least integer of $E'$ that is not in $E''$, i.e. $k \in E' \setminus E''$ and $(k = 0$ or $k - 1 \in E''$).

- If $k = 0$ then, since $k = n + m$ with $n \in E$, $0 \in E$, and hence $0 \in E''$, a contradiction.
- Otherwise, $k - 1 \in E'' \implies k = (k - 1) + 1 \in C(E'') \subseteq E''$, a contradiction.

We deduce: $\hat{C}(E) = \{m \mid m \geq \inf(E)\} = \hat{C}(\{\inf(E)\})$. Let $P$ be such that $n \in P \implies n + 1 \in P$. Then $C(P) \subseteq P$, and hence $\inf(E) \in P \implies \hat{C}(E) \subseteq P$. 


If \( \inf(E) = 0 \) then \( \hat{C}(E) = \mathbb{N} \), and we again find the induction principle on the integers.

**Exercise 3.24** Let \( U = \mathbb{N} \) and \( C(E) = \{ \frac{n + m}{n} \mid n, m \in E \} \). Let \( k\mathbb{N} = \{ kn \mid n \in \mathbb{N} \} \). Show that if \( E \subseteq k\mathbb{N} \) then \( \hat{C}(E) \subseteq k\mathbb{N} \).

Let \( C : \mathcal{P}(U) \rightarrow \mathcal{P}(U) \) be such that \( E \subseteq E' \implies C(E) \subseteq C(E') \). \( C \) is finitary if it also verifies: \( \forall E, \forall e \in C(E) \), there exists a finite subset \( F \) of \( E \) such that \( e \in C(F) \).

Let \( E \subseteq U \). Consider the monotone increasing (for inclusion) sequence

\[
\begin{align*}
E_0 &= E \\
E_1 &= E_0 \cup C(E_0) \\
& \vdots \\
E_{i+1} &= E_i \cup C(E_i) \\
& \vdots \\
\hat{E} &= \bigcup_{i \geq 0} E_i.
\end{align*}
\]

**Proposition 3.30** \( \hat{E} \subseteq \hat{C}(E) \). If \( C \) is finitary, \( \hat{E} = \hat{C}(E) \).

**Proof.** Let \( E' = \hat{C}(E) \). We show by induction on the integers that \( \forall i \geq 0, E_i \subseteq E' \).

- \( E_0 = E \subseteq E' \).
- We assume \( E_i \subseteq E' \). Then \( C(E_i) \subseteq C(E') \subseteq E' \) and \( E_{i+1} = E_i \cup C(E_i) \subseteq E' \). Since \( \forall i \geq 0, E_i \subseteq E' \), we have \( \hat{E} = \bigcup_{i \geq 0} E_i \subseteq E' \).

We show that if \( C \) is finitary then \( \hat{E} \) is \( C \)-closed, and we will therefore deduce that \( E' \subseteq \hat{E} \). Let \( e \in C(\hat{E}) \). Because \( C \) is finitary, there exists a finite subset \( F = \{ x_1, \ldots, x_p \} \) of \( \hat{E} \) such that \( e \in C(F) \). Since \( x_j \in \bigcup_{i \geq 0} E_i \), there exists \( i_j \) such that \( x_j \in E_{i_j} \); let \( k = \max\{ i_j \mid j = 1, \ldots, p \} \). We thus have that \( F \subseteq E_k \) and \( e \in C(F) \subseteq C(E_k) \subseteq E_{k+1} \subseteq \hat{E} \). Hence \( C(\hat{E}) \subseteq \hat{E} \).

**Example 3.31** The mapping \( C \) that inductively defines a set \( X \) (see Example 3.28) is finitary, whence Theorem 3.10.
Closure operations

Example 3.32 The mapping $C$ from $P(\mathbb{R})$ to itself, which is defined by $y \in C(X)$ if and only if there exists $Y \subseteq X$ such that $y = \inf Y$, is not finitary. Indeed, let $X = \{1/n \mid n \in \mathbb{N}, n > 0\}$. We thus have $0 \in C(X)$. But for all finite subsets $F$ of $X$, $0 \notin C(F)$ because the greatest lower bound of any finite subset of $X$ is of the form $1/n$ for some $n > 0$.

Exercise 3.25 Let $E$ be a vector space on $\mathbb{R}$. For $a, b \in E$, let
\[ [a, b] = \{\lambda a + \mu b / \lambda \geq 0, \mu \geq 0, \text{ and } \lambda + \mu = 1\} \]
be the closed segment subtended by $a$ and $b$. Let $C : P(E) \rightarrow P(E)$ be defined by
\[ C(X) = \bigcup_{a, b \in X} [a, b]. \]

What usual name is given to $C(X)$?

1. Is $C$ monotone increasing and finitary?
2. Given $a \in \dot{C}(A)$ and $b \in \dot{C}(B)$, show that $[a, b] \subseteq \dot{C}(A \cup B)$.
3. Deduce that $\bigcup_{F \in \mathrm{fin}(X)} \dot{C}(F)$ is $C$-closed, where $\mathrm{fin}(X)$ is the set of finite subsets of $X$.
4. Is $C$ monotone increasing and finitary?
5. Can you generalize (4) to any set transformation which is monotone increasing and finitary?

We can apply the closure operations in order to define the terms. Let $U$, $F$, and $F_0$ be as in Section 3.3. Let $C : P(U) \rightarrow P(U)$ be defined by
\[ C(F) = \bigcup_{i > 0} \{f(\sigma_1, \ldots, \sigma_i) / \sigma_j \in F_i, f \in F_0\}. \]

Then $C$ is finitary and the set $T$ of terms built on $P$ is identical to $\dot{C}(F_0)$.

Exercise 3.26 Let $C' = C(F) \cup F_0$. Show that $\dot{C}(F_0) = C'(0)$.

Exercise 3.27 Let $\dot{E} = \dot{C}(F_0)$. Show that $T = \dot{E}$.

Exercise 3.28 Show that there exists a unique function $h^*$ verifying conditions (B') and (1') of Proposition 3.15.

Exercise 3.29 Let $U = \mathbb{N}$ and let $C : P(\mathbb{N}) \rightarrow P(\mathbb{N})$ be defined by
\[ C(X) = \begin{cases} \{x + 1 / x \in X\} & \text{if } X \text{ is finite}, \\ \{x + 1 / x \in X\} \cup \{0\} & \text{if } X \text{ is infinite}. \end{cases} \]

1. Show that the limit of the sequence
\[ \begin{align*} E_0 &= \{1\} \\ E_1 &= E_0 \cup C(E_0) \\ & \vdots \\ E_{i+1} &= E_i \cup C(E_i) \end{align*} \]
is equal to \( \mathbb{N} \setminus \{0\} \).
2. Show that \( C(\{1\}) = \mathbb{N} \).
3. Explain this result.