# Contents


**I Foundations of Formal Number Systems** **1**

**1 Identity, Equality, and Equivalence** **2**
1.1 Identity, Equality, and Equivalence . . . 2
1.2 Equality . . . 2
1.3 Properties of Equality . . . 3
1.4 Substitution . . . 5
1.5 Equivalence . . . 8

**2 Arithmetic Operations and Relations** **16**
2.1 Arithmetic Operations and Relations . . . 16
2.2 Closure . . . 16
2.3 Algebraic Laws . . . 19
2.4 Identity Elements . . . 22
2.5 Inverse Operations . . . 23
2.6 Algebraic Structures . . . 25
2.7 Derived Laws . . . 29
2.8 Order Relations . . . 31
2.9 Composed Relations . . . 33
2.10 Order Compatibility . . . 34
2.11 Structure-Preserving Maps . . . 37
2.12 Field and Ordered-Field Laws . . . 38

**3 Peano Systems** **44**
3.1 Presburger Arithmetic . . . 44
3.2 Defining Peano Systems . . . 45
3.3 Examples of Peano Systems . . . 66
3.4 Recursion on Peano Systems . . . 70

**4 Natural Numbers ($\mathbb{N}$)** **91**
4.1 Axioms for $\mathbb{N}$ . . . 91
4.2 Canonical Natural Number System . . . 92
4.3 Addition on N . . . 93
4.4 Multiplication on N . . . 100
4.5 Order on N . . . 108
4.6 Powers and Growth . . . 116
4.7 Well Ordering Principle . . . 121
4.8 Induction . . . 124
4.9 Algebraic Structure of N . . . 124

**5 Constructing the Whole Numbers** **128**
5.1 Why Whole Numbers Are Introduced . . . 128
5.2 Definition of the Whole Numbers . . . 129
5.3 Extending Successor . . . 130
5.4 Extending Addition to $\mathbb{W}$ . . . 132
5.5 Extending Multiplication to $\mathbb{W}$ . . . 135
5.6 Order on $\mathbb{W}$ . . . 139
5.7 Algebraic Structure of $\mathbb{W}$ . . . 143
5.8 Transition to $\mathbb{Z}$ . . . 146

**6 Integers ($\mathbb{Z}$)** **147**
6.1 Construction of $\mathbb{Z}$ . . . 147
6.2 Operations on $\mathbb{Z}$-Classes . . . 150
6.3 Algebraic Structure of $\mathbb{Z}$ . . . 159
6.4 Order on $\mathbb{Z}$ . . . 161
6.5 Embedding $\mathbb{W}$ into $\mathbb{Z}$ . . . 172
6.6 Divisibility in $\mathbb{Z}$ . . . 174
6.7 Transition to $\mathbb{Q}$ . . . 178

**7 Rational Numbers ($\mathbb{Q}$)** **179**
7.1 Construction of $\mathbb{Q}$ . . . 179
7.2 Operations on $\mathbb{Q}$-Classes . . . 183
7.3 Field Structure of Q . . . 195
7.4 Order on $\mathbb{Q}$ . . . 199
7.5 Ordered Field Structure of Q . . . 209
7.6 Arbitrary Closeness in $\mathbb{Q}$ . . . 211
7.7 Rational Gaps . . . 214
7.8 Embedding $\mathbb{Z}$ into $\mathbb{Q}$ . . . 216
7.9 Transition to $\mathbb{R}$ . . . 219
7.10 Canonical Rational Number Statement Plan . . . 219

**8 Real Numbers ($\mathbb{R}$)** **226**
8.1 Construction of $\mathbb{R}$ . . . 226
8.2 Order on $\mathbb{R}$ . . . 228
8.3 Completeness of $\mathbb{R}$ . . . 230
8.4 Addition on $\mathbb{R}$ . . . 232
8.5 Multiplication on $\mathbb{R}$ . . . 234
8.6 Field and Ordered-Field Structure of $\mathbb{R}$ . . . 239
8.7 Embedding $\mathbb{Q}$ into $\mathbb{R}$ . . . 241
8.8 The Gap Filled . . . 243
8.9 Characterization of $\mathbb{R}$ . . . 244
8.10 Transition and Forward . . . 245

**9 Summary: The Number Systems** **246**
9.1 Axiom Systems for $\mathbb{N}$, $\mathbb{W}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ . . . 246

**10 Embedding the Number Systems** **250**
10.1 Embeddings as Structure Preserving Maps . . . 250
10.2 Embedding N Into W . . . 253
10.3 Embedding W Into Z . . . 254
10.4 Embedding Z Into Q . . . 255
10.5 Embedding Q Into R . . . 256
10.6 Compatibility of Arithmetic and Order . . . 257
10.7 One Chain Many Structures . . . 258

**11 Number Lines and Intervals** **259**
11.1 Number Lines . . . 259
11.2 Intervals on a Number Line . . . 260

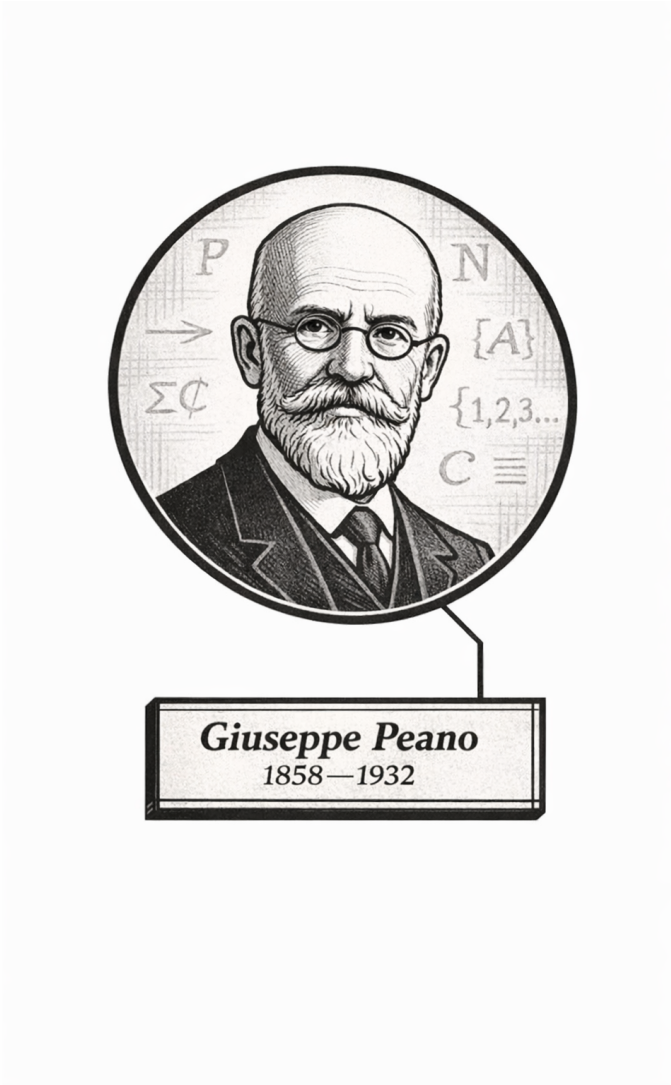
**12 Structure of the Real Line** **262**
12.1 Real Line One Dimensional . . . 262
12.2 Order Distance Length . . . 262
12.3 Absolute Value as Distance . . . 263
12.4 Intervals as Subsets . . . 264
12.5 Neighborhoods and Balls . . . 267
12.6 Open Sets . . . 268
12.7 Closed Sets . . . 269
12.8 Interior Exterior Boundary . . . 270
12.9 Limit and Isolated Points . . . 271
12.10 Unions and Intersections . . . 272
12.11 Covers and Subcovers . . . 273
12.12 Compactness on R . . . 274
12.13 Heine Borel . . . 275

**13 Formalizing the Number Systems** **277**

**14 Lean 4 Formalization** **279**
14.1 A First Lean 4 Reading . . . 279

# Volume II

Origins of the Numbers

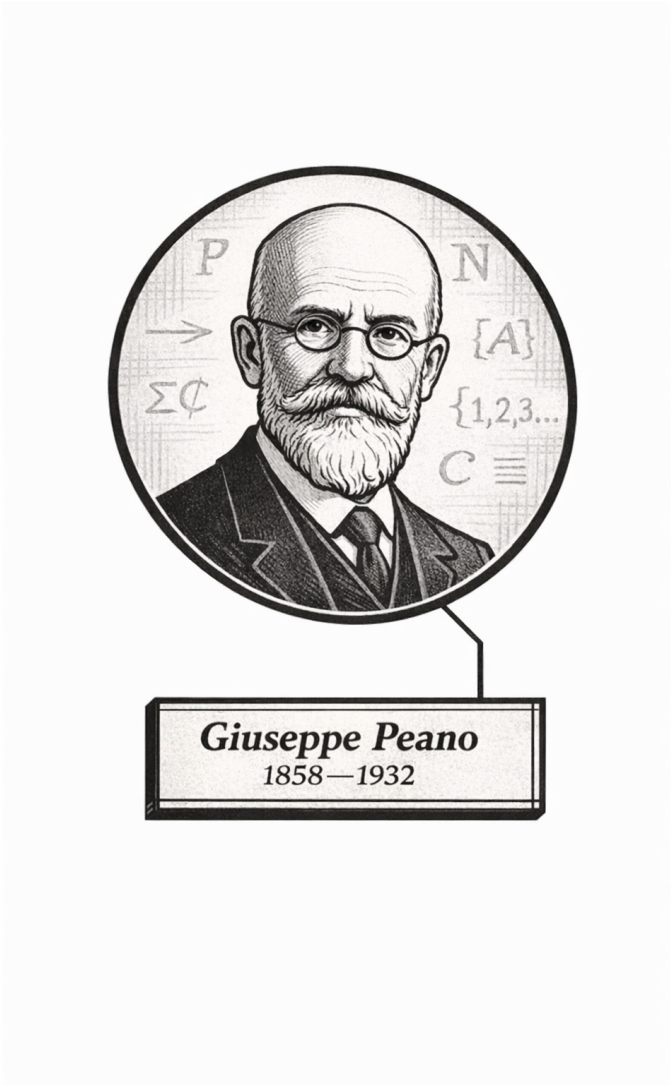



*One, ah-ah-ah... Two, ah-ah-ah...*

Guiseppe Peano (probably)

# Volume II

Origins of the Numbers




*One, ah-ah-ah... Two, ah-ah-ah...*

Guiseppe Peano (probably)

# Volume II

## Origins of the Numbers

*Self-Study Notes*

William Sollers

June 19, 2026

# Part I

# Foundations of Formal Number Systems

# Chapter 1

# Identity, Equality, and Equivalence

**identity-equality-equivalence**

Peano Systems $\rightarrow$ **Identity, Equality, and Equivalence** $\rightarrow$ Arithmetic Operations and Relations

## 1.1 Identity, Equality, and Equivalence

## 1.2 Equality

**Definition (Identity)**

**Definition 1.2.1** (Identity). Two expressions $x$ and $y$ are *identical* when they denote the same object.

*Remark* (Standard quantified statement).

$$x = y \iff x \text{ and } y \text{ denote the same object.}$$

*Remark* (Interpretation). Identity is sameness, not similarity. It is the relation used when no distinction remains between two denotations.

*Remark* (Dependencies).

- Relation

**Axiom (Leibniz's Law)**

**Axiom 1.2.2** (Leibniz's Law). Identical objects have exactly the same properties.

$$\forall x \, \forall y \left( x = y \implies \forall P \left( P(x) \iff P(y) \right) \right).$$

*Remark* (Standard quantified statement).

$$\text{Indiscernible}(x, y) \iff \forall P \left( P(x) \iff P(y) \right).$$

*Remark* (Interpretation). Leibniz's law says that equality permits no observable mathematical difference. Any property true of one equal object is true of the other.

**Definition (Equality on a Domain)**

**Definition 1.2.3** (Equality on a Domain). Let $A$ be a domain. *Equality on $A$* is the relation, written $=$, that holds exactly when two terms or constructions denote the same element of $A$.

*Remark* (Standard quantified statement).

$$\text{Equality}_A(=) \iff \forall x, y \in A,\ \big(x = y \iff x \text{ and } y \text{ denote the same element of } A\big).$$

*Remark* (Interpretation). Equality is the identity relation restricted to the domain currently under discussion. Later equivalence relations may identify objects for a purpose, but equality records literal sameness.

*Remark* (Dependencies).

- Relation

**Definition (Definitional and Propositional Equality)**

**Definition 1.2.4** (Definitional and Propositional Equality). *Definitional equality* records an introduced meaning, written with $:=$. *Propositional equality* is a statement of sameness inside the theory, written with $=$.

*Remark* (Standard quantified statement).

$$a := b \quad \text{introduces notation, while} \quad a = b \quad \text{asserts equality.}$$

*Remark* (Interpretation). The distinction prevents definitions from being confused with theorems. Definitions create abbreviations; propositional equalities must be available as axioms, assumptions, or proved statements.

## 1.3 Properties of Equality

**Axiom (E1, Reflexivity of Equality)**

**Axiom 1.3.1** (Reflexivity of Equality).

$$\forall x,\ \ x = x.$$

*Remark* (Standard quantified statement).

$$\text{Reflexive}(=) \iff \forall x,\ \ x = x.$$

*Remark* (Interpretation). Every object is itself. This is the primitive equality axiom used before symmetry, transitivity, or arithmetic structure is derived.

**Theorem (E2, Symmetry of Equality)**

**Theorem 1.3.2** (Symmetry of Equality).

$$\forall x\, \forall y,\ \ (x = y \Longrightarrow y = x).$$

*Remark* (Standard quantified statement).

$$\text{Symmetric}(=) \Longleftrightarrow \forall x\, \forall y,\ \ (x = y \Longrightarrow y = x).$$

*Remark* (Interpretation). If $x$ is the same object as $y$, then $y$ is the same object as $x$. Symmetry follows from Leibniz's law by substituting into the property $P(t) \equiv (t = x)$ and using reflexivity.

*Remark* (Dependencies).

- Leibniz's Law
- Reflexivity of Equality

**Theorem (E3, Transitivity of Equality)**

**Theorem 1.3.3** (Transitivity of Equality).

$$\forall x\, \forall y\, \forall z,\ \big(x = y \wedge y = z \Longrightarrow x = z\big).$$

*Remark* (Standard quantified statement).

$$\text{Transitive}(=) \Longleftrightarrow \forall x\, \forall y\, \forall z,\ \big(x = y \wedge y = z \Longrightarrow x = z\big).$$

*Remark* (Interpretation). Sameness chains: if the first object is the second and the second is the third, then the first is the third.

*Remark* (Dependencies).

- Leibniz's Law
- Reflexivity of Equality
- Symmetry of Equality

**Corollary (Equality Is an Equivalence Relation)**

**Corollary 1.3.4** (Equality Is an Equivalence Relation). *The equality relation is reflexive, symmetric, and transitive.*

*Remark* (Standard quantified statement).

$$\text{EquivalenceRelation}(=) \Longleftrightarrow \text{Reflexive}(=) \wedge \text{Symmetric}(=) \wedge \text{Transitive}(=).$$

*Remark* (Interpretation). Equality is the strictest equivalence relation: it groups only objects that are literally the same.

*Remark* (Dependencies).

- Reflexivity of Equality
- Symmetry of Equality
- Transitivity of Equality

## 1.4 Substitution

**Axiom (Substitution of Equals)**

**Axiom 1.4.1** (Substitution of Equals). For every formula $\varphi(z)$ whose substitution is admissible,

$$x = y \implies \big(\varphi(x) \iff \varphi(y)\big).$$

***Remark*** (Standard quantified statement).

$$\forall x\, \forall y \Big(x = y \implies \big(\varphi(x) \iff \varphi(y)\big)\Big).$$

***Remark*** (Interpretation). Substitution is the operational meaning of equality. Equal objects may replace one another in legitimate statements without changing truth.

**Proposition (Substitution Preserves Predicates)**

**Proposition 1.4.2** (Substitution Preserves Predicates).

$$x = y \implies \big(P(x) \iff P(y)\big).$$

***Remark*** (Standard quantified statement).

$$\forall x\, \forall y,\ x = y \implies \big(P(x) \iff P(y)\big).$$

***Remark*** (Interpretation). Predicates cannot distinguish equal arguments.

*Remark* (Dependencies).

- Substitution of Equals

**Proposition (Substitution Preserves Relations on the Left)**

**Proposition 1.4.3** (Substitution Preserves Relations on the Left).

$$x = y \implies \big(R(x, z) \iff R(y, z)\big).$$

***Remark*** (Standard quantified statement).

$$\forall x\, \forall y\, \forall z,\ x = y \implies \big(R(x, z) \iff R(y, z)\big).$$

***Remark*** (Interpretation). Relations respect equality in their first coordinate.

*Remark* (Dependencies).

- Substitution of Equals

**Proposition (Substitution Preserves Relations on the Right)**

**Proposition 1.4.4** (Substitution Preserves Relations on the Right).

$$x = y \implies \big(R(z, x) \iff R(z, y)\big).$$

***Remark*** (Standard quantified statement).

$$\forall x \, \forall y \, \forall z, \; x = y \implies \big(R(z, x) \iff R(z, y)\big).$$

***Remark*** (Interpretation). Relations respect equality in their second coordinate

*Remark* (Dependencies).

- Substitution of Equals

**Proposition (Substitution Preserves Relations)**

**Proposition 1.4.5** (Substitution Preserves Relations).

$$x = x' \wedge y = y' \implies \big(R(x, y) \iff R(x', y')\big).$$

***Remark*** (Standard quantified statement).

$$\forall x, x', y, y', \; \big(x = x' \wedge y = y'\big) \implies \big(R(x, y) \iff R(x', y')\big).$$

***Remark*** (Interpretation). Relations are congruent with respect to equality in all displayed arguments.

*Remark* (Dependencies).

- Substitution Preserves Relations on the Left
- Substitution Preserves Relations on the Right

**Proposition (Substitution Preserves Functions)**

**Proposition 1.4.6** (Substitution Preserves Functions).

$$x = y \implies f(x) = f(y).$$

***Remark*** (Standard quantified statement).

$$\forall x \, \forall y, \; x = y \implies f(x) = f(y).$$

***Remark*** (Interpretation). Functions send equal inputs to equal outputs.

*Remark* (Dependencies).

- Substitution of Equals

**Proposition (Left Congruence for Operations)**

**Proposition 1.4.7** (Left Congruence for Operations).

$$x = x' \implies x \, O \, y = x' \, O \, y.$$

*Remark* (Standard quantified statement).

$$\forall x, x', y, \; x = x' \implies x \, O \, y = x' \, O \, y.$$

*Remark* (Interpretation). A binary operation respects equality in its left argument.

*Remark* (Dependencies).

- Substitution Preserves Functions

**Proposition (Right Congruence for Operations)**

**Proposition 1.4.8** (Right Congruence for Operations).

$$y = y' \implies x \, O \, y = x \, O \, y'.$$

*Remark* (Standard quantified statement).

$$\forall x, y, y', \; y = y' \implies x \, O \, y = x \, O \, y'.$$

*Remark* (Interpretation). A binary operation respects equality in its right argument.

*Remark* (Dependencies).

- Substitution Preserves Functions

**Proposition (Full Congruence for Operations)**

**Proposition 1.4.9** (Full Congruence for Operations).

$$x = x' \wedge y = y' \implies x \, O \, y = x' \, O \, y'.$$

*Remark* (Standard quantified statement).

$$\forall x, x', y, y', \; \left(x = x' \wedge y = y'\right) \implies x \, O \, y = x' \, O \, y'.$$

*Remark* (Interpretation). Binary operations are congruent with respect to equality in both arguments.

*Remark* (Dependencies).

- Left Congruence for Operations
- Right Congruence for Operations

**Proposition (Congruence with Respect to Equality Is Automatic)**

**Proposition 1.4.10** (Congruence with Respect to Equality Is Automatic). *Every function, predicate, relation, and operation respects equality in its displayed arguments.*

*Remark* (Standard quantified statement).

$$\begin{aligned} x = y &\Longrightarrow f(x) = f(y), \\ x = y &\Longrightarrow \big(P(x) \Longleftrightarrow P(y)\big), \\ x = x' \wedge y = y' &\Longrightarrow \big(R(x,y) \Longleftrightarrow R(x',y')\big), \\ x = x' \wedge y = y' &\Longrightarrow x \, O \, y = x' \, O \, y'. \end{aligned}$$

*Remark* (Interpretation). Respecting equality is automatic. Respecting a nontrivial equivalence relation is not automatic; that is why quotient constructions must prove separate respect lemmas before operations, predicates, or relations descend to classes.

*Remark* (Dependencies).

- Substitution Preserves Predicates
- Substitution Preserves Relations
- Substitution Preserves Functions
- Full Congruence for Operations

## 1.5 Equivalence

**Definition (Equivalence Relation)**

**Definition 1.5.1** (Equivalence Relation). Let $A$ be a set. A relation $\sim$ on $A$ is an *equivalence relation* if it is reflexive, symmetric, and transitive.

*Remark* (Standard quantified statement).

$$\text{EquivalenceRelation}_A(\sim) \Longleftrightarrow \Big(\forall x \in A,\ \ x \sim x\Big) \wedge \Big(\forall x, y \in A,\ \ x \sim y \Rightarrow y \sim x\Big) \wedge \Big(\forall x, y, z \in A,\ \ (x \sim$$

*Remark* (Interpretation). An equivalence relation formalizes sameness for a chosen purpose.

*Remark* (Dependencies).

- Relation
- Reflexive Relation
- Symmetric Relation
- Transitive Relation

**Corollary (Equality Is the Finest Equivalence)**

**Corollary 1.5.2** (Equality Is the Finest Equivalence). *Let $\sim$ be an equivalence relation on $A$. For all $a, b \in A$,*

$$a = b \Longrightarrow a \sim b.$$

*Remark* (Standard quantified statement).

$$\forall a, b \in A,\ a = b \Longrightarrow a \sim b.$$

*Remark* (Interpretation). Equality is the finest equivalence relation because it identifies only objects that are already the same. Any coarser equivalence relation must at least identify equal representatives.

*Remark* (Dependencies).

- Equivalence Relation
- Reflexivity of Equality

**Definition (Equivalence Class)**

**Definition 1.5.3** (Equivalence Class). Let $\sim$ be an equivalence relation on $A$. The *equivalence class* of $a \in A$ is

$$[a] := \{x \in A \mid x \sim a\}.$$

*Remark* (Standard quantified statement).

$$x \in [a] \Longleftrightarrow x \sim a.$$

*Remark* (Interpretation). The class $[a]$ collects all elements that are equivalent to the representative $a$.

*Remark* (Dependencies).

- Equivalence Relation

**Definition (Quotient Set)**

**Definition** (Quotient Set). The *quotient set* of $A$ by $\sim$ is the set of equivalence classes

$$A/\!\sim := \{[a] \mid a \in A\}.$$

**Theorem (Equivalence Classes Partition the Domain)**

**Theorem 1.5.4** (Equivalence Classes Partition the Domain). *The equivalence classes of an equivalence relation on $A$ are nonempty, cover $A$, and are pairwise disjoint.*

*Remark* (Standard quantified statement).

$$\forall a \in A,\ \ a \in [a], \qquad A = \bigcup_{a \in A} [a], \qquad [a] \cap [b] \neq \emptyset \Longrightarrow [a] = [b].$$

*Remark* (Interpretation). Equivalence classes divide the domain into non-overlapping blocks.

*Remark* (Dependencies).

- Equivalence Relation
- Equivalence Class

**Proposition (Representatives Related iff Classes Equal)**

**Proposition 1.5.5** (Representatives Related iff Classes Equal).

$$a \sim b \iff [a] = [b].$$

***Remark*** (Standard quantified statement).

$$\forall a, b \in A, \quad a \sim b \iff [a] = [b].$$

***Remark*** (Interpretation). Changing representatives inside an equivalence class does not change the class.

*Remark* (Dependencies).

- Equivalence Classes Partition the Domain

## Canonical Projection

**Definition (Canonical Projection)**

**Definition 1.5.6** (Canonical Projection). Let $\sim$ be an equivalence relation on $A$. The *canonical projection* is the map

$$\pi : A \to A/\sim, \qquad \pi(a) := [a].$$

*Remark* (Dependencies).

- Quotient Set
- Equivalence Class

**Proposition (Projection Is Surjective)**

**Proposition 1.5.7** (Projection Is Surjective). *The canonical projection $\pi : A \to A/\sim$ is surjective.*

***Remark*** (Standard quantified statement).

$$\forall C \in A/\sim \exists a \in A, \quad \pi(a) = C.$$

*Remark* (Dependencies).

- Canonical Projection
- Quotient Set

**Proposition (Projection Identifies Exactly the Equivalent Elements)**

**Proposition 1.5.8** (Projection Identifies Exactly the Equivalent Elements). *For all* $a, b \in A$,

$$\pi(a) = \pi(b) \iff a \sim b.$$

*Remark* (Dependencies).

- Canonical Projection
- Representatives Related iff Classes Equal

**Theorem (Existence of a Quotient)**

**Theorem 1.5.9** (Existence of a Quotient). *For any equivalence relation* $\sim$ *on* $A$, *the quotient set* $A/\sim$ *exists as the set of equivalence classes, and the canonical projection*

$$\pi : A \to A/\sim$$

*is well-defined.*

*Remark* (Dependencies).

- Equivalence Relation
- Quotient Set
- Canonical Projection

## Induced Structure on a Quotient

**Definition (Operation Respects an Equivalence)**

**Definition 1.5.10** (Operation Respects an Equivalence). Let $O : A \times A \to A$ be a binary operation and let $\sim$ be an equivalence relation on $A$. The operation $O$ *respects* $\sim$ if

$$\forall a, a', b, b' \in A,\ \big(a \sim a' \wedge b \sim b'\big) \implies \big(a\, O\, b\big) \sim \big(a'\, O\, b'\big).$$

*Remark* (Dependencies).

- Equivalence Relation
- Binary Arithmetic Operation
- Cartesian Product

**Definition (Left and Right Respect)**

**Definition 1.5.11** (Left and Right Respect). The operation $O$ *respects* $\sim$ on the left if

$$a \sim a' \implies (a\, O\, b) \sim (a'\, O\, b),$$

and respects $\sim$ on the right if

$$b \sim b' \implies (a\, O\, b) \sim (a\, O\, b').$$

*Remark* (Dependencies).

- Operation Respects an Equivalence
- Equivalence Relation

**Lemma (Respect Splits)**

**Lemma 1.5.12** (Respect Splits). *A binary operation $O$ respects $\sim$ if and only if it respects $\sim$ on the left and on the right.*

*Remark* (Dependencies).

- Operation Respects an Equivalence
- Left and Right Respect

**Theorem (Induced Operation on a Quotient)**

**Theorem 1.5.13** (Induced Operation on a Quotient). *If $O : A \times A \to A$ respects $\sim$, then*

$$[a]\, \overline{O}\, [b] := [a\, O\, b]$$

*defines a well-defined binary operation on $A/\sim$.*

*Remark* (Dependencies).

- Operation Respects an Equivalence
- Representatives Related iff Classes Equal
- Existence of a Quotient

**Theorem (Induced Unary Operation)**

**Theorem 1.5.14** (Induced Unary Operation). *Let $f : A \to A$. If*

$$a \sim a' \implies f(a) \sim f(a'),$$

*then*

$$\overline{f}([a]) := [f(a)]$$

*defines a well-defined unary operation on $A/\sim$.*

*Remark* (Dependencies).

- Representatives Related iff Classes Equal
- Existence of a Quotient

**Definition (Predicate Respects an Equivalence)**

**Definition 1.5.15** (Predicate Respects an Equivalence). Let $P$ be a predicate on $A$. The predicate $P$ *respects* $\sim$ if

$$\forall a, a' \in A,\ a \sim a' \implies \big(P(a) \iff P(a')\big).$$

*Remark* (Dependencies).

- Equivalence Relation
- Predicate

**Theorem (Induced Predicate on a Quotient)**

**Theorem 1.5.16** (Induced Predicate on a Quotient). *If $P$ respects $\sim$, then*

$$\overline{P}([a]) :\iff P(a)$$

*defines a well-defined predicate on $A/\sim$.*

*Remark* (Dependencies).

- Predicate Respects an Equivalence
- Representatives Related iff Classes Equal

**Definition (Relation Respects an Equivalence)**

**Definition 1.5.17** (Relation Respects an Equivalence). Let $R$ be a binary relation on $A$. The relation $R$ *respects* $\sim$ if

$$\forall a, a', b, b' \in A,\ \big(a \sim a' \wedge b \sim b'\big) \implies \big(R(a, b) \iff R(a', b')\big).$$

*Remark* (Dependencies).

- Equivalence Relation
- Relation

**Theorem (Induced Relation on a Quotient)**

**Theorem 1.5.18** (Induced Relation on a Quotient). *If $R$ respects $\sim$, then*

$$\overline{R}([a], [b]) :\iff R(a, b)$$

*defines a well-defined binary relation on $A/\sim$.*

*Remark* (Dependencies).

- Relation Respects an Equivalence

- Representatives Related iff Classes Equal

**Theorem (Induced Map out of a Quotient)**

**Theorem 1.5.19** (Induced Map out of a Quotient). *Let* $g : A \to B$. *If*

$$a \sim a' \implies g(a) = g(a'),$$

*then there is a unique well-defined map*

$$\overline{g} : A/\!\sim \to B$$

*such that*

$$\overline{g}([a]) = g(a) \qquad \textit{and} \qquad \overline{g} \circ \pi = g,$$

*where* $\pi : A \to A/\!\sim$ *is the canonical projection.*

*Remark* (Dependencies).

- Canonical Projection
- Projection Identifies Exactly the Equivalent Elements
- Projection Is Surjective

**Theorem (Induced Partial Operation on a Quotient)**

**Theorem 1.5.20** (Induced Partial Operation on a Quotient). *Let* $D \subseteq A$ *be saturated under* $\sim$, *meaning*

$$a \in D \wedge a \sim a' \implies a' \in D.$$

*If* $f : D \to A$ *respects* $\sim$ *on* $D$, *then*

$$\overline{f}([a]) := [f(a)]$$

*is a well-defined partial operation on* $A/\!\sim$ *with domain*

$$D/\!\sim := \{[a] \in A/\!\sim : a \in D\}.$$

*Remark* (Dependencies).

- Induced Unary Operation
- Induced Predicate on a Quotient

**Proposition (Representative Independence Is Necessary)**

**Proposition 1.5.21** (Representative Independence Is Necessary). *If a rule on classes*

$$[a] \,\overline{O}\, [b] := [a \, O \, b]$$

*is well-defined on* $A/\!\sim$, *then* $O$ *respects* $\sim$.

*Remark* (Dependencies).

- Operation Respects an Equivalence
- Projection Identifies Exactly the Equivalent Elements
- Induced Operation on a Quotient

**Lemma (Distinct Equivalence Classes Are Disjoint)**

**Lemma 1.5.22** (Distinct Equivalence Classes Are Disjoint).

$$[a] \neq [b] \Longrightarrow [a] \cap [b] = \emptyset.$$

***Remark*** (Standard quantified statement).

$$\forall a, b \in A, \quad [a] \neq [b] \Longrightarrow [a] \cap [b] = \emptyset.$$

***Remark*** (Interpretation). Two equivalence classes either coincide or do not overlap.

*Remark* (Dependencies).

- Equivalence Classes Partition the Domain

# Chapter 2

# Arithmetic Operations and Relations

**arithmetic-operations-relations**

Identity, Equality, and Equivalence $\rightarrow$
**Arithmetic Operations and Relations** $\rightarrow$ Natural Numbers ($\mathbb{N}$)

## 2.1 Arithmetic Operations and Relations

## 2.2 Closure

**Definition (Unary Arithmetic Operation)**

**Definition 2.2.1** (Unary Arithmetic Operation). Let $K$ be a domain. A *unary arithmetic operation* on $K$ is a function

$$O : K \to K.$$

*Remark* (Standard quantified statement).

$$\text{UnaryOperation}_K(O) \iff \forall x \in K,\ \ O(x) \in K.$$

*Remark* (Interpretation). Unary closure is the abstract form behind successor, negation, and other single-input transformations.

*Remark* (Dependencies).

- Function

**Definition (Binary Arithmetic Operation)**

**Definition 2.2.2** (Binary Arithmetic Operation). Let $K$ be a domain. A *binary arithmetic operation* on $K$ is a function

$$O : K \times K \to K.$$

*Remark* (Standard quantified statement).

$$\text{BinaryOperation}_K(O) \iff \forall x, y \in K, \;\; x \, O \, y \in K.$$

*Remark* (Interpretation). Binary closure is the abstract form behind addition and multiplication.

*Remark* (Dependencies).

- Function
- Cartesian Product

**Definition (Unary Closure)**

**Definition 2.2.3** (Unary Closure). A unary operation $O$ is *closed on* $K$ if applying it to an element of $K$ produces an element of $K$.

*Remark* (Standard quantified statement).

$$\forall x \in K, \;\; O(x) \in K.$$

*Remark* (Interpretation). The operation does not leave the domain.

*Remark* (Dependencies).

- Unary Arithmetic Operation

**Definition (Binary Closure)**

**Definition 2.2.4** (Binary Closure). A binary operation $O$ is *closed on* $K$ if combining two elements of $K$ produces an element of $K$.

*Remark* (Standard quantified statement).

$$\forall x, y \in K, \;\; x \, O \, y \in K.$$

*Remark* (Interpretation). The operation is internal to the domain.

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Operation Well-Definedness)**

**Definition 2.2.5** (Operation Well-Definedness). Let $\sim$ be an equivalence relation on $A$. A binary operation $O$ respects $\sim$ if

$$a \sim a' \wedge b \sim b' \Longrightarrow a\, O\, b \sim a'\, O\, b'.$$

*Remark* (Standard quantified statement).

$$\forall a, a', b, b' \in A,\ \left(a \sim a' \wedge b \sim b'\right) \Longrightarrow a\, O\, b \sim a'\, O\, b'.$$

*Remark* (Interpretation). This is the condition that lets an operation descend to equivalence classes.

*Remark* (Dependencies).

- Equivalence Relation

**Lemma (Left Respect)**

**Lemma 2.2.6** (Left Respect). *If a binary operation respects $\sim$, then*

$$a \sim a' \Longrightarrow a\, O\, b \sim a'\, O\, b.$$

*Remark* (Standard quantified statement).

$$\forall a, a', b \in A,\ a \sim a' \Rightarrow a\, O\, b \sim a'\, O\, b.$$

*Remark* (Interpretation). Full respect includes respect in the left coordinate

*Remark* (Dependencies).

- Operation Well-Definedness

**Lemma (Right Respect)**

**Lemma 2.2.7** (Right Respect). *If a binary operation respects $\sim$, then*

$$b \sim b' \Longrightarrow a\, O\, b \sim a\, O\, b'.$$

*Remark* (Standard quantified statement).

$$\forall a, b, b' \in A,\ b \sim b' \Rightarrow a\, O\, b \sim a\, O\, b'.$$

*Remark* (Interpretation). Full respect includes respect in the right coordinate

*Remark* (Dependencies).

- Operation Well-Definedness

**Theorem (Induced Operation)**

**Theorem 2.2.8** (Induced Operation). *The operation*

$$[a]\, \overline{O}\, [b] := [a\, O\, b]$$

*on $A/\sim$ is well-defined if and only if $O$ respects $\sim$.*

*Remark* (Standard quantified statement).

$$\text{WellDefined}(\overline{O}) \iff \forall a, a', b, b' \in A,\ \big(a \sim a' \wedge b \sim b'\big) \Rightarrow a\, O\, b \sim a'\, O\, b'.$$

*Remark* (Interpretation). This theorem is the abstract engine behind quotient constructions such as integers and rationals.

*Remark* (Dependencies).

- Quotient Set
- Operation Well-Definedness

## 2.3 Algebraic Laws

**Definition (Associativity)**

**Definition 2.3.1** (Associativity).

$$\forall x, y, z,\ \ x\, O\, (y\, O\, z) = (x\, O\, y)\, O\, z.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Theorem (General Associativity)**

**Theorem 2.3.2** (General Associativity). *If $O$ is associative, then every bracketing of $x_1\, O \cdots O\, x_n$ has the same value.*

*Remark* (Dependencies).

- Associativity

**Definition (Commutativity)**

**Definition 2.3.3** (Commutativity).

$$\forall x, y,\ \ x\, O\, y = y\, O\, x.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Theorem (General Commutativity)**

**Theorem 2.3.4** (General Commutativity). *If $O$ is associative and commutative, then any finite reordering of $x_1\ O\ \cdots\ O\ x_n$ has the same value.*

*Remark* (Dependencies).

- Associativity

- Commutativity

**Definition (Left Distributivity)**

**Definition 2.3.5** (Left Distributivity). The operation $\otimes$ is left distributive over $\oplus$ if

$$x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z),$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Right Distributivity)**

**Definition 2.3.6** (Right Distributivity). The operation $\otimes$ is right distributive over $\oplus$ if

$$(y \oplus z) \otimes x = (y \otimes x) \oplus (z \otimes x),$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Two-Sided Distributivity)**

**Definition 2.3.7** (Two-Sided Distributivity). The operation $\otimes$ is two-sided distributive over $\oplus$ if it is both left distributive and right distributive.

*Remark* (Dependencies).

- Left Distributivity
- Right Distributivity

**Definition (Left Cancellation)**

**Definition 2.3.8** (Left Cancellation). An operation has left cancellation if

$$x \, O \, y = x \, O \, z \Longrightarrow y = z,$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Right Cancellation)**

**Definition 2.3.9** (Right Cancellation). An operation has right cancellation if

$$y \, O \, x = z \, O \, x \Longrightarrow y = z.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Two-Sided Cancellation)**

**Definition 2.3.10** (Two-Sided Cancellation). An operation has two-sided cancellation if it has both left cancellation and right cancellation.

*Remark* (Dependencies).

- Left Cancellation
- Right Cancellation

**Proposition (Invertibility Implies Left Cancellation)**

**Proposition 2.3.11** (Invertibility Implies Left Cancellation). *In an associative operation with identity, invertible elements satisfy left cancellation.*

*Remark* (Dependencies).

- Associativity
- Two-Sided Identity
- Two-Sided Inverse
- Left Cancellation

**Proposition (Invertibility Implies Cancellation)**

**Proposition 2.3.12** (Invertibility Implies Cancellation). *In an associative operation with identity, invertible elements satisfy two-sided cancellation.*

*Remark* (Dependencies).

- Invertibility Implies Left Cancellation
- Invertibility Implies Right Cancellation
- Two-Sided Cancellation

**Proposition (Invertibility Implies Right Cancellation)**

**Proposition 2.3.13** (Invertibility Implies Right Cancellation). *In an associative operation with identity, invertible elements satisfy right cancellation.*

*Remark* (Dependencies).

- Associativity
- Two-Sided Identity
- Two-Sided Inverse
- Right Cancellation

## 2.4 Identity Elements

**Definition (Left Identity)**

**Definition 2.4.1** (Left Identity).

$$\forall x, \quad e_L \, O \, x = x.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Right Identity)**

**Definition 2.4.2** (Right Identity).

$$\forall x, \quad x \, O \, e_R = x.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Two-Sided Identity)**

**Definition 2.4.3** (Two-Sided Identity). An element $e$ is a *two-sided identity* for $O$ if it is both a left identity and a right identity.

*Remark* (Dependencies).

- Left Identity
- Right Identity

**Lemma (Left and Right Identities Coincide)**

**Lemma 2.4.4** (Left and Right Identities Coincide). *If $e_L$ is a left identity and $e_R$ is a right identity for the same operation, then $e_L = e_R$.*

*Remark* (Standard quantified statement).

$$(\forall x, \quad e_L \, O \, x = x) \wedge (\forall x, \quad x \, O \, e_R = x) \Longrightarrow e_L = e_R.$$

*Remark* (Dependencies).

- Left Identity
- Right Identity

**Theorem (Uniqueness of Identity)**

**Theorem 2.4.5** (Uniqueness of Identity). *A two-sided identity for a binary operation is unique.*

*Remark* (Standard quantified statement).

$$\text{Identity}(e) \wedge \text{Identity}(e') \implies e = e'.$$

*Remark* (Dependencies).

- Left and Right Identities Coincide

## 2.5 Inverse Operations

**Definition (Left Inverse)**

**Definition 2.5.1** (Left Inverse). An element $x'$ is a left inverse of $x$ with respect to identity $e$ if

$$x' \, O \, x = e.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation
- Two-Sided Identity

**Definition (Right Inverse)**

**Definition 2.5.2** (Right Inverse). An element $x''$ is a right inverse of $x$ with respect to identity $e$ if

$$x \, O \, x'' = e.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation
- Two-Sided Identity

**Definition (Two-Sided Inverse)**

**Definition 2.5.3** (Two-Sided Inverse). An element $y$ is a two-sided inverse of $x$ if

$$y \, O \, x = e \quad \text{and} \quad x \, O \, y = e.$$

*Remark* (Dependencies).

- Left Inverse
- Right Inverse

**Lemma (Left and Right Inverses Coincide)**

**Lemma 2.5.4** (Left and Right Inverses Coincide). *In an associative operation with identity, a left inverse and a right inverse of the same element are equal.*

*Remark* (Standard quantified statement).

$$(x' \, O \, x = e) \wedge (x \, O \, x'' = e) \Longrightarrow x' = x''.$$

*Remark* (Dependencies).

- Associativity
- Two-Sided Identity
- Left Inverse
- Right Inverse

**Theorem (Uniqueness of Inverse)**

**Theorem 2.5.5** (Uniqueness of Inverse). *In an associative operation with identity, inverses are unique.*

*Remark* (Dependencies).

- Left and Right Inverses Coincide

**Definition (Right Solvability)**

**Definition 2.5.6** (Right Solvability). Right solvability means

$$\forall x, y \, \exists z, \quad x = y \, O \, z,$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Left Solvability)**

**Definition 2.5.7** (Left Solvability). Left solvability means

$$\forall x, y \, \exists z, \quad x = z \, O \, y.$$

*Remark* (Dependencies).

- Binary Arithmetic Operation

**Definition (Two-Sided Solvability)**

**Definition 2.5.8** (Two-Sided Solvability). Two-sided solvability means both left solvability and right solvability hold.

*Remark* (Dependencies).

- Left Solvability
- Right Solvability

**Proposition (Solvability and Inverses)**

**Proposition 2.5.9** (Solvability and Inverses). *In a monoid, solvability is equivalent to the existence of inverses.*

*Remark* (Dependencies).

- Monoid
- Two-Sided Inverse
- Right Solvability
- Left Solvability
- Two-Sided Solvability

## 2.6 Algebraic Structures

**Definition (Semigroup)**

**Definition 2.6.1** (Semigroup). A *semigroup* is a domain with a closed associative binary operation.

*Remark* (Dependencies).

- Binary Arithmetic Operation
- Binary Closure
- Associativity

**Definition (Monoid)**

**Definition 2.6.2** (Monoid). A *monoid* is a semigroup with a two-sided identity element.

*Remark* (Dependencies).

- Semigroup
- Two-Sided Identity

**Definition (Commutative Monoid)**

**Definition 2.6.3** (Commutative Monoid). A *commutative monoid* is a monoid whose operation is commutative.

*Remark* (Dependencies).

- Monoid
- Commutativity

**Definition (Group)**

**Definition 2.6.4** (Group). A *group* is a monoid in which every element has a two-sided inverse.

*Remark* (Dependencies).

- Monoid
- Two-Sided Inverse

**Definition (Abelian Group)**

**Definition 2.6.5** (Abelian Group). An *abelian group* is a group whose operation is commutative.

*Remark* (Dependencies).

- Group
- Commutativity

**Definition (Semiring)**

**Definition 2.6.6** (Semiring). A *semiring* is a domain with addition and multiplication such that addition is a commutative monoid, multiplication is a monoid, multiplication distributes over addition, and the additive identity is absorbing for multiplication.

*Remark* (Dependencies).

- Commutative Monoid
- Monoid

**Definition (Ring)**

**Definition 2.6.7** (Ring). A *ring* is a domain whose additive structure is an abelian group, whose multiplicative structure is a monoid, and whose multiplication distributes over addition.

*Remark* (Dependencies).

- Abelian Group
- Monoid

**Definition (Commutative Ring)**

**Definition 2.6.8** (Commutative Ring). A *commutative ring* is a ring whose multiplication is commutative.

*Remark* (Dependencies).

- Ring
- Commutativity

**Definition (Integral Domain)**

**Definition 2.6.9** (Integral Domain). An *integral domain* is a commutative ring with $0 \neq 1$ and no zero divisors.

*Remark* (Dependencies).

- Commutative Ring

**Definition (Field)**

**Definition 2.6.10** (Field). A *field* is a commutative ring in which every nonzero element has a multiplicative inverse.

*Remark* (Dependencies).

- Commutative Ring
- Two-Sided Inverse

**Definition (Ordered Field)**

**Definition 2.6.11** (Ordered Field). An *ordered field* is a field equipped with a total order compatible with addition and multiplication by positive elements.

*Remark* (Dependencies).

- Field
- Ordered Field Trichotomy
- Ordered Field Addition Law
- Ordered Field Multiplication Law

**Axiom (Ordered Field Trichotomy)**

**Axiom 2.6.12** (Ordered Field Trichotomy). In an ordered field, for all elements $a$ and $b$, exactly one of the following holds:

$$a = b, \qquad a < b, \qquad b < a.$$

*Remark* (Interpretation). Any two elements of an ordered field are comparable, and comparison has no overlap: equality, being below, and being above are mutually exclusive cases.

**Axiom (Ordered Field Addition Law)**

**Axiom 2.6.13** (Ordered Field Addition Law). In an ordered field, for all elements $a$, $b$, and $c$,

$$a < b \iff a + c < b + c.$$

*Remark* (Interpretation). Adding the same element to both sides of a strict inequality preserves and reflects the order.

**Axiom (Ordered Field Multiplication Law)**

**Axiom 2.6.14** (Ordered Field Multiplication Law). In an ordered field, for all elements $a$, $b$, and $c$,

$$c > 0 \implies \big(ac < bc \iff a < b\big),$$

and

$$c < 0 \implies \big(ac < bc \iff b < a\big).$$

Moreover, positive elements are closed under multiplication:

$$0 < a \wedge 0 < b \implies 0 < ab.$$

*Remark* (Interpretation). Multiplication by a positive element preserves strict order, while multiplication by a negative element reverses strict order. The positive elements are also closed under multiplication.

**Theorem (Ordered Field Transitivity)**

**Theorem 2.6.15** (Ordered Field Transitivity). *In an ordered field, for all elements $a$, $b$, and $c$,*

$$a < b \wedge b < c \implies a < c.$$

*Remark* (Interpretation). Strict order in an ordered field chains: an element below a second element which is below a third is below the third.

*Remark* (Dependencies).

- Ordered Field

**Definition (Positive Cone)**

**Definition 2.6.16** (Positive cone). Let $F$ be a field. A subset $P \subset F$ is called a *positive cone* if it satisfies:

1. if $a, b \in P$, then $a + b \in P$;
2. if $a, b \in P$, then $ab \in P$;
3. for every $a \in F$, exactly one of the following holds:

$$a \in P, \qquad a = 0, \qquad -a \in P.$$

Define the order relation by

$$a < b \iff b - a \in P.$$

*Remark* (Dependencies).

- Field
- Subset
- Trichotomy

## 2.7 Derived Laws

**Proposition (Left Zero Absorption)**

**Proposition 2.7.1** (Left Zero Absorption). *In a ring,*

$$0 \cdot a = 0.$$

*Remark* (Dependencies).

- Ring

**Proposition (Right Zero Absorption)**

**Proposition 2.7.2** (Right Zero Absorption). *In a ring,*

$$a \cdot 0 = 0.$$

*Remark* (Dependencies).

- Ring

**Proposition (Zero Absorption)**

**Proposition 2.7.3** (Zero Absorption). *In a ring,*

$$0 \cdot a = 0 \qquad \textit{and} \qquad a \cdot 0 = 0.$$

*Remark* (Dependencies).

- Left Zero Absorption
- Right Zero Absorption

**Proposition (Left Sign Rule)**

**Proposition 2.7.4** (Left Sign Rule). *In a ring,*

$$(-a) \cdot b = -(a \cdot b).$$

*Remark* (Dependencies).

- Ring
- Left Zero Absorption

**Proposition (Right Sign Rule)**

**Proposition 2.7.5** (Right Sign Rule). *In a ring,*

$$a \cdot (-b) = -(a \cdot b).$$

*Remark* (Dependencies).

- Ring
- Right Zero Absorption

**Proposition (Sign Rule)**

**Proposition 2.7.6** (Sign Rule). *In a ring,*

$$(-a) \cdot b = -(a \cdot b) \qquad \textit{and} \qquad a \cdot (-b) = -(a \cdot b).$$

*Remark* (Dependencies).

- Left Sign Rule
- Right Sign Rule

**Corollary (Negative One Rule)**

**Corollary 2.7.7** (Negative One Rule). *In a ring,*

$$(-1) \cdot a = -a.$$

*Remark* (Dependencies).

- Left Sign Rule

**Proposition (Product of Negatives)**

**Proposition 2.7.8** (Product of Negatives). *In a ring,*

$$(-a) \cdot (-b) = a \cdot b.$$

*Remark* (Dependencies).

- Left Sign Rule
- Right Sign Rule

**Theorem (No Zero Divisors)**

**Theorem 2.7.9** (No Zero Divisors). *In an integral domain,*

$$a \cdot b = 0 \Longrightarrow a = 0 \vee b = 0.$$

*Remark* (Dependencies).

- Integral Domain

## 2.8 Order Relations

**Definition (Reflexive Relation)**

**Definition 2.8.1** (Reflexive Relation).

$$\forall x, \quad xRx.$$

*Remark* (Dependencies).

- Relation
- Reflexive Relation

**Definition (Irreflexive Relation)**

**Definition 2.8.2** (Irreflexive Relation).

$$\forall x, \quad \neg(xRx).$$

*Remark* (Dependencies).

- Relation
- Irreflexive Relation

**Definition (Symmetric Relation)**

**Definition 2.8.3** (Symmetric Relation).

$$\forall x, y, \quad xRy \Longrightarrow yRx.$$

*Remark* (Dependencies).

- Relation
- Symmetric Relation

**Definition (Antisymmetric Relation)**

**Definition 2.8.4** (Antisymmetric Relation).

$$\forall x, y, \quad (xRy \wedge yRx) \Longrightarrow x = y.$$

*Remark* (Dependencies).

- Relation
- Antisymmetric Relation

**Definition (Asymmetric Relation)**

**Definition 2.8.5** (Asymmetric Relation).

$$\forall x, y, \quad xRy \Longrightarrow \neg(yRx).$$

*Remark* (Dependencies).

- Relation
- Asymmetric Relation

**Definition (Transitive Relation)**

**Definition 2.8.6** (Transitive Relation).

$$\forall x, y, z, \quad (xRy \wedge yRz) \Longrightarrow xRz.$$

*Remark* (Dependencies).

- Relation
- Transitive Relation

**Definition (Total Relation)**

**Definition 2.8.7** (Total Relation).

$$\forall x, y, \quad xRy \vee yRx.$$

*Remark* (Dependencies).

- Relation
- Total Relation

**Definition (Trichotomy)**

**Definition 2.8.8** (Trichotomy). The relation $<$ satisfies *trichotomy* if exactly one of

$$x < y, \qquad x = y, \qquad y < x$$

holds for every pair $x, y$.

*Remark* (Dependencies).

- Strict Order
- Total Relation

## 2.9 Composed Relations

**Definition (Preorder)**

**Definition** (Preorder). A *preorder* is a reflexive and transitive relation.

**Definition (Partial Order)**

**Definition** (Partial Order). A *partial order* is a preorder that is antisymmetric.

**Definition (Strict Partial Order)**

**Definition 2.9.1** (Strict Partial Order). A *strict partial order* is an irreflexive and transitive relation.

*Remark* (Dependencies).

- Irreflexive Relation
- Transitive Relation

**Definition (Total Order)**

**Definition** (Total Order). A *total order* is a partial order whose relation is total.

**Definition (Strict Total Order)**

**Definition 2.9.2** (Strict Total Order). A *strict total order* is a strict partial order satisfying trichotomy.

*Remark* (Dependencies).

- Strict Partial Order
- Trichotomy

**Definition (Well-Order)**

**Definition** (Well-Order). A *well-order* is a total order in which every nonempty subset has a least element.

## 2.10 Order Compatibility

**Definition (Right Monotony)**

**Definition 2.10.1** (Right Monotony). Right monotony means

$$yRz \implies (x\,O\,y)R(x\,O\,z),$$

*Remark* (Dependencies).

- Relation
- Binary Arithmetic Operation

**Definition (Left Monotony)**

**Definition 2.10.2** (Left Monotony). Left monotony means

$$yRz \implies (y\,O\,x)R(z\,O\,x).$$

*Remark* (Dependencies).

- Relation
- Binary Arithmetic Operation

**Definition (Two-Sided Monotony)**

**Definition 2.10.3** (Two-Sided Monotony). Two-sided monotony means both left monotony and right monotony hold.

*Remark* (Dependencies).

- Left Monotony
- Right Monotony

**Definition (Right Order Reflection)**

**Definition 2.10.4** (Right Order Reflection). Right reflection means

$$(x\,O\,y)R(x\,O\,z) \implies yRz,$$

*Remark* (Dependencies).

- Relation
- Binary Arithmetic Operation

**Definition (Left Order Reflection)**

**Definition 2.10.5** (Left Order Reflection). Left reflection means

$$(y\,O\,x)R(z\,O\,x) \implies yRz.$$

*Remark* (Dependencies).

- Relation
- Binary Arithmetic Operation

**Definition (Two-Sided Order Reflection)**

**Definition 2.10.6** (Two-Sided Order Reflection). Two-sided order reflection means both left order reflection and right order reflection hold.

*Remark* (Dependencies).

- Left Order Reflection
- Right Order Reflection

**Definition (Monotone Map)**

**Definition 2.10.7** (Monotone Map).

$$x \leq y \Longrightarrow f(x) \leq f(y).$$

*Remark* (Dependencies).

- Function
- Relation

**Definition (Strictly Monotone Map)**

**Definition 2.10.8** (Strictly Monotone Map).

$$x < y \Longrightarrow f(x) < f(y).$$

*Remark* (Dependencies).

- Function
- Relation

**Proposition (Addition Preserves Order on the Left)**

**Proposition 2.10.9** (Addition Preserves Order on the Left).

$$y < z \Longrightarrow x + y < x + z$$

*in ordered arithmetic systems where addition is strictly order-preserving.*

*Remark* (Dependencies).

- Right Monotony

**Proposition (Addition Preserves Order on the Right)**

**Proposition 2.10.10** (Addition Preserves Order on the Right).

$$y < z \implies y + x < z + x$$

*in ordered arithmetic systems where addition is strictly order-preserving.*

*Remark* (Dependencies).

- Left Monotony

**Proposition (Addition Preserves Order)**

**Proposition 2.10.11** (Addition Preserves Order). *In ordered arithmetic systems where addition is strictly order-preserving, addition preserves order on both sides.*

*Remark* (Dependencies).

- Addition Preserves Order on the Left
- Addition Preserves Order on the Right

**Proposition (Multiplication Preserves Order on the Left)**

**Proposition 2.10.12** (Multiplication Preserves Order on the Left).

$$0 < x \wedge y < z \implies x \cdot y < x \cdot z$$

*in ordered arithmetic systems where multiplication by positive elements is strictly order-preserving.*

*Remark* (Dependencies).

- Right Monotony
- Ordered Field

**Proposition (Multiplication Preserves Order on the Right)**

**Proposition 2.10.13** (Multiplication Preserves Order on the Right).

$$0 < x \wedge y < z \implies y \cdot x < z \cdot x$$

*in ordered arithmetic systems where multiplication by positive elements is strictly order-preserving.*

*Remark* (Dependencies).

- Left Monotony
- Ordered Field

**Proposition (Multiplication Preserves Order)**

**Proposition 2.10.14** (Multiplication Preserves Order). *In ordered arithmetic systems where multiplication by positive elements is strictly order-preserving, multiplication preserves order on both sides.*

*Remark* (Dependencies).

- Multiplication Preserves Order on the Left
- Multiplication Preserves Order on the Right

## 2.11 Structure-Preserving Maps

**Definition (Homomorphism)**

**Definition 2.11.1** (Homomorphism). A map $f$ is a *homomorphism* for an operation $O$ when

$$f(x\,O\,y) = f(x)\,O'\,f(y).$$

For ring-like structures it preserves both addition and multiplication, and preserves $1$ when the language includes a distinguished multiplicative identity.

*Remark* (Dependencies).

- Function
- Binary Arithmetic Operation

**Definition (Injective Homomorphism)**

**Definition 2.11.2** (Injective Homomorphism). An *injective homomorphism* is a homomorphism whose underlying function is injective.

*Remark* (Dependencies).

- Homomorphism
- Injective Function

**Definition (Order Embedding)**

**Definition 2.11.3** (Order Embedding).

$$xRy \iff f(x)R'f(y).$$

*Remark* (Dependencies).

- Function
- Relation

- Order Embedding

**Definition (Structure Embedding)**

**Definition 2.11.4** (Structure Embedding). A *structure embedding* is an injective homomorphism that also embeds the relevant order relation.

*Remark* (Dependencies).

- Injective Homomorphism
- Order Embedding

**Proposition (Composition of Homomorphisms)**

**Proposition 2.11.5** (Composition of Homomorphisms). *The composition of two homomorphisms is a homomorphism.*

*Remark* (Dependencies).

- Homomorphism

**Corollary (Identity Map Is a Homomorphism)**

**Corollary 2.11.6** (Identity Map Is a Homomorphism). *The identity map on a structure is a homomorphism from that structure to itself.*

*Remark* (Dependencies).

- Homomorphism

## 2.12 Field and Ordered-Field Laws

**Theorem (Group Cancellation)**

**Theorem 2.12.1** (Group Cancellation). *In a group, cancellation holds on both sides:*

$$a * c = b * c \Longrightarrow a = b \qquad \textit{and} \qquad c * a = c * b \Longrightarrow a = b.$$

*Remark* (Dependencies).

- Group
- Uniqueness of Inverse

**Theorem (Group Uniqueness of Inverse)**

**Theorem 2.12.2** (Group Uniqueness of Inverse). *In a group, each element has a unique inverse.*

*Remark* (Dependencies).

- Group
- Uniqueness of Inverse

**Theorem (Field Zero Product)**

**Theorem 2.12.3** (Field Zero Product). *In a field,*

$$a \cdot b = 0 \quad \iff \quad a = 0 \ \textit{or} \ b = 0.$$

*Remark* (Dependencies).

- Field
- No Zero Divisors

**Theorem (Ring Double Negation)**

**Theorem 2.12.4** (Ring Double Negation). *In a ring,*

$$-(-a) = a.$$

*Remark* (Dependencies).

- Ring
- Uniqueness of Inverse

**Theorem (Ring Negation of Product)**

**Theorem 2.12.5** (Ring Negation of Product). *In a ring,*

$$(-a) \cdot b = -(a \cdot b) \qquad \textit{and} \qquad a \cdot (-b) = -(a \cdot b).$$

*Remark* (Dependencies).

- Sign Rule

**Theorem (Ring Product of Negatives)**

**Theorem 2.12.6** (Ring Product of Negatives). *In a ring,*

$$(-a) \cdot (-b) = a \cdot b.$$

*Remark* (Dependencies).

- Product of Negatives

**Theorem (Field Multiplicative Cancellation)**

**Theorem 2.12.7** (Field Multiplicative Cancellation). *In a field,*

$$c \neq 0 \wedge a \cdot c = b \cdot c \Longrightarrow a = b.$$

*Remark* (Dependencies).

- Field
- Group Cancellation

**Theorem (Field Uniqueness of Inverse)**

**Theorem 2.12.8** (Field Uniqueness of Inverse). *In a field, each nonzero element has a unique multiplicative inverse.*

*Remark* (Dependencies).

- Field
- Group Uniqueness of Inverse

**Theorem (Field Inverse of One)**

**Theorem 2.12.9** (Field Inverse of One). *In a field,*

$$1^{-1} = 1.$$

*Remark* (Dependencies).

- Field Uniqueness of Inverse

**Theorem (Field Inverse of Product)**

**Theorem 2.12.10** (Field Inverse of Product). *In a field, if $a \neq 0$ and $b \neq 0$, then*

$$(a \cdot b)^{-1} = a^{-1} \cdot b^{-1}.$$

*Remark* (Dependencies).

- Field Uniqueness of Inverse
- Field Zero Product

**Theorem (Field Inverse Involution)**

**Theorem 2.12.11** (Field Inverse Involution). *In a field, if $a \neq 0$, then*

$$(a^{-1})^{-1} = a.$$

*Remark* (Dependencies).

- Field Uniqueness of Inverse

**Definition (Field Integer Power)**

**Definition 2.12.12** (Field Integer Power). Let $F$ be a field. For $a \in F$, powers $a^n$ with $n \in \mathbb{W}$ are defined recursively. If $a \neq 0$ and $n \in \mathbb{Z}$ is negative, define

$$a^n := (a^{-1})^{-n}.$$

*Remark* (Dependencies).

- Field
- Field Uniqueness of Inverse

**Theorem (Field Exponent Addition)**

**Theorem 2.12.13** (Field Exponent Addition). *In a field, if* $a \neq 0$ *and* $m, n \in \mathbb{Z}$, *then*

$$a^{m+n} = a^m \cdot a^n.$$

*Remark* (Dependencies).

- Field Integer Power

**Theorem (Field Power of Power)**

**Theorem 2.12.14** (Field Power of Power). *In a field, if* $a \neq 0$ *and* $m, n \in \mathbb{Z}$, *then*

$$(a^m)^n = a^{m \cdot n}.$$

*Remark* (Dependencies).

- Field Integer Power

**Theorem (Field Power of Product)**

**Theorem 2.12.15** (Field Power of Product). *In a field, if* $a \neq 0$, $b \neq 0$, *and* $n \in \mathbb{Z}$, *then*

$$(a \cdot b)^n = a^n \cdot b^n.$$

*Remark* (Dependencies).

- Field Integer Power
- Field Inverse of Product

**Corollary (Generic Field Laws)**

**Corollary 2.12.16** (Generic Field Laws). *Every field satisfies the zero-product law, cancellation laws, uniqueness of inverses, inverse laws, ring sign laws, and integer-exponent laws listed in this topic.*

*Remark* (Dependencies).

- Field Zero Product
- Field Multiplicative Cancellation
- Field Inverse of Product
- Field Exponent Addition

**Theorem (Ordered Field Squares Are Nonnegative)**

**Theorem 2.12.17** (Ordered Field Squares Are Nonnegative). *In an ordered field,*

$$0 \leq a^2.$$

*Remark* (Dependencies).

- Ordered Field

**Theorem (Ordered Field One Is Positive)**

**Theorem 2.12.18** (Ordered Field One Is Positive). *In an ordered field,*

$$0 < 1.$$

*Remark* (Dependencies).

- Ordered Field
- Ordered Field Squares Are Nonnegative

**Theorem (Ordered Field Positive Product)**

**Theorem 2.12.19** (Ordered Field Positive Product). *In an ordered field,*

$$0 < a \wedge 0 < b \Longrightarrow 0 < a \cdot b.$$

*Remark* (Dependencies).

- Ordered Field

**Theorem (Ordered Field Positive Inverses)**

**Theorem 2.12.20** (Ordered Field Positive Inverses). *In an ordered field,*

$$0 < a \Longrightarrow 0 < a^{-1}.$$

*Remark* (Dependencies).

- Ordered Field Positive Product
- Field Inverse of One

**Theorem (Ordered Field Positive Multiplication Preserves Order)**

**Theorem 2.12.21** (Ordered Field Positive Multiplication Preserves Order). *In an ordered field,*

$$0 < c \wedge a < b \Longrightarrow c \cdot a < c \cdot b.$$

*The corresponding non-strict implication also holds.*

*Remark* (Dependencies).

- Ordered Field
- Ordered Field Positive Product

**Theorem (Ordered Field Negative Multiplication Reverses Order)**

**Theorem 2.12.22** (Ordered Field Negative Multiplication Reverses Order). *In an ordered field,*

$$c < 0 \wedge a < b \Longrightarrow c \cdot b < c \cdot a.$$

*The corresponding non-strict implication also holds.*

*Remark* (Dependencies).

- Ordered Field Positive Multiplication Preserves Order
- Ring Negation of Product

**Theorem (Ordered Field Fraction Comparison)**

**Theorem 2.12.23** (Ordered Field Fraction Comparison). *In an ordered field, if* $b \cdot d > 0$, *then*

$$\frac{a}{b} < \frac{c}{d} \quad \Longleftrightarrow \quad a \cdot d < c \cdot b.$$

*Remark* (Dependencies).

- Ordered Field Positive Multiplication Preserves Order
- Field Multiplicative Cancellation

**Corollary (Generic Ordered-Field Laws)**

**Corollary 2.12.24** (Generic Ordered-Field Laws). *Every ordered field satisfies the standard positivity, order-compatibility, inverse-order, sign, and comparison laws listed in this topic.*

*Remark* (Dependencies).

- Ordered Field One Is Positive
- Ordered Field Positive Product
- Ordered Field Positive Inverses
- Ordered Field Positive Multiplication Preserves Order

# Chapter 3

# Peano Systems

**peano-systems**

proof-techniques $\rightarrow$ **Peano Systems** $\rightarrow$ Identity, Equality, and Equivalence

## 3.1 Presburger Arithmetic

**Definition (Presburger Signature)**

**Definition 3.1.1** (Presburger Signature). The *Presburger signature* is the first-order signature

$$\mathcal{L}_{\text{Pres}} = \{0, S, +, <, =\},$$

where $0$ is a constant symbol, $S$ is a unary function symbol, $+$ is a binary function symbol, and $<$ and $=$ are binary relation symbols.

*Remark* (Standard quantified statement).

$$\text{PresburgerSignature}(\mathcal{L}) \iff \mathcal{L} = \{0, S, +, <, =\} \wedge \text{ar}(S) = 1 \wedge \text{ar}(+) = 2 \wedge \text{ar}(<) = 2 \wedge \text{ar}(=) = 2.$$

*Remark* (Interpretation). The Presburger signature names addition and order, but it deliberately omits multiplication as a function symbol. This makes Presburger arithmetic a theory of the additive structure of the natural numbers rather than a theory of full arithmetic.

**Definition (Presburger Arithmetic)**

**Definition 3.1.2** (Presburger Arithmetic). *Presburger arithmetic* is the first-order theory of the natural numbers in the Presburger signature. Its intended structure is

$$(\mathbb{N}, 0, S, +, <, =),$$

and its axioms describe the behavior of zero, successor, addition, order, and induction for formulas written in the language $\mathcal{L}_{\mathrm{Pres}}$.

*Remark* (Standard quantified statement).

$$\mathrm{PresburgerArithmetic}(T) \iff T \subseteq \mathrm{Sentences}(\mathcal{L}_{\mathrm{Pres}}) \wedge T \text{ axiomatizes } (\mathbb{N}, 0, S, +, <, =).$$

*Remark* (Interpretation). Presburger arithmetic isolates the part of arithmetic governed by addition. It is strong enough to express linear equations, congruences, and order properties of natural numbers, but it cannot directly express multiplication as a primitive operation. The later Peano-system material supplies the more general recursive framework from which addition, multiplication, order, and induction are developed structurally.

## 3.2 Defining Peano Systems

**defining-peano-systems**

Peano Systems → **Peano Axioms** → Recursion on Peano Systems

### The Peano Axioms

The Peano axioms make precise what it means for a system to behave like the natural numbers. They isolate the first element, the successor map, and the induction principle. These axioms do not define addition, multiplication, or order. They fix the structure that supports those later constructions. The formulation used here belongs to the Dedekind–Peano line of axiomatizing the natural-number structure [Dedekind, 1901, Peano, 1889].

**Definition (Peano System)**

**Definition 3.2.1** (Peano System). A triple $(P, S, 1)$ is called a *Peano system* exactly when $P$ is a set, $1 \in P$, $S : P \to P$ is a function, and the following axioms hold:

**P1.** $1 \in P$.

**P2.** For every $n \in P$, $S(n) \in P$.

**P3.** For every $n \in P$, $S(n) \neq 1$.

**P4.** For all $m, n \in P$, if $S(m) = S(n)$ then $m = n$.

**P5.** If $A \subseteq P$, $1 \in A$, and

$$\forall n \in P\ (n \in A \Longrightarrow S(n) \in A),$$

then $A = P$.

*Remark* (Standard quantified statement).

$$\begin{aligned}
&(P, S, 1) \text{ is a Peano system} \\
&\Longleftrightarrow \Big(P \text{ is a set} \wedge 1 \in P \wedge S : P \to P \\
&\qquad \wedge \forall n \in P\ (S(n) \in P) \\
&\qquad \wedge \forall n \in P\ (S(n) \neq 1) \\
&\qquad \wedge \forall m, n \in P\ (S(m) = S(n) \Longrightarrow m = n) \\
&\qquad \wedge \forall A \subseteq P\ \big((1 \in A \wedge \forall n \in P\ (n \in A \Longrightarrow S(n) \in A)) \Longrightarrow A = P\big)\Big).
\end{aligned}$$

*Remark* (Failure modes). A triple $(P, S, 1)$ fails to be a Peano system if the distinguished element does not lie in the underlying set, if the successor map leaves the set, if the successor hits $1$, if the successor identifies two distinct elements, or if there exists a proper subset of $P$ that contains $1$ and is closed under successor.

*Remark* (Failure mode decomposition).

$$\begin{aligned}
\neg\,(P, S, 1) \text{ is a Peano system} \Longrightarrow\ & 1 \notin P \\
& \vee\, \exists n \in P\ (S(n) \notin P) \\
& \vee\, \exists n \in P\ (S(n) = 1) \\
& \vee\, \exists m, n \in P\ (m \neq n \wedge S(m) = S(n)) \\
& \vee\, \exists A \subseteq P\ \big(1 \in A \wedge \forall n \in P\ (n \in A \Longrightarrow S(n) \in A) \wedge A \neq P\big).
\end{aligned}$$

*Remark* (Interpretation). A Peano system is the abstract data of counting: an underlying set, a first element, and a successor operation satisfying the usual non-collision and induction principles. The purpose of the definition is to isolate the ambient object before the individual axioms are recorded and used one by one.

*Remark* (Source alignment). The five displayed clauses are the house-style Peano-system presentation of Peano's axiomatization of arithmetic [Peano, 1889]. They are stated here in structural form so that later definitions and proofs can refer directly to the successor and induction clauses.

*Remark* (Comparison with Feferman). This definition corresponds closely to Definition 3.1 in Feferman [1964], where Peano systems are introduced as the abstract structural framework underlying the positive integers. The present notes expand the definition into explicit quantified, failure-mode, and decomposition forms in order to support later logical analysis and dependency extraction.

*Remark* (Dependencies).

- Existence of a Peano System

**Axiom (Base Element Lies in the Underlying Set)**

**Axiom 3.2.2** (Distinguished Element Lies in the Underlying Set).

$$1 \in P.$$

*Remark* (Standard quantified statement).

$$1 \in P.$$

*Remark* (Interpretation). The counting system begins from a distinguished first element of the underlying set. This is the base point from which the remaining elements are generated by the successor operation.

**Axiom (Closure Under Successor)**

**Axiom 3.2.3** (Closure Under Successor). If $n \in P$, then $S(n) \in P$.

*Remark* (Standard quantified statement).

$$\forall n \in P\ (S(n) \in P).$$

*Remark* (Interpretation). Applying the successor operation to an element of the underlying set never leaves that set. Successor is therefore an internal operation on the counting system.

**Axiom (Base Element Is Not a Successor)**

**Axiom 3.2.4** (Distinguished Element Is Not a Successor). For every $n \in P$, $S(n) \neq 1$.

*Remark* (Standard quantified statement).

$$\forall n \in P\ (S(n) \neq 1).$$

*Remark* (Interpretation). The first natural number does not arise by taking the successor of an earlier natural number. This keeps the counting system from folding backward onto its starting point.

**Axiom (Injectivity of Successor)**

**Axiom 3.2.5** (Injectivity of Successor). If $m, n \in P$ and $S(m) = S(n)$, then $m = n$.

*Remark* (Standard quantified statement).

$$\forall m, n \in P\ (S(m) = S(n) \Longrightarrow m = n).$$

*Remark* (Interpretation). Different natural numbers do not collapse to the same successor. The successor operation preserves distinctness and lets predecessor reasoning work whenever a successor is known.

**Axiom (Induction Axiom)**

**Axiom 3.2.6** (Induction Axiom). Let $A \subseteq P$. If $1 \in A$ and if

$$\forall n \in P\ (n \in A \Longrightarrow S(n) \in A),$$

then $A = P$.

*Remark* (Standard quantified statement).

$$\forall A \subseteq P\ \Big(\big(1 \in A \wedge \forall n \in P\ (n \in A \Longrightarrow S(n) \in A)\big) \Longrightarrow A = P\Big).$$

*Remark* (Interpretation). Any subset of the underlying set that contains the distinguished element and is closed under successor must already contain the entire set. This axiom rules out rogue elements and is the logical source of mathematical induction.

**Axiom (Existence of a Peano System)**

**Axiom 3.2.7** (Existence of a Peano System). There exists a Peano system.

*Remark* (Standard quantified statement).

$$\exists P\ \exists S\ \big((P, S, 1) \text{ is a Peano system}\big).$$

*Remark* (Interpretation). The Peano axioms are not merely a formal pattern; at least one structure satisfies them. This axiom guarantees that the abstract counting system is available before we give it the conventional name $\mathbb{N}$.

*Remark* (Historical note). This axiom corresponds to Axiom 3.2 of Feferman [1964], where the existence of at least one Peano system is taken as a foundational assumption.

*Remark* (Dependencies).

- Language

## First Theorems on Peano Systems

### Induction as a Working Principle

The induction axiom for a Peano system is formulated in terms of inductive subsets of the underlying set. Arithmetic arguments are usually written in terms of predicates rather than subsets. This theorem identifies the predicate-form induction principle derived from the Peano axioms. It is the standard form of induction used in the rest of the chapter. Historically, this is the working proof principle attached to Peano's axiomatization of the natural numbers [Peano, 1889].

**Theorem (Induction Principle for a Peano System)**

**Theorem 3.2.8** (Induction Principle for a Peano System). *Let* $(P, S, 1)$ *be a Peano system, and let* $\varphi$ *be a predicate on* $P$. *If*

$$\varphi(1) \quad \textit{and} \quad \forall n \in P\ (\varphi(n) \Longrightarrow \varphi(S(n))),$$

*then*

$$\forall n \in P\ \varphi(n).$$

*Go to proof.*

**Lean 4 Verification Record**

**Source anchor:** Peano, *Arithmetices principia*; Feferman, *The Number Systems*, Section 3.1.
**Formalizes:** Theorem 3.2.8.
**Lean module:** `LRA.VolumeII.PeanoSystems.Induction`.
**Lean declaration:** `induction_principle`.
**Status:** checked against `lra-lean`.

```
theorem induction_principle
    (ps : PeanoSystem)
    (predicate : ps.carrier -> Prop)
    (base_case : predicate ps.one)
    (successor_step :
      forall element : ps.carrier,
        predicate element ->
        predicate (ps.successor element)) :
    forall element : ps.carrier,
      predicate element :=
  ps.induction predicate base_case successor_step
```

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system and let $\varphi$ be a predicate on $P$.

$$\Big(\varphi(1) \wedge \forall n \in P\ (\varphi(n) \Longrightarrow \varphi(S(n)))\Big) \Longrightarrow \forall n \in P\ \varphi(n).$$

*Remark* (Contrapositive quantified statement).

Let $(P, S, 1)$ be a Peano system and let $\varphi$ be a predicate on $P$.

$$\neg\forall n \in P\ \varphi(n) \Longrightarrow \neg\Big(\varphi(1) \wedge \forall n \in P\ (\varphi(n) \Longrightarrow \varphi(S(n)))\Big).$$

*Remark* (Interpretation). This theorem converts the subset form of induction into the predicate form used in ordinary arithmetic arguments. A property propagates through the entire Peano system once it holds at the distinguished first element and is preserved by successor. The theorem therefore turns the structural induction axiom into a proof principle for arbitrary predicates on $P$.

*Remark* (Comparison with Feferman). Peano's axiomatization includes induction as part of the natural-number structure [Peano, 1889]. Feferman derives the working form of mathematical induction from the subset formulation of the Peano axioms immediately after Theorem 3.3 in Chapter 3 of Feferman [1964]. The present notes formalize that induction principle as an explicit reusable theorem.

*Remark* (Dependencies).

- Peano System
- Induction Axiom

### Strong Induction

Ordinary induction propagates a property from each element to its immediate successor. Strong induction allows the inductive step to use the property at every predecessor, not just the immediate one. This broader hypothesis makes strong induction convenient when the successor step needs information from the entire earlier segment of the Peano system.

**Theorem (Strong Induction on a Peano System)**

**Theorem 3.2.9** (Strong Induction on a Peano System). *Let $(P, S, 1)$ be a Peano system, and let $\varphi$ be a predicate on $P$. If*

$$\forall n \in P \left(\forall k \in P \left(k < n \Longrightarrow \varphi(k)\right)\right) \Longrightarrow \varphi(n),$$

*then*

$$\forall n \in P \; \varphi(n).$$

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system and let $\varphi$ be a predicate on $P$.

$$\left(\forall n \in P \left(\forall k \in P \left(k < n \Longrightarrow \varphi(k)\right)\right) \Longrightarrow \varphi(n)\right) \Longrightarrow \forall n \in P \; \varphi(n).$$

*Remark* (Interpretation). Strong induction grants the inductive step access to the entire history below $n$, not just the single predecessor. It is the proof form behind least counterexample arguments.

*Remark* (Dependencies).

- Induction Principle for a Peano System
- Strict Less Than on a Peano System
- Trichotomy on a Peano System

### Inductive Subsets and Predecessors

Induction is expressed in terms of subsets of a Peano system. Two structural notions appear repeatedly in that setting: closure under successor and membership generated from the distinguished first element. Predecessor language is also convenient once it is known that every non-initial element arises as a successor.

**Definition (Successor-Closed Subset)**

**Definition 3.2.10** (Successor-Closed Subset). Let $(P, S, 1)$ be a Peano system, and let $A \subseteq P$. The subset $A$ is called *successor-closed* exactly when

$$\forall n \in P\ (n \in A \Longrightarrow S(n) \in A).$$

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } A \subseteq P.\\ &A \text{ is successor-closed} \Longleftrightarrow \forall n \in P\ (n \in A \Longrightarrow S(n) \in A).\end{aligned}$$

*Remark* (Negated quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } A \subseteq P.\\ &A \text{ is not successor-closed} \Longleftrightarrow \exists n \in P\ (n \in A \wedge S(n) \notin A).\end{aligned}$$

*Remark* (Failure modes). A subset fails to be successor-closed exactly when it contains some element of $P$ but fails to contain that element's successor.

*Remark* (Failure mode decomposition).

$$A \text{ is not successor-closed} \Longleftrightarrow \exists n \in P\ \underbrace{(n \in A \wedge S(n) \notin A)}_{\text{closure fails at } n}.$$

*Remark* (Interpretation). A successor-closed subset is stable under one step of counting. Once an element of $A$ is present, its successor remains in $A$ as well.

*Remark* (Comparison with Feferman). The notion of a successor-closed subset is implicit in condition (iii) of Feferman's Definition 3.1 [Feferman, 1964]. The present notes isolate it as a separate definition so that induction arguments can refer to the closure condition directly.

*Remark* (Dependencies).

- Peano System

**Definition (Inductive Subset of a Peano System)**

**Definition 3.2.11** (Inductive Subset of a Peano System). Let $(P, S, 1)$ be a Peano system, and let $A \subseteq P$. The subset $A$ is called *inductive* exactly when $1 \in A$ and $A$ is successor-closed.

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } A \subseteq P.\\ &A \text{ is inductive} \iff \big(1 \in A \wedge \forall n \in P\ (n \in A \implies S(n) \in A)\big).\end{aligned}$$

*Remark* (Negated quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } A \subseteq P.\\ &A \text{ is not inductive} \iff \big(1 \notin A \vee \exists n \in P\ (n \in A \wedge S(n) \notin A)\big).\end{aligned}$$

*Remark* (Failure modes). A subset fails to be inductive in exactly two ways: it may fail to contain the distinguished first element, or it may contain some element while failing to contain that element's successor.

*Remark* (Failure mode decomposition).

$$\begin{aligned}A \text{ is not inductive} \iff\ & \underbrace{1 \notin A}_{\text{base point missing}}\\ & \vee \underbrace{\exists n \in P\ (n \in A \wedge S(n) \notin A)}_{\text{successor closure fails}}.\end{aligned}$$

*Remark* (Interpretation). An inductive subset contains the distinguished first element and is closed under successor. Such a subset contains every stage obtained by starting at $1$ and applying successor repeatedly.

*Remark* (Comparison with Feferman). The notion of an inductive subset packages the two hypotheses appearing in the induction clause of Feferman's Definition 3.1 [Feferman, 1964]. The present notes name this package so that the internal structure of induction proofs remains explicit.

*Remark* (Dependencies).

- Peano System
- Successor-Closed Subset

**Theorem (Minimality of a Peano System)**

**Theorem 3.2.12** (Minimality of a Peano System). *Let $(P, S, 1)$ be a Peano system. If $A \subseteq P$ contains $1$ and is closed under successor, then $A = P$.*

$$\big(1 \in A \wedge \forall n \in P\ (n \in A \implies S(n) \in A)\big) \implies A = P.$$

*Go to proof.*

*Remark* (Interpretation). A Peano system has no elements outside the successor chain generated by its base element. Any subset that contains the base element and is closed under successor already exhausts the whole carrier.

*Remark* (Dependencies).

- Induction Axiom
- Inductive Subset of a Peano System

**Definition (Predecessor in a Peano System)**

**Definition 3.2.13** (Predecessor in a Peano System). Let $(P, S, 1)$ be a Peano system, and let $x, u \in P$. The element $u$ is called a *predecessor* of $x$ exactly when

$$S(u) = x.$$

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system and let } x, u \in P.$$
$$u \text{ is a predecessor of } x \iff S(u) = x.$$

*Remark* (Negated quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system and let } x, u \in P.$$
$$u \text{ is not a predecessor of } x \iff S(u) \neq x.$$

*Remark* (Interpretation). A predecessor of $x$ is an element whose successor is $x$. This language isolates the inverse-looking question attached to the successor map without assuming that successor is globally invertible.

*Remark* (Dependencies).

- Peano System

### Generation by Successor

The Peano axioms describe a counting system generated from a distinguished first element by the successor operation. The next theorems make that generation principle explicit: every element is either the first element or a successor, every non-initial element has a unique predecessor, and no element can be its own successor.

**Theorem (Every Element Is Either 1 or a Successor)**

**Theorem 3.2.14** (Every Element Is Either 1 or a Successor). *Let $(P, S, 1)$ be a Peano system. For every $x \in P$, either $x = 1$ or there exists $u \in P$ such that $S(u) = x$. Go to proof.*

**Lean 4 Verification Record**

**Source anchor:** Landau, *Foundations of Analysis*, Section 1; Feferman, *The Number Systems*, Theorem 3.3.
**Formalizes:** Theorem 3.2.14.
**Lean module:** `LRA.VolumeII.PeanoSystems.BasicTheorems`.
**Lean declaration:** `every_element_is_one_or_successor`.
**Status:** checked against `lra-lean`.

```
theorem every_element_is_one_or_successor
    (ps : PeanoSystem) :
    forall element : ps.carrier,
      Or (element = ps.one)
        (Exists (fun predecessor : ps.carrier =>
          ps.successor predecessor = element)) := by
  let D : ps.carrier -> Prop :=
    fun candidate_element =>
      Or (candidate_element = ps.one)
        (Exists (fun predecessor : ps.carrier =>
          ps.successor predecessor = candidate_element))

  apply induction_principle ps D

  case base_case =>
    exact Or.inl rfl

  case successor_step =>
    intro element induction_hypothesis
    exact Or.inr (Exists.intro element rfl)
```

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall x \in P \left(x = 1 \vee \exists u \in P \left(S(u) = x\right)\right).$$

*Remark* (Interpretation). Every element of a Peano system is reached by starting at the distinguished first element and moving forward by successor. The only possible exception to having a predecessor is the element 1 itself.

*Remark* (Historical note). This theorem is one of the basic successor consequences of Peano's axiomatization [Peano, 1889]. In the present organization it corresponds directly to Theorem 3.3 in Feferman [1964]. Feferman uses this result to show that the distinguished initial element is the only possible non-successor in a Peano system.

*Remark* (Dependencies).

- Peano System
- Distinguished Element Lies in the Underlying Set

- Closure Under Successor
- Induction Principle for a Peano System

**Theorem (Successor Inequality Reflection)**

**Theorem 3.2.15** (Successor Inequality Reflection). *Let $(P, S, 1)$ be a Peano system. For all $a, b \in P$, if $a \neq b$, then $S(a) \neq S(b)$. Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b \in P\ (a \neq b \Longrightarrow S(a) \neq S(b)).$$

*Remark* (Negated quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\exists a, b \in P\ (a \neq b \wedge S(a) = S(b)).$$

*Remark* (Failure modes). Successor inequality reflection fails exactly when two distinct elements have the same successor.

*Remark* (Failure mode decomposition).

$$\exists a, b \in P \qquad \underbrace{(a \neq b \wedge S(a) = S(b))}_{\text{distinct elements with a common successor}} \qquad .$$

*Remark* (Contrapositive quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b \in P\ (S(a) = S(b) \Longrightarrow a = b).$$

*Remark* (Interpretation). The successor operation reflects inequality: distinct inputs remain distinct after one successor step. This is the contrapositive form of successor injectivity, and it is the form used when an induction step must preserve a negative equality statement.

*Remark* (Dependencies).

- Peano System
- Injectivity of Successor

**Corollary (Non-One Elements Have a Predecessor)**

**Corollary 3.2.16** (Non-One Elements Have a Predecessor). *Let $(P, S, 1)$ be a Peano system. For every $x \in P$, if $x \neq 1$, then there exists $u \in P$ such that $S(u) = x$. Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall x \in P\ \big(x \neq 1 \Longrightarrow \exists u \in P\ (S(u) = x)\big).$$

*Remark* (Negated quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\exists x \in P \left(x \neq 1 \wedge \forall u \in P \left(S(u) \neq x\right)\right).$$

*Remark* (Failure modes). The statement fails exactly when some element distinct from 1 has no predecessor.

*Remark* (Failure mode decomposition).

$$\exists x \in P \quad \underbrace{\left(x \neq 1 \wedge \forall u \in P \left(S(u) \neq x\right)\right)}_{\text{non-initial element with no predecessor}} .$$

*Remark* (Interpretation). Every non-initial element of a Peano system is reached by one successor step from an earlier element. This separates predecessor existence from predecessor uniqueness so that later arguments can cite only the part they need.

*Remark* (Comparison with Feferman). Feferman's Theorem 3.3 in Feferman [1964] contains this predecessor-existence fact as part of the structural analysis of Peano systems. The present notes record it separately before adding uniqueness.

*Remark* (Dependencies).

- Peano System
- Every Element Is Either 1 or a Successor

**Corollary (Predecessor Existence and Uniqueness Away from 1)**

**Corollary 3.2.17** (Predecessor Existence and Uniqueness Away from 1). *Let $(P, S, 1)$ be a Peano system, and let $x \in P$ with $x \neq 1$. Then there exists a unique $u \in P$ such that $S(u) = x$. Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall x \in P \left(x \neq 1 \Longrightarrow \exists! u \in P \left(S(u) = x\right)\right).$$

*Remark* (Interpretation). Every element distinct from 1 has exactly one predecessor. The successor map therefore organizes the non-initial part of a Peano system into one-step extensions of earlier elements.

*Remark* (Comparison with Feferman). Feferman proves predecessor existence and uniqueness together inside Theorem 3.3 of Feferman [1964]. The present notes separate the uniqueness statement into an independent corollary so that it can be cited directly in later proofs.

*Remark* (Dependencies).

- Peano System

- Non-One Elements Have a Predecessor
- Injectivity of Successor

**Corollary (Unique Predecessor Characterization Away from 1)**

**Corollary 3.2.18** (Unique Predecessor Characterization Away from 1). *Let $(P, S, 1)$ be a Peano system. For every $x \in P$, $x \neq 1$ if and only if there exists a unique $u \in P$ such that $S(u) = x$. Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\&\forall x \in P \left(x \neq 1 \iff \exists! u \in P\ (S(u) = x)\right).\end{aligned}$$

*Remark* (Negated quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\&\exists x \in P \Big(\big(x \neq 1 \wedge (\forall u \in P\ (S(u) \neq x)\\&\qquad \vee \exists u, v \in P\ (S(u) = x \wedge S(v) = x \wedge u \neq v)\big)\\&\qquad \vee \big(x = 1 \wedge \exists u \in P\ (S(u) = x \wedge \forall v \in P\ (S(v) = x \implies v = u))\big)\Big).\end{aligned}$$

*Remark* (Failure modes). The biconditional can fail in two ways: a non-initial element can fail to have a unique predecessor, or the distinguished first element can have a unique predecessor.

*Remark* (Failure mode decomposition).

$$\begin{aligned}\exists x \in P \Big(&\underbrace{\big(x \neq 1 \wedge (\forall u \in P\ (S(u) \neq x) \vee \exists u, v \in P\ (S(u) = x \wedge S(v) = x \wedge u \neq v)\big)}_{\text{non-initial element lacks a unique predecessor}}\\&\vee \underbrace{\big(x = 1 \wedge \exists u \in P\ (S(u) = x \wedge \forall v \in P\ (S(v) = x \implies v = u))\big)}_{\text{the first element has a unique predecessor}}\Big).\end{aligned}$$

*Remark* (Interpretation). Being different from the distinguished first element is exactly the same as having a unique predecessor. This turns the informal phrase "away from 1" into an internal characterization of the successor structure.

*Remark* (Comparison with Feferman). This corollary packages the predecessor analysis surrounding Feferman's Theorem 3.3 in Feferman [1964]. The forward direction is the predecessor existence-and-uniqueness result, while the reverse direction uses the axiom that 1 is not a successor.

*Remark* (Dependencies).

- Peano System
- Distinguished Element Is Not a Successor

- Predecessor Existence and Uniqueness Away from 1

**Corollary (The Distinguished Element Is the Unique Non-Successor)**

**Corollary 3.2.19** (The Distinguished Element Is the Unique Non-Successor). *Let $(P, S, 1)$ be a Peano system. An element $x \in P$ is not a successor of any element of $P$ if and only if $x = 1$. Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall x \in P \left(\neg \exists u \in P\, (S(u) = x) \iff x = 1\right).$$

*Remark* (Interpretation). The distinguished first element is characterized internally as the unique element with no predecessor. This gives a structural description of 1 that does not depend on external notation.

*Remark* (Comparison with Feferman). This corollary extracts a structural characterization of the distinguished initial element that is implicit in the discussion surrounding Theorem 3.3 of Feferman [1964].

*Remark* (Dependencies).

- Distinguished Element Is Not a Successor
- Predecessor Existence and Uniqueness Away from 1

**Theorem (No Object Is Its Own Successor)**

**Theorem 3.2.20** (No Object Is Its Own Successor). *Let $(P, S, 1)$ be a Peano system. For every $n \in P$,*

$$S(n) \neq n.$$

*Go to proof.*

**Lean 4 Verification Record**

**Source anchor:** Landau, *Foundations of Analysis*, Section 2, Theorem 2.
**Formalizes:** Theorem 3.2.20.
**Lean module:** `LRA.VolumeII.PeanoSystems.BasicTheorems`.
**Lean declaration:** `successor_not_self`.
**Status:** checked against `lra-lean`.

```
theorem successor_not_self
    (ps : PeanoSystem) :
    forall element : ps.carrier,
      Not (ps.successor element = element) := by
  apply induction_principle ps

  case base_case =>
    exact ps.one_not_successor ps.one

  case successor_step =>
    intro element induction_hypothesis
    exact
      successor_preserves_inequality
        ps
        (ps.successor element)
        element
        induction_hypothesis
```

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall n \in P\ (S(n) \neq n).$$

*Remark* (Interpretation). Successor always moves forward in the counting structure. No element can be recovered by applying successor to itself.

*Remark* (Historical note). This theorem is a standard consequence of the Peano axioms in the classical development of the natural numbers [Peano, 1889]. Feferman includes the corresponding result as an exercise in Chapter 3 of Feferman [1964].

*Remark* (Dependencies).

- Distinguished Element Is Not a Successor
- Injectivity of Successor
- Induction Principle for a Peano System

## Peano System Figure Plates

The following figure plates collect the visual diagrams for the Peano system and iterator material. Each plate is presented separately so that the diagram can be read as a full-page reference.

*Model-Theoretic Signature $\Sigma = (P; 1; S, +, \cdot, \hat{}\,)$*

SORT P — the carrier set (the Peano system)

One sort only. All function symbols have domain and codomain in P.

CONSTANT 1 : P — the distinguished initial element

Axiom: $1 \notin \mathrm{rng}(S)$ (1 is not a successor)

FUNCTION SYMBOLS

| Symbol | Type | Arity | Base clause | Successor clause |
|---|---|---|---|---|
| S | $P \to P$ | 1 | — | $S(n) \neq S(m) \to n \neq m$ (injective) |
| + | $P \times P \to P$ | 2 | $m + 1 = S(m)$ | $m + S(n) = S(m + n)$ |
| · | $P \times P \to P$ | 2 | $m \cdot 1 = m$ | $m \cdot S(n) = (m \cdot n) + m$ |
| ^ | $P \times P \to P$ | 2 | $m \hat{}\, 1 = m$ | $m \hat{}\, S(n) = (m \hat{}\, n) \cdot m$ |

Dependency: S (primitive) → + → · → ^ (each step rule uses the previous operation)

PEANO AXIOMS (for S and 1)

P1: $1 \in P$

P2: $S : P \to P$

P3: $1 \notin \mathrm{rng}(S)$

P4: S injective

P5: Induction (second-order)

ALGEBRAIC LAWS (to be proved)

Commutativity: $m + n = n + m$

Associativity: $(m+n)+k = m+(n+k)$

Distributivity: $m \cdot (n+k) = m \cdot n + m \cdot k$

Cancellation: $m+k=n+k \to m=n$

Exp laws: $m\hat{}(n+k) = m\hat{}n \cdot m\hat{}k$

Categoricity: any two Σ-structures satisfying P1–P5 are uniquely isomorphic (Uniqueness of Peano Systems up to Isomorphism)

Figure 3.1: The model-theoretic signature for a Peano system and the later arithmetic operations generated from it.

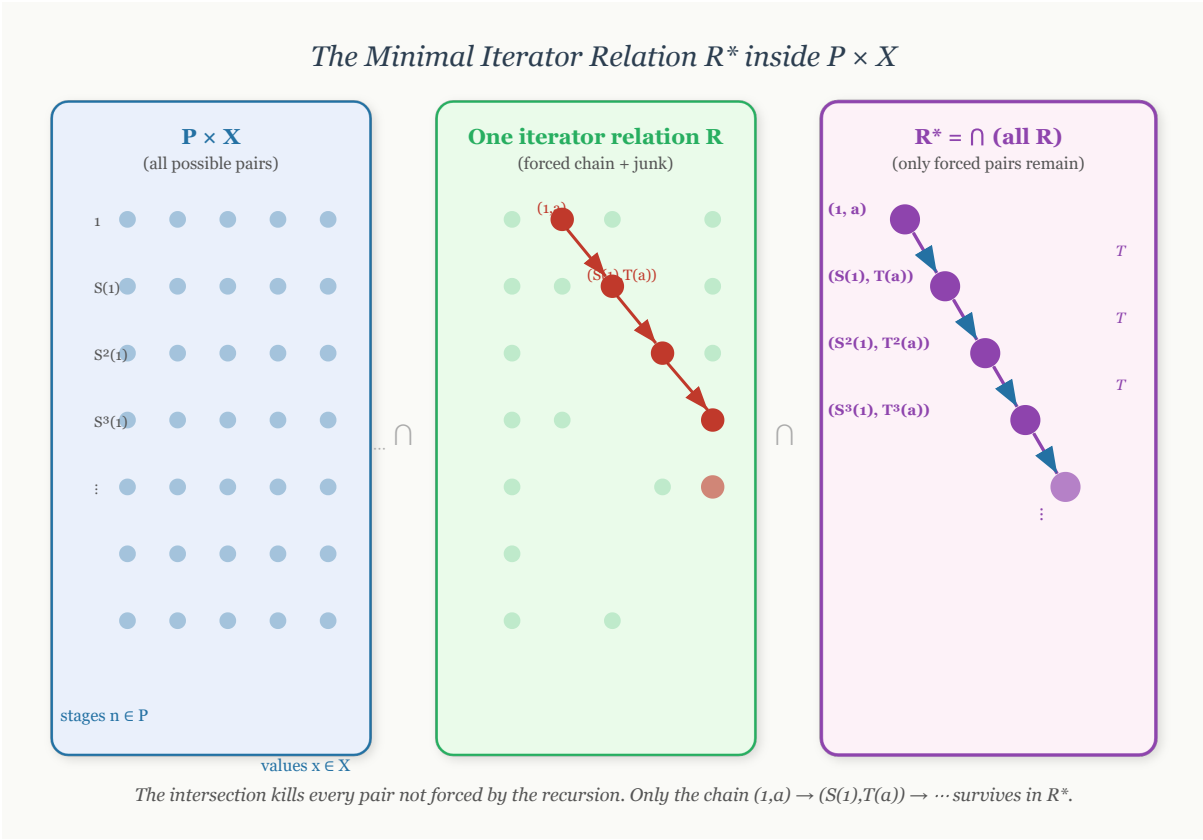



Figure 3.2: The intuitive construction of the minimal iterator relation $R^*$ inside $P \times W$.

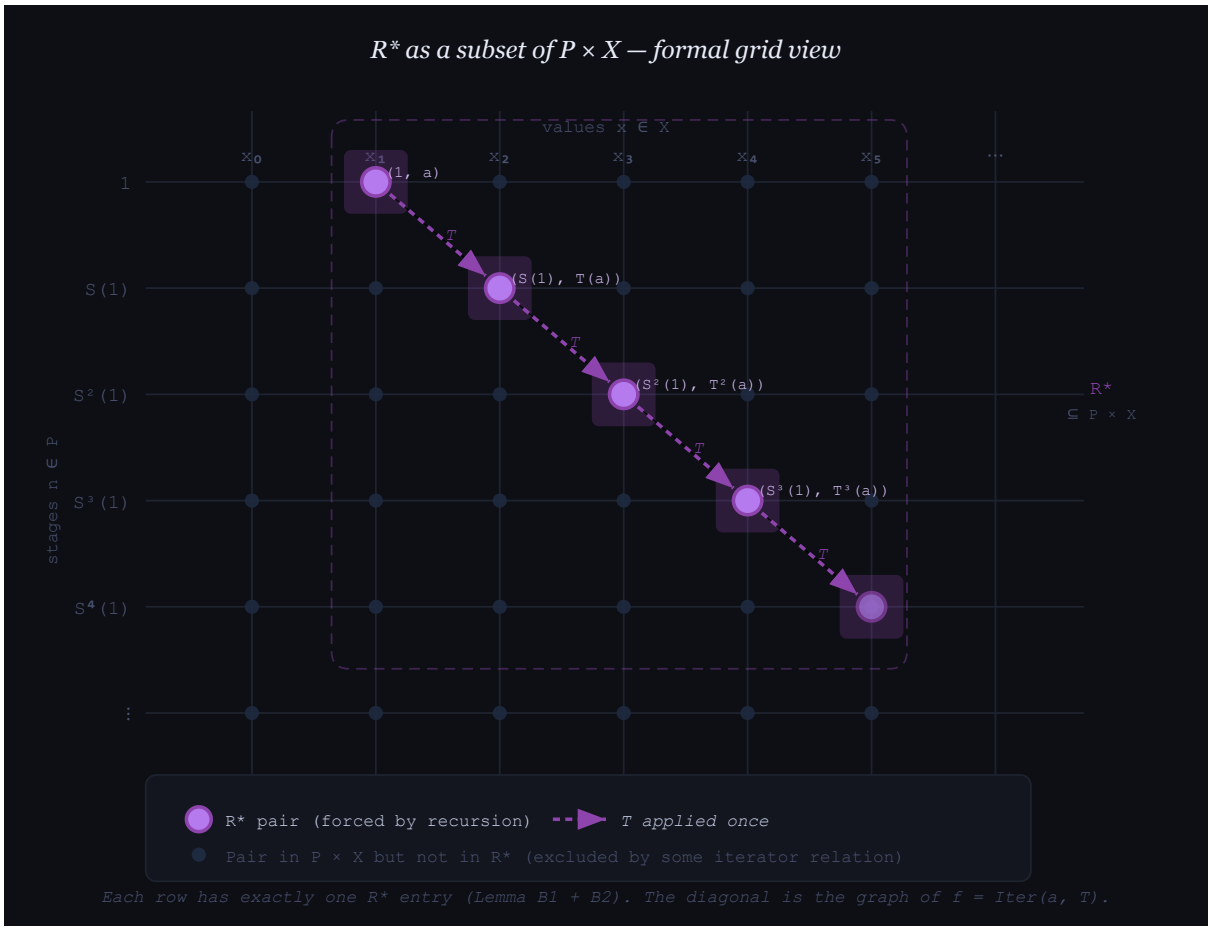



Figure 3.3: A formal grid view of $R^*$ as the chain of forced ordered pairs in $P \times W$.

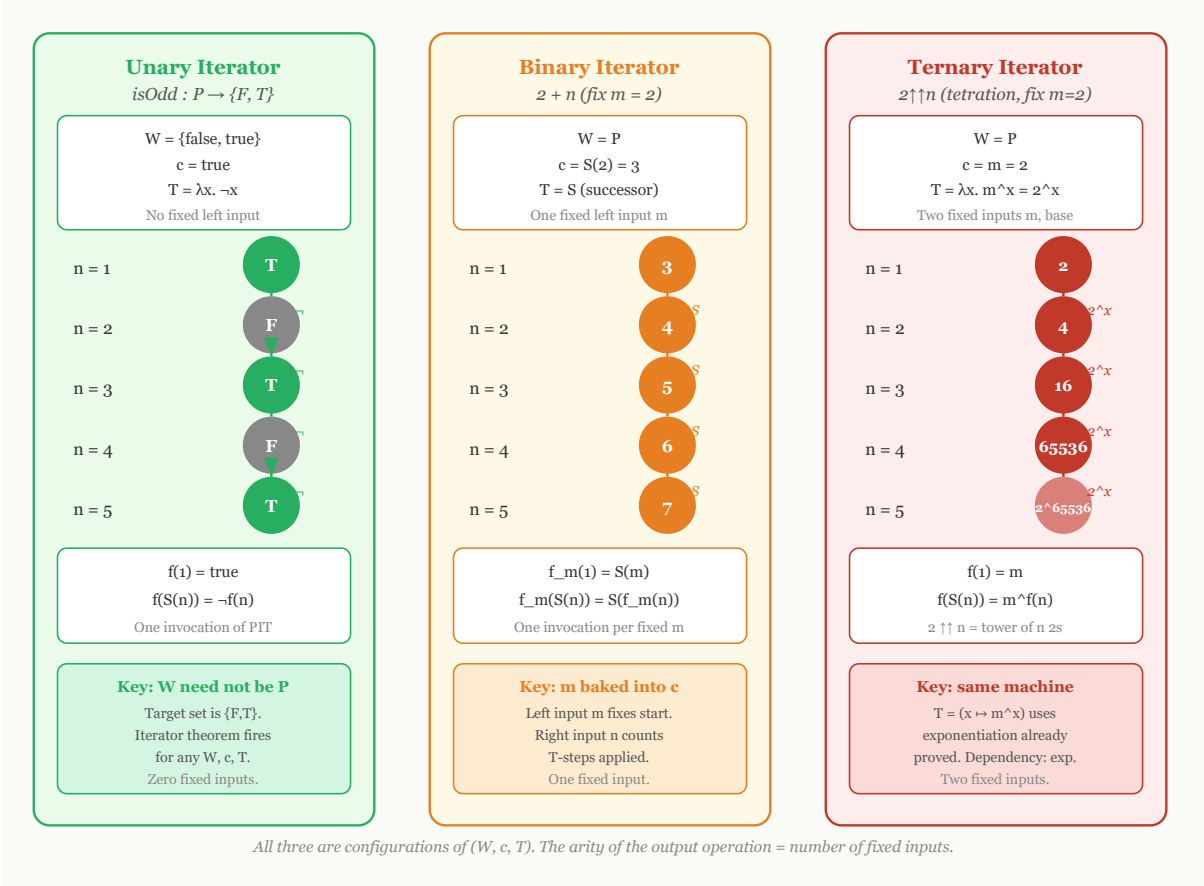



Figure 3.4: Unary, binary, and higher-arity iterator configurations after fixing the relevant parameters.

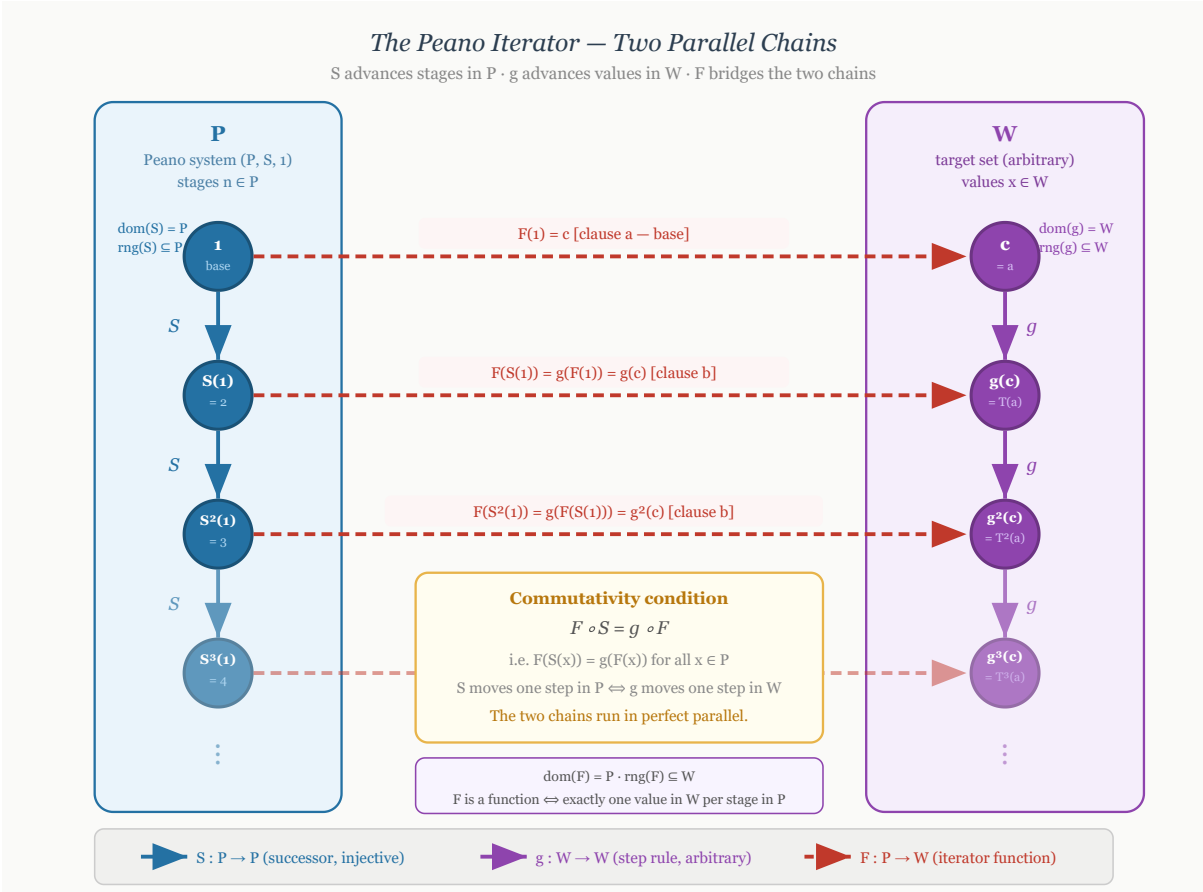



Figure 3.5: The Peano iterator as two parallel chains, one in $P$ and one in $W$, connected by the iterator function.

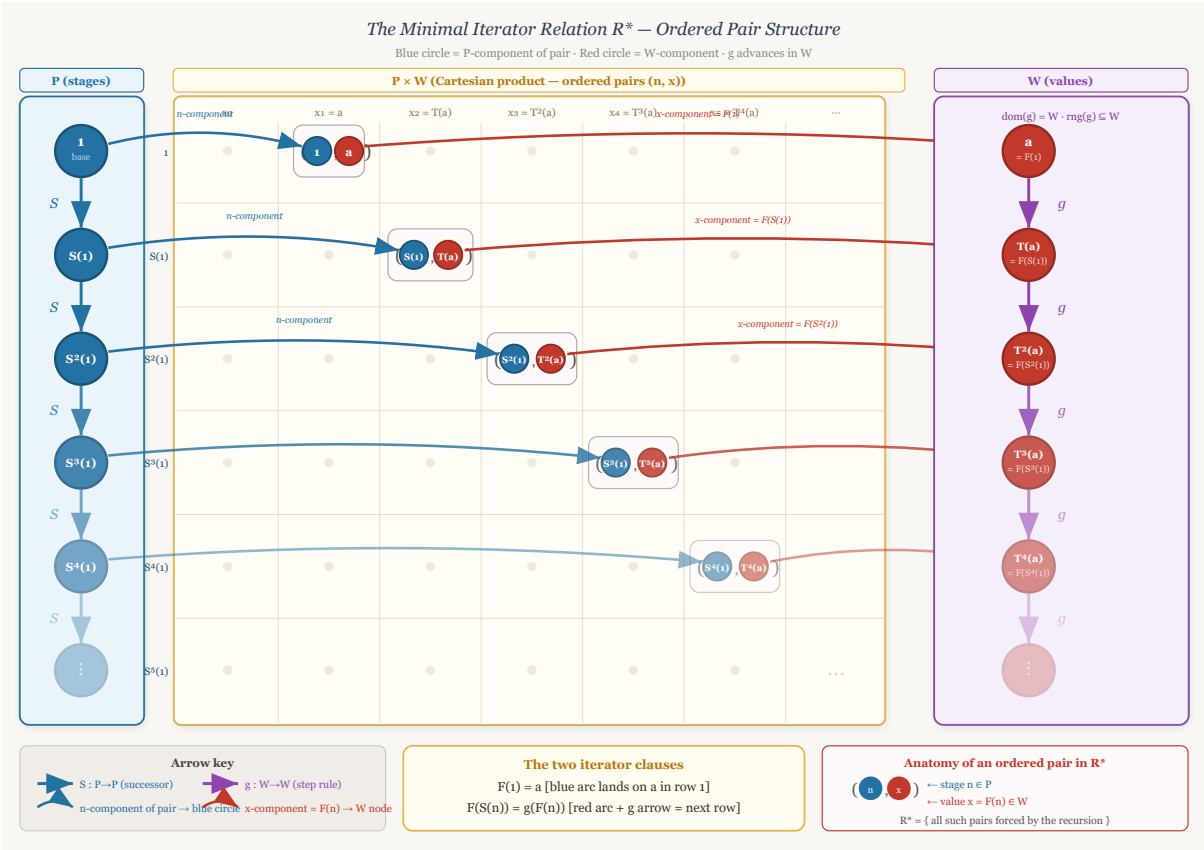



Figure 3.6: The three-zone view of $R^*$, separating stages in $P$, ordered pairs in $P \times W$, and values in $W$.

## 3.3 Examples of Peano Systems

**examples-of-peano-systems**

Peano Systems Defining Peano Systems → **Examples of Peano Systems** → Recursion on Peano Systems

### Peano Systems as Abstract Structures

The examples in this section separate the abstract idea of a Peano system from any particular representation of the natural numbers. A Peano system is not a specific set such as $\mathbb{N}$. It is a structure consisting of a carrier set, a distinguished initial element, and a successor operation satisfying the Peano conditions. Different carriers can therefore realize the same abstract successor structure.

*Remark* (Formal reference). The formal definition used throughout this section is Peano System. In each example, the data are displayed as a triple $(P, S, e)$ consisting of a carrier $P$, a successor map $S : P \to P$, and a distinguished first element $e \in P$.

*Remark* (Interpretation). The axioms say that the system has one starting point, the successor operation never merges two different elements, and induction reaches every element of the carrier. Thus the carrier is an infinite one-way chain generated from its initial element.

## The Standard One-Based System

### The Usual One-Based Carrier

The familiar natural-number model is the reference example, but it is only one realization of the Peano axioms. The carrier is the usual one-based set of natural numbers, the distinguished element is 1, and successor is the usual next-number operation.

*Example* (The Usual Natural Numbers). Let

$$P = \mathbb{N}, \qquad e = 1,$$

and define

$$S : P \to P, \qquad S(n) = n + 1.$$

Then $(P, S, e) = (\mathbb{N}, S, 1)$ is the usual one-based Peano system.

*Remark* (Structure). The Peano data are

$$\big(P, S, e\big) = \big(\mathbb{N},\, n \mapsto n + 1,\, 1\big).$$

*Remark* (Interpretation). This is the familiar model, but it is not the only possible carrier for a Peano system. The remaining examples use different elements while preserving the same abstract successor pattern.

## Von Neumann Ordinals

### Set-Theoretic Carriers

Von Neumann ordinals realize natural-number structure inside set theory. In this representation, each stage is a set, and successor is formed by adjoining the current stage as a new element.

*Example* (The Zero-Based Von Neumann Chain). Let

$$0 = \varnothing, \qquad 1 = \{0\}, \qquad 2 = \{0, 1\}, \qquad 3 = \{0, 1, 2\}, \quad \ldots$$

Let $P = \omega$, let $e = 0$, and define

$$S : P \to P, \qquad S(n) = n \cup \{n\}.$$

Then $(P, S, e) = (\omega, S, 0)$ is a zero-based Peano system.

*Remark* (Structure). The Peano data are

$$\big(P, S, e\big) = \big(\omega,\, n \mapsto n \cup \{n\},\, 0\big).$$

This example uses the zero-based convention common in set-theoretic foundations. The house convention for arithmetic in this volume remains one-based.

*Example* (The One-Based Von Neumann Chain). Let

$$P = \omega \setminus \{0\}, \qquad e = 1,$$

and define

$$S : P \to P, \qquad S(n) = n \cup \{n\}.$$

Then $(P, S, e)$ is a one-based Peano system.

*Remark* (Structure). The Peano data are

$$\big(P, S, e\big) = \big(\omega \setminus \{0\},\, n \mapsto n \cup \{n\},\, 1\big).$$

*Remark* (Interpretation). In the Von Neumann model, each natural number is literally the set of all smaller natural numbers. This is not merely notation; it is a concrete set-theoretic realization of the abstract Peano structure.

## Nonnumeric Peano Systems

### Different Carriers, Same Successor Shape

A Peano carrier need not be the usual natural-number set, and its successor operation need not agree with the ambient order or arithmetic of a larger structure. What matters is the internally generated successor chain.

*Example* (The Powers of Ten). Let

$$P = \{10^{n-1} : n \in \mathbb{N}\}, \qquad e = 1,$$

and define

$$S : P \to P, \qquad S(10^{n-1}) = 10^n \quad (n \in \mathbb{N}).$$

Equivalently, for every $x \in P$,

$$S(x) = 10x.$$

Then $(P, S, e)$ is a Peano system.

*Remark* (Structure). The Peano data are

$$\big(P, S, e\big) = \big(\{1, 10, 10^2, 10^3, \ldots\},\, x \mapsto 10x,\, 1\big).$$

*Remark* (Interpretation). The elements are not all natural numbers. They are only powers of ten. Nevertheless, starting from $1$ and repeatedly multiplying by $10$ produces a successor chain with exactly the Peano shape:

$$1 \mapsto 10 \mapsto 10^2 \mapsto 10^3 \mapsto \cdots.$$

*Example* (Strings of Repeated Letters). Let $\Sigma = \{a\}$ and let

$$P = \{a^n : n \in \mathbb{N}\}, \qquad e = a,$$

where $a^n$ denotes the string consisting of $n$ copies of $a$. Define

$$S : P \to P, \qquad S(a^n) = a^{n+1} \quad (n \in \mathbb{N}).$$

Then $(P, S, e)$ is a Peano system.

*Remark* (Structure). The Peano data are

$$\big(P, S, e\big) = \big(\{a, aa, aaa, aaaa, \ldots\},\, a^n \mapsto a^{n+1},\, a\big).$$

*Remark* (Interpretation). This example is useful because the elements are not numbers at all. They are finite strings. The Peano structure is carried by the successor relation, not by the surface appearance of the elements.

*Example* (The Reciprocal Chain). Let

$$P = \left\{\frac{1}{n} : n \in \mathbb{N}\right\}, \qquad e = 1,$$

and define

$$S : P \to P, \qquad S\left(\frac{1}{n}\right) = \frac{1}{n+1} \quad (n \in \mathbb{N}).$$

Then $(P, S, e)$ is a Peano system.

*Remark* (Structure). The Peano data are

$$\big(P, S, e\big) = \left(\left\{1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots\right\},\, \frac{1}{n} \mapsto \frac{1}{n+1},\, 1\right).$$

*Remark* (Interpretation). This is a good example precisely because the carrier is a decreasing sequence in the usual order on $\mathbb{R}$:

$$1 > \frac{1}{2} > \frac{1}{3} > \frac{1}{4} > \cdots .$$

The Peano successor does not have to mean "larger than." It only means the next element in the generated successor chain. The order inherited from $\mathbb{R}$ is external structure, not part of the Peano-system data.

## A Finite Alphabet Is Not a Peano System

### A Failed Carrier

Finite chains help isolate what the Peano axioms forbid. A finite alphabet can support a partial next-letter operation, or it can support a cyclic successor operation, but neither choice gives a Peano system.

*Example* (The Finite Alphabet Fails). Let

$$P = \{A, B, C, \ldots, Z\}.$$

If one tries to define

$$S(A) = B, \quad S(B) = C, \quad \ldots,$$

then there is no value of $S(Z)$ inside the same finite alphabet that extends the one-way chain. If instead one defines $S(Z) = A$, then $S : P \to P$ is total but the successor operation forms a cycle.

*Remark* (Failure modes). The finite alphabet fails in one of two ways:

either $S$ is not a total map $P \to P$,
or $S$ loops back into a cycle.

The first failure violates closure of successor. The second failure violates the one-way successor structure forced by the Peano axioms.

*Remark* (Interpretation). A finite alphabet cannot be a Peano system. Peano systems are necessarily infinite because induction and successor closure generate an unending chain from the distinguished first element.

## Summary

### The Core Lesson

The carrier of a Peano system can be numerical, set-theoretic, symbolic, or embedded in another ordered structure. The common content is the same abstract successor pattern.

*Remark* (Examples). The following are Peano systems once their successor

operations are specified:

$$
\begin{aligned}
&(\mathbb{N},\, n \mapsto n+1,\, 1),\\
&(\omega,\, n \mapsto n \cup \{n\},\, 0),\\
&(\omega \setminus \{0\},\, n \mapsto n \cup \{n\},\, 1),\\
&(\{1, 10, 10^2, 10^3, \ldots\},\, x \mapsto 10x,\, 1),\\
&(\{a, aa, aaa, aaaa, \ldots\},\, a^n \mapsto a^{n+1},\, a),\\
&\left(\left\{1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots\right\},\, \frac{1}{n} \mapsto \frac{1}{n+1},\, 1\right).
\end{aligned}
$$

They look different, but they have the same abstract Peano shape. Each is an infinite chain generated from a distinguished initial element by repeated successor.

## 3.4 Recursion on Peano Systems

### Why Recursion Requires a Theorem

The Peano axioms describe the shape of a Peano system: there is a first element, a successor operation, no predecessor of the first element, no successor collisions, and an induction principle. They do not, by themselves, say that every recursive specification determines a function. A recursive display such as

$$f(1) = c \qquad \text{and} \qquad f(S(n)) = g(f(n))$$

is only a specification. It becomes a definition only after one proves that there exists a unique function satisfying it. This is the "specification is not existence" point emphasized in Feferman's treatment of recursive definition [Feferman, 1964, p. 67].
The proof used here follows the function-as-graph viewpoint. A function $f : P \to W$ is a subset of $P \times W$ whose elements are ordered pairs $(n, w)$, one for each stage $n \in P$. The construction therefore moves through three zones: the stage set $P$, the Cartesian product grid $P \times W$, and the value set $W$. The recursion theorem is the result that identifies the right graph inside $P \times W$ and proves that it is the graph of exactly one function.

*Remark* (Prerequisite definition). An *inductive subset* of a Peano system is defined in Inductive Subset of a Peano System. In this section, induction is used only through the earlier Induction Principle for a Peano System.

## Arity Examples Before the Theorem

### Iterator Configurations

The iterator theorem has one formal shape but many uses. In every example below, the stage variable is the Peano stage $n \in P$, the value set is $W$, the first value is $c$, and the update rule is $g : W \to W$. These examples are motivating configurations only; the

theorem establishing their legitimacy is proved afterward.

*Remark* (Example: isOdd).

| component | choice |
|---|---|
| $W$ | $\{\texttt{false}, \texttt{true}\}$ |
| $c$ | true |
| $g$ | $x \mapsto \neg x$ |

| $n$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $h(n)$ | true | false | true | false | true |

The defining clauses are

$$h(1) = \texttt{true} \qquad \text{and} \qquad h(S(n)) = \neg h(n).$$

This example shows that the value set $W$ need not be the Peano system $P$. The stages are natural-number-like, but the values are truth values.

*Remark* (Example: addition with the left parameter fixed). Fix the left parameter at $m = 2$.

| component | choice |
|---|---|
| $W$ | $P$ |
| $c$ | $S(2) = 3$ |
| $g$ | $S$ |

| $n$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $h(n)$ | 3 | 4 | 5 | 6 | 7 |

The defining clauses are

$$h(1) = S(2) \qquad \text{and} \qquad h(S(n)) = S(h(n)).$$

This is the iterator pattern behind $2 \quad + \quad n$ in the one-based convention. One parameter is fixed, and recursion runs in the remaining argument.

*Remark* (Example: tetration with the base fixed). Fix the base at $m = 2$.

| component | choice |
|---|---|
| $W$ | $P$ |
| $c$ | 2 |
| $g$ | $x \mapsto 2^x$ |

| $n$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $h(n)$ | 2 | $2^2$ | $2^{2^2}$ | $2^{2^{2^2}}$ | $2^{2^{2^{2^2}}}$ |

The defining clauses are

$$h(1) = 2 \qquad \text{and} \qquad h(S(n)) = 2^{h(n)}.$$

This example shows that the iterator theorem is not limited to the first arithmetic operations. Higher arithmetic still has the same formal configuration once the relevant operation on $W$ is available.

*Remark* (Example: string repetition). Fix a string $s \in \Sigma^*$.

| component | choice |
|---|---|
| $W$ | $\Sigma^*$ |
| $c$ | $s$ |
| $g$ | $x \mapsto x \cdot s$ |

| $n$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $h(n)$ | $s$ | $s \cdot s$ | $s \cdot s \cdot s$ | $s \cdot s \cdot s \cdot s$ | $s \cdot s \cdot s \cdot s \cdot s$ |

The defining clauses are

$$h(1) = s \qquad \text{and} \qquad h(S(n)) = h(n) \cdot s.$$

This example is not arithmetic. It illustrates that recursion on $P$ can generate values in any set equipped with an appropriate update rule.

## Formal Development

### Minimal Relations and Iterator Functions

The formal proof is organized in Feferman's graph style. First define the admissible relations in $P \times W$. Then take their intersection. The resulting minimal relation is shown to be consistent, deterministic, and complete, hence it is the graph of the required function. This follows the minimal-relation construction used in Feferman, Hrbacek–Jech, and related lecture-note presentations of the recursion theorem [Feferman, 1964, Hrbacek and Jech, 1999, Freiwald, 2009].

**Definition (Iterator Data)**

**Definition 3.4.1** (Iterator Data). Let $(P, S, 1)$ be a Peano system, let $W$ be a set, let $c \in W$, and let $g : W \to W$ be a function. The triple $(W, c, g)$ is called *iterator data* for $(P, S, 1)$.

*Remark* (Standard quantified statement).

$$(W, c, g) \text{ is iterator data for } (P, S, 1) \iff \big(W \text{ is a set} \wedge c \in W \wedge g : W \to W\big).$$

*Remark* (Definition predicate reading).

$$\mathrm{IteratorData}(W, c, g) \iff \big(W \text{ is a set} \wedge c \in W \wedge g : W \to W\big).$$

*Remark* (Negated quantified statement).

$$(W, c, g) \text{ is not iterator data for } (P, S, 1) \iff \big(W \text{ is not a set} \vee c \notin W \vee g \text{ is not a functi}$$

*Remark* (Failure modes). Iterator data fails if the proposed value collection is not a set, if the initial value does not lie in the value set, or if the proposed update rule is not a self-map of the value set.

*Remark* (Failure mode decomposition).

$$\begin{aligned} & (W, c, g) \text{ is not iterator data for } (P, S, 1) \\ \iff & \underbrace{W \text{ is not a set}}_{\text{no value set}} \\ & \vee \quad \underbrace{c \notin W}_{\text{invalid initial value}} \\ & \vee \underbrace{g \text{ is not a function } W \to W}_{\text{invalid update rule}}. \end{aligned}$$

*Remark* (Interpretation). Iterator data records the value set, the first value, and the fixed rule used to pass from each value to the next value.

*Remark* (Dependencies).

- Peano System

**Definition (Iterator Relation)**

**Definition 3.4.2** (Iterator Relation). Let $(P, S, 1)$ be a Peano system, and let $(W, c, g)$ be iterator data for it. A relation $R \subseteq P \times W$ is called an *iterator relation* for $(W, c, g)$ exactly when

$$(1, c) \in R$$

and

$$\forall n \in P\ \forall w \in W\ \big((n, w) \in R \Longrightarrow (S(n), g(w)) \in R\big).$$

*Remark* (Standard quantified statement).

$$R \text{ is an iterator relation for } (W, c, g) \Longleftrightarrow R \subseteq P \times W \wedge (1, c) \in R \\ \wedge \forall n \in P\ \forall w \in W\ \big((n, w) \in R \Longrightarrow (S(n), g(w)) \in$$

*Remark* (Definition predicate reading).

$$\mathrm{IteratorRelation}(R, (P, S, 1), W, c, g) \Longleftrightarrow \mathrm{Relation}(R, P, W) \wedge (1, c) \in R \\ \wedge \forall n \in P\ \forall w \in W\ \big((n, w) \in R \Longrightarrow (S(n), g(w)) \in R\big).$$

*Remark* (Negated quantified statement).

$$R \text{ is not an iterator relation for } (W, c, g) \Longleftrightarrow R \not\subseteq P \times W \vee (1, c) \notin R \\ \vee \exists n \in P\ \exists w \in W\ \big((n, w) \in R \wedge (S(n), g(w))$$

*Remark* (Failure modes). An iterator relation fails if it is not a relation inside $P \times W$, if it omits the required base pair, or if it is not closed under the iterator step.

*Remark* (Failure mode decomposition).

$$R \text{ is not an iterator relation for } (W, c, g) \Longleftrightarrow \underbrace{R \not\subseteq P \times W}_{\text{wrong ambient relation}} \\ \vee \underbrace{(1, c) \notin R}_{\text{base pair missing}} \\ \vee \underbrace{\exists n \in P\ \exists w \in W\ \big((n, w) \in R \wedge (S(n), g(w))}_{\text{step closure fails}}$$

*Remark* (Interpretation). An iterator relation is any graph-like set of possible stage-value pairs that contains the required first pair and is closed under the recursive step.

*Remark* (Dependencies).

- Iterator Data

**Definition (Minimal Iterator Relation)**

**Definition 3.4.3** (Minimal Iterator Relation). Let $(P, S, 1)$ be a Peano system, and let $(W, c, g)$ be iterator data for it. The *minimal iterator relation* for $(W, c, g)$ is

$$R^* = \bigcap \{R \subseteq P \times W : R \text{ is an iterator relation for } (W, c, g)\}.$$

**Lean 4 Verification Record**

**Source anchor:** Feferman, *The Number Systems*, Section 3.4; Mendelson, *Number Systems and the Foundations of Analysis*, Section 2.2.
**Formalizes:** Definition 3.4.3.
**Lean module:** `LRA.VolumeII.PeanoSystems.Recursion`.
**Lean declaration:** `minimal_iterator_relation`.
**Status:** checked against `lra-lean`.

```
def minimal_iterator_relation
    (ps : PeanoSystem)
    (data : IteratorData ps)
    (element : ps.carrier)
    (value : data.target) : Prop :=
  forall relation : ps.carrier -> data.target -> Prop,
    iterator_relation ps data relation ->
    relation element value
```

*Remark* (Standard quantified statement).

$$(n, w) \in R^* \iff \forall R \subseteq P \times W \left(R \text{ is an iterator relation for } (W, c, g) \implies (n, w) \in R\right).$$

*Remark* (Definition predicate reading). Let $\mathcal{C}_{\mathrm{it}}$ be the closure condition

$$\mathcal{C}_{\mathrm{it}}(R) \iff \mathrm{IteratorRelation}(R, (P, S, 1), W, c, g).$$

Then the definition says

$$\mathrm{MinimalRelation}(R^*, \mathcal{C}_{\mathrm{it}}).$$

*Remark* (Negated quantified statement).

$$(n, w) \notin R^* \iff \exists R \subseteq P \times W \left(R \text{ is an iterator relation for } (W, c, g) \wedge (n, w) \notin R\right).$$

*Remark* (Failure modes). A pair fails to belong to the minimal iterator relation exactly when some iterator relation omits that pair. Such a pair is not forced by the base and successor-closure clauses.

*Remark* (Interpretation). The minimal iterator relation consists exactly of the stage-value pairs forced by the base pair and the successor closure rule. This is the $R^*$ construction used in Feferman's proof of recursive definition, and in the related set-theoretic presentations cited here [Feferman, 1964, Hrbacek and Jech, 1999, Freiwald, 2009].

*Remark* (Dependencies).

- Iterator Relation

**Lemma (Consistency of the Minimal Iterator Relation)**

**Lemma 3.4.4** (Consistency of the Minimal Iterator Relation). *Let $(P, S, 1)$ be a Peano system, let $(W, c, g)$ be iterator data for it, and let $R^*$ be the minimal iterator relation for $(W, c, g)$. Then $R^*$ is an iterator relation for $(W, c, g)$. Go to proof.*

*Remark* (Standard quantified statement).

$$R^* \text{ is an iterator relation for } (W, c, g).$$

*Remark* (Predicate reading).

$$\mathrm{MinimalRelation}(R^*, \mathcal{C}_{\mathrm{it}}) \Longrightarrow \mathrm{IteratorRelation}(R^*, (P, S, 1), W, c, g).$$

*Remark* (Negated quantified statement).

$$R^* \text{ is not an iterator relation for } (W, c, g).$$

*Remark* (Contrapositive quantified statement).

$R^*$ is not an iterator relation for $(W, c, g) \Longrightarrow R^*$ is not the minimal iterator relatio

*Remark* (Interpretation). The intersection construction does not lose the base pair or the successor closure condition. The minimal relation is therefore still one of the admissible iterator relations.

*Remark* (Dependencies).

- Minimal Iterator Relation

**Lemma (Forced Values Are Unique)**

**Lemma 3.4.5** (Forced Values Are Unique). *Let $(P, S, 1)$ be a Peano system, let $(W, c, g)$ be iterator data for it, and let $R^*$ be the minimal iterator relation. If $(n, w_0) \in R^*$ and $(n, w_1) \in R^*$, then $w_0 = w_1$. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall n \in P\ \forall w_0, w_1 \in W\ \big((n, w_0) \in R^* \wedge (n, w_1) \in R^* \Longrightarrow w_0 = w_1\big).$$

*Remark* (Predicate reading).

$$\mathrm{FunctionalRelation}(R^*, P, W).$$

*Remark* (Negated quantified statement).

$$\exists n \in P\ \exists w_0, w_1 \in W\ \big((n, w_0) \in R^* \wedge (n, w_1) \in R^* \wedge w_0 \neq w_1\big).$$

*Remark* (Failure modes). Forced value uniqueness fails exactly when one stage is assigned two distinct values by the minimal iterator relation.

*Remark* (Contrapositive quantified statement).

$\exists n \in P\ \exists w_0, w_1 \in W\ \big((n, w_0) \in R^* \wedge (n, w_1) \in R^* \wedge w_0 \neq w_1\big) \Longrightarrow R^*$ is not functional at every s

*Remark* (Interpretation). No stage can be forced to carry two different values. This is the key determinacy step in the minimal-relation construction used to turn recursive specifications into functions [Feferman, 1964].

*Remark* (Dependencies).

- Consistency of the Minimal Iterator Relation

**Lemma (Determinism of the Minimal Iterator Relation)**

**Lemma 3.4.6** (Determinism of the Minimal Iterator Relation). *Let $(P, S, 1)$ be a Peano system, let $(W, c, g)$ be iterator data for it, and let $R^*$ be the minimal iterator relation. For each $n \in P$, there is at most one $w \in W$ such that $(n, w) \in R^*$. Go to proof.*

**Lean 4 Verification Record**

**Source anchor:** Feferman, *The Number Systems*, Section 3.4; Mendelson, *Number Systems and the Foundations of Analysis*, Section 2.2.
**Formalizes:** Lemma 3.4.6.
**Lean module:** `LRA.VolumeII.PeanoSystems.Recursion`.
**Lean declaration:** `minimal_iterator_relation_deterministic`.
**Status:** checked against `lra-lean`.

```
theorem minimal_iterator_relation_deterministic
    (ps : PeanoSystem)
    (data : IteratorData ps) :
    forall element : ps.carrier,
      forall first_value second_value : data.target,
        minimal_iterator_relation ps data element first_value ->
        minimal_iterator_relation ps data element second_value ->
        first_value = second_value := by
  apply induction_principle ps

  case base_case =>
    intro first_value second_value first_forced second_forced
    exact
      Eq.trans
        (minimal_iterator_relation_base_unique ps data first_value first_forced)
        (minimal_iterator_relation_base_unique ps data second_value second_forced).symm

  case successor_step =>
    intro element induction_hypothesis
    intro first_successor_value second_successor_value
    intro first_forced second_forced
    cases minimal_iterator_relation_complete ps data element with
    | intro current_value current_forced =>
        exact
          Eq.trans
            (forced_successor_values_are_unique
              ps data element current_value first_successor_value
              induction_hypothesis current_forced first_forced)
            (forced_successor_values_are_unique
              ps data element current_value second_successor_value
              induction_hypothesis current_forced second_forced).symm
```

*Remark* (Standard quantified statement).

$$\forall n \in P\ \forall w_0, w_1 \in W\ \big((n, w_0) \in R^* \wedge (n, w_1) \in R^* \Longrightarrow w_0 = w_1\big).$$

*Remark* (Predicate reading).

$$\mathrm{FunctionalRelation}(R^*, P, W).$$

*Remark* (Negated quantified statement).

$$\exists n \in P\ \exists w_0, w_1 \in W\ \big((n, w_0) \in R^* \wedge (n, w_1) \in R^* \wedge w_0 \neq w_1\big).$$

*Remark* (Failure modes). Determinism fails exactly when a single stage has two distinct values in the minimal iterator relation.

*Remark* (Dependencies).

- Forced Values Are Unique

**Lemma (Completeness of the Minimal Iterator Relation)**

**Lemma 3.4.7** (Completeness of the Minimal Iterator Relation). *Let $(P, S, 1)$ be a Peano system, let $(W, c, g)$ be iterator data for it, and let $R^*$ be the minimal iterator relation. For each $n \in P$, there exists $w \in W$ such that $(n, w) \in R^*$. Go to proof.*

**Lean 4 Verification Record**

**Source anchor:** Feferman, *The Number Systems*, Section 3.4; Mendelson, *Number Systems and the Foundations of Analysis*, Section 2.2.
**Formalizes:** Lemma 3.4.7.
**Lean module:** `LRA.VolumeII.PeanoSystems.Recursion`.
**Lean declaration:** `minimal_iterator_relation_complete`.
**Status:** checked against `lra-lean`.

```
theorem minimal_iterator_relation_complete
    (ps : PeanoSystem)
    (data : IteratorData ps) :
    forall element : ps.carrier,
      Exists (fun value : data.target =>
        minimal_iterator_relation ps data element value) := by
  apply induction_principle ps

  case base_case =>
    exact
      Exists.intro
        data.initial_value
        (minimal_iterator_relation_is_iterator_relation ps data).left

  case successor_step =>
    intro element induction_hypothesis
    cases induction_hypothesis with
    | intro value value_is_forced =>
        exact
          Exists.intro
            (data.step_rule value)
            ((minimal_iterator_relation_is_iterator_relation ps data).right
              element value value_is_forced)
```

*Remark* (Standard quantified statement).

$$\forall n \in P\ \exists w \in W\ ((n, w) \in R^*).$$

*Remark* (Predicate reading).

$$\mathrm{TotalRelation}(R^*, P, W).$$

*Remark* (Negated quantified statement).

$$\exists n \in P\ \forall w \in W\ ((n, w) \notin R^*).$$

*Remark* (Failure modes). Completeness fails exactly when some stage of $P$ receives no value in the minimal iterator relation.

*Remark* (Source alignment). This is the totality half of Feferman's minimal-relation construction for recursive definition: after determinacy gives at most one value at each stage, completeness gives at least one value at each stage [Feferman, 1964].

*Remark* (Dependencies).

- Consistency of the Minimal Iterator Relation
- Induction Principle for a Peano System

**Lemma (Uniqueness of Iterator Functions)**

**Lemma 3.4.8** (Uniqueness of Iterator Functions). *Let $(P, S, 1)$ be a Peano system, and let $(W, c, g)$ be iterator data for it. If $f, h : P \to W$ satisfy*

$$f(1) = c, \qquad h(1) = c,$$

*and*

$$\forall n \in P\ (f(S(n)) = g(f(n))) \qquad \textit{and} \qquad \forall n \in P\ (h(S(n)) = g(h(n))),$$

*then $f = h$. Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}\forall f, h : P \to W \Big( & f(1) = c \wedge h(1) = c \\ & \wedge \forall n \in P\ (f(S(n)) = g(f(n))) \\ & \wedge \forall n \in P\ (h(S(n)) = g(h(n))) \Longrightarrow f = h \Big).\end{aligned}$$

*Remark* (Predicate reading).

$$\forall f, h : P \to W \ \big(\text{IteratorFunction}(f, (P, S, 1), W, c, g) \wedge \text{IteratorFunction}(h, (P, S, 1), W, c, g) \Longrightarrow f = h\big)$$

*Remark* (Negated quantified statement).

$$\begin{aligned}\exists f, h : P \to W \Big( & f(1) = c \wedge h(1) = c \\ & \wedge \forall n \in P\ (f(S(n)) = g(f(n))) \\ & \wedge \forall n \in P\ (h(S(n)) = g(h(n))) \wedge f \neq h \Big).\end{aligned}$$

*Remark* (Failure modes). Uniqueness fails exactly when two distinct functions satisfy the same base clause and the same successor clause.

*Remark* (Contrapositive quantified statement).

$$\begin{aligned} f \neq h \Longrightarrow & f(1) \neq c \vee h(1) \neq c \\ & \vee \exists n \in P\ (f(S(n)) \neq g(f(n))) \\ & \vee \exists n \in P\ (h(S(n)) \neq g(h(n))). \end{aligned}$$

*Remark* (Interpretation). The iterator clauses determine at most one function. Any difference between two candidate functions must show up as a failure of the base clause or a failure of one of the successor clauses.

*Remark* (Dependencies).

- Iterator Data
- Induction Principle for a Peano System

**Lemma (Existence of an Iterator Function)**

**Lemma 3.4.9** (Existence of an Iterator Function). *Let $(P, S, 1)$ be a Peano system, and let $(W, c, g)$ be iterator data for it. Then there exists a function $f : P \to W$ such that*

$$f(1) = c \qquad \textit{and} \qquad \forall n \in P\ (f(S(n)) = g(f(n))).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\exists f : P \to W\ \big(f(1) = c \wedge \forall n \in P\ (f(S(n)) = g(f(n)))\big).$$

*Remark* (Predicate reading).

$$\exists f : P \to W\ \ \mathrm{IteratorFunction}(f, (P, S, 1), W, c, g).$$

*Remark* (Negated quantified statement).

$$\forall f : P \to W\ \big(f(1) \neq c \vee \exists n \in P\ (f(S(n)) \neq g(f(n)))\big).$$

*Remark* (Failure modes). Existence fails exactly when every candidate function misses the base value or misses the recursive successor equation at some stage.

*Remark* (Contrapositive quantified statement).

$\neg\exists f : P \to W\ \big(f(1) = c \wedge \forall n \in P\ (f(S(n)) = g(f(n)))\big) \implies R^*$ is not both deterministic and co

*Remark* (Interpretation). Existence is the assertion that the minimal relation is not merely graph-like locally, but supplies a value at every stage and hence determines an actual function.

*Remark* (Dependencies).

- Determinism of the Minimal Iterator Relation
- Completeness of the Minimal Iterator Relation

**Theorem (Peano Iterator Theorem)**

**Theorem 3.4.10** (Peano Iterator Theorem). *Let $(P, S, 1)$ be a Peano system, and let $(W, c, g)$ be iterator data for it. Then there exists a unique function $f : P \to W$ such that*

$$f(1) = c \qquad \textit{and} \qquad \forall n \in P\ (f(S(n)) = g(f(n))).$$

*Go to proof.*

**Lean 4 Verification Record**

**Source anchor:** Feferman, *The Number Systems*, Theorem 3.4; Mendelson, *Number Systems and the Foundations of Analysis*, Section 2.2.
**Formalizes:** Theorem 3.4.10.
**Lean module:** `LRA.VolumeII.PeanoSystems.Recursion`.
**Lean declaration:** `peano_iterator_theorem`.
**Status:** checked against `lra-lean`.

```
theorem peano_iterator_theorem
    (ps : PeanoSystem)
    (target : Type)
    (initial_value : target)
    (step_rule : target -> target) :
    Exists (fun iterator_function : ps.carrier -> target =>
      And
        (satisfies_iterator_clauses
          ps target initial_value step_rule iterator_function)
        (forall other_iterator : ps.carrier -> target,
          satisfies_iterator_clauses
            ps target initial_value step_rule other_iterator ->
          forall element : ps.carrier,
            other_iterator element = iterator_function element)) := by
  refine
    Exists.intro
      (iter ps target initial_value step_rule)
      ?_
  constructor
  exact iter_satisfies_iterator_clauses ps target initial_value step_rule
  intro other_iterator other_satisfies element
  exact
    Eq.symm
      (iterator_function_unique
        ps target initial_value step_rule
        (iter ps target initial_value step_rule)
        other_iterator
        (iter_satisfies_iterator_clauses ps target initial_value step_rule)
        other_satisfies
        element)
```

*Remark* (Standard quantified statement).

$$\exists! f : P \to W \ \big(f(1) = c \wedge \forall n \in P \ (f(S(n)) = g(f(n)))\big).$$

*Remark* (Predicate reading).

$$\exists! f : P \to W \ \ \mathrm{IteratorFunction}(f, (P, S, 1), W, c, g).$$

*Remark* (Negated quantified statement).

$$\neg\exists! f : P \to W \ \big(f(1) = c \wedge \forall n \in P \ (f(S(n)) = g(f(n)))\big).$$

*Remark* (Failure modes). The iterator theorem fails only if no function satisfies the iterator clauses or if more than one function satisfies them.

*Remark* (Failure mode decomposition).

$$\begin{aligned}
&\neg\exists! f : P \to W \left(f(1) = c \wedge \forall n \in P \left(f(S(n)) = g(f(n))\right)\right) \\
\iff &\underbrace{\neg\exists f : P \to W \left(f(1) = c \wedge \forall n \in P \left(f(S(n)) = g(f(n))\right)\right)}_{\text{existence fails}} \\
&\vee \underbrace{\exists f, h : P \to W \left(f \neq h \wedge f(1) = c \wedge h(1) = c \wedge \forall n \in P \left(f(S(n)) = g(f(n))\right) \wedge \forall n \in P \left(h(S(n))\right.\right.}_{\text{uniqueness fails}}
\end{aligned}$$

*Remark* (Interpretation). The theorem says that iterator data is sufficient to turn a recursive specification into a genuine definition: there is exactly one graph and hence exactly one function satisfying the base and successor clauses.

*Remark* (Historical note). This is the one-based Peano-system version of the iterator theorem in Feferman [1964], Mendelson [1982], Simpson [2009]. It is also the set-theoretic instance of the natural numbers object universal property [Lawvere, 1969]; Dedekind's recursion theorem is the classical ancestor [Dedekind, 1901].

*Remark* (Dependencies).

- Uniqueness of Iterator Functions
- Existence of an Iterator Function

**Definition (Iterator-Generated Function)**

**Definition 3.4.11** (Iterator-Generated Function). Let $(P, S, 1)$ be a Peano system, and let $(W, c, g)$ be iterator data for it. The unique function $f : P \to W$ satisfying

$$f(1) = c \qquad \text{and} \qquad \forall n \in P \left(f(S(n)) = g(f(n))\right)$$

is called the *iterator-generated function determined by* $(W, c, g)$.

*Remark* (Standard quantified statement).

$f$ is the iterator-generated function determined by $(W, c, g) \iff \big(f : P \to W \wedge f(1) = c \wedge \forall n$

*Remark* (Definition predicate reading).

$\mathrm{IteratorGeneratedFunction}(f, (P, S, 1), W, c, g) \iff \mathrm{IteratorFunction}(f, (P, S, 1), W, c, g) \wedge \exists! h : P \to W$

*Remark* (Negated quantified statement).

$f$ is not the iterator-generated function determined by $(W, c, g) \iff f$ is not a functio
$\vee\, f(1) \neq c \vee \exists n \in P$

*Remark* (Failure modes). A candidate fails to be the iterator-generated function if it is not a function from $P$ to $W$, if it has the wrong base value, or if it violates the successor clause.

*Remark* (Interpretation). This definition packages the unique function supplied by the Peano Iterator Theorem as a named object available for later constructions.

*Remark* (Dependencies).

- Peano Iterator Theorem

**Theorem (Iterator Base Clause)**

**Theorem 3.4.12** (Iterator Base Clause). *Let $(P, S, 1)$ be a Peano system, let $(W, c, g)$ be iterator data for it, and let $f$ be the iterator-generated function determined by $(W, c, g)$. Then*

$$f(1) = c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$f$ is the iterator-generated function determined by $(W, c, g) \implies f(1) = c.$

*Remark* (Predicate reading).

$$\text{IteratorGeneratedFunction}(f, (P, S, 1), W, c, g) \implies f(1) = c.$$

*Remark* (Contrapositive quantified statement).

$f(1) \neq c \implies f$ is not the iterator-generated function determined by $(W, c, g)$.

*Remark* (Interpretation). The base clause is not an additional theorem about the generated function; it is one of the clauses built into the unique recursive specification.

*Remark* (Dependencies).

- Iterator-Generated Function

**Theorem (Iterator Successor Clause)**

**Theorem 3.4.13** (Iterator Successor Clause). *Let $(P, S, 1)$ be a Peano system, let $(W, c, g)$ be iterator data for it, and let $f$ be the iterator-generated function determined by $(W, c, g)$. Then*

$$\forall n \in P\ (f(S(n)) = g(f(n))).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$f$ is the iterator-generated function determined by $(W, c, g) \implies \forall n \in P\ (f(S(n)) = g(f(n))$

*Remark* (Predicate reading).

$$\text{IteratorGeneratedFunction}(f, (P, S, 1), W, c, g) \implies \forall n \in P\ (f(S(n)) = g(f(n))).$$

*Remark* (Contrapositive quantified statement).

$\exists n \in P\ (f(S(n)) \neq g(f(n))) \implies f$ is not the iterator-generated function determined by $(W$

*Remark* (Interpretation). The successor clause says that the generated function advances by applying the update rule to the previous value at every Peano stage.

*Remark* (Dependencies).

- Iterator-Generated Function

**Definition (Iteration of a Self-Map)**

**Definition 3.4.14** (Iteration of a Self-Map). Let $(P, S, 1)$ be a Peano system, let $W$ be a set, let $c \in W$, and let $g : W \to W$. The iterator-generated function determined by $(W, c, g)$ is called the *iteration of $g$ from $c$ along $P$*.

*Remark* (Standard quantified statement).

$f$ is the iteration of $g$ from $c$ along $P \iff f$ is the iterator-generated function deter

*Remark* (Definition predicate reading).

$f$ is the iteration of $g$ from $c$ along $P \iff \text{IteratorGeneratedFunction}(f, (P, S, 1), W, c, g)$.

*Remark* (Negated quantified statement).

$f$ is not the iteration of $g$ from $c$ along $P \iff f$ is not the iterator-generated functi

*Remark* (Interpretation). Iteration is the special case of recursion in which the same self-map is used at every stage.

*Remark* (Dependencies).

- Iterator-Generated Function

**Corollary (Uniqueness of Binary Iterator Operations)**

**Corollary 3.4.15** (Uniqueness of Binary Iterator Operations). *Let $(P, S, 1)$ be a Peano system, let $A$ be a set, let $W$ be a set, and suppose each $a \in A$ determines iterator data $(W, c_a, g_a)$ for $(P, S, 1)$. If $F, G : A \times P \to W$ satisfy*

$$F(a, 1) = c_a, \qquad G(a, 1) = c_a$$

*and*

$$F(a, S(n)) = g_a(F(a, n)), \qquad G(a, S(n)) = g_a(G(a, n))$$

*for all $a \in A$ and all $n \in P$, then $F = G$. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in A\, \forall n \in P \Big( F(a,1) = c_a \wedge G(a,1) = c_a \\ \wedge F(a,S(n)) = g_a(F(a,n)) \wedge G(a,S(n)) = g_a(G(a,n)) \Big) \Longrightarrow F = G.$$

*Remark* (Predicate reading). For each $a \in A$, write $F_a(n) = F(a,n)$ and $G_a(n) = G(a,n)$. Then

$$\forall a \in A \left(\text{IteratorFunction}(F_a,(P,S,1),W,c_a,g_a) \wedge \text{IteratorFunction}(G_a,(P,S,1),W,c_a,g_a) \Longrightarrow F_a = G_a\right.$$

*Remark* (Negated quantified statement).

$$\exists a \in A\, \exists n \in P \Big( F(a,1) = c_a \wedge G(a,1) = c_a \\ \wedge F(a,S(n)) = g_a(F(a,n)) \wedge G(a,S(n)) = g_a(G(a,n)) \wedge F \neq G \Big).$$

*Remark* (Failure modes). Binary iterator uniqueness fails if two different binary operations satisfy the same one-parameter iterator specification for every fixed parameter.

*Remark* (Interpretation). For each fixed first argument, the binary operation is an ordinary iterator in the second argument. Uniqueness for the binary operation is therefore pointwise uniqueness of iterator functions.

*Remark* (Historical note). This isolates the binary-operation uniqueness pattern used by Thurston before existence is invoked [Thurston, 1956].

*Remark* (Dependencies).

- Uniqueness of Iterator Functions

### General Recursion

Pure iteration uses a fixed self-map $g : W \to W$. General recursion allows the successor rule to depend on the current stage as well as on the current value. The state-encoding theorem below reduces that stage-dependent case to the Peano Iterator Theorem.

**Definition (Stage-Dependent Step Rule)**

**Definition 3.4.16** (Stage-Dependent Step Rule). Let $(P,S,1)$ be a Peano system and let $W$ be a set. A *stage-dependent step rule* on $W$ over $P$ is a function

$$\Phi : P \times W \to W.$$

*Remark* (Standard quantified statement).

$$\Phi \text{ is a stage-dependent step rule on } W \text{ over } P \Longleftrightarrow \Phi : P \times W \to W.$$

*Remark* (Definition predicate reading).

$$\text{StageDependentStepRule}(\Phi, P, W) \iff \Phi : P \times W \to W.$$

*Remark* (Negated quantified statement).

$\Phi$ is not a stage-dependent step rule on $W$ over $P \iff \Phi$ is not a function $P \times W \to W$

*Remark* (Failure modes). A stage-dependent step rule fails precisely when it is not a well-defined function from stage-value pairs back into the value set.

*Remark* (Interpretation). A stage-dependent step rule allows the next value to depend on both the current Peano stage and the current value.

*Remark* (Dependencies).

- Peano System

**Definition (General Recursive Function)**

**Definition 3.4.17** (General Recursive Function). Let $(P, S, 1)$ be a Peano system, let $W$ be a set, let $c \in W$, and let $\Phi : P \times W \to W$ be a stage-dependent step rule. A function $f : P \to W$ is called the *general recursive function determined by* $(W, c, \Phi)$ exactly when

$$f(1) = c \quad \text{and} \quad \forall n \in P\ (f(S(n)) = \Phi(n, f(n))).$$

*Remark* (Standard quantified statement).

$$f \text{ is the general recursive function determined by } (W, c, \Phi) \iff f : P \to W \wedge f(1) = c \wedge \forall n \in P\ (f(S(n)) = \Phi(n,$$

*Remark* (Definition predicate reading).

$$\text{GeneralRecursiveFunction}(f, (P, S, 1), W, c, \Phi) \iff f : P \to W \wedge f(1) = c \wedge \forall n \in P\ (f(S(n)) = \Phi(n, f(n))).$$

*Remark* (Negated quantified statement).

$$f \text{ is not the general recursive function determined by } (W, c, \Phi) \iff f \text{ is not a functior} \vee f(1) \neq c \vee \exists n \in P\ ($$

*Remark* (Failure modes). A candidate fails to be the general recursive function if it has the wrong domain or codomain, the wrong base value, or a failed stage-dependent successor equation.

*Remark* (Interpretation). General recursive functions are the stage-dependent analogues of iterator-generated functions.

*Remark* (Dependencies).

- Stage-Dependent Step Rule

**Lemma (Uniqueness of General Recursive Functions)**

**Lemma 3.4.18** (Uniqueness of General Recursive Functions). *Let* $(P, S, 1)$ *be a Peano system, let* $W$ *be a set, let* $c \in W$*, and let* $\Phi : P \times W \to W$ *be a stage-dependent step rule. If* $f, h : P \to W$ *satisfy*

$$f(1) = c, \qquad h(1) = c,$$

*and*

$$\forall n \in P\ (f(S(n)) = \Phi(n, f(n))) \qquad \textit{and} \qquad \forall n \in P\ (h(S(n)) = \Phi(n, h(n))),$$

*then* $f = h$*. Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}\forall f, h : P \to W \Big( & f(1) = c \wedge h(1) = c \\ & \wedge \forall n \in P\ (f(S(n)) = \Phi(n, f(n))) \\ & \wedge \forall n \in P\ (h(S(n)) = \Phi(n, h(n))) \implies f = h \Big).\end{aligned}$$

*Remark* (Predicate reading).

$$\forall f, h : P \to W \ \big(\text{GeneralRecursiveFunction}(f, (P, S, 1), W, c, \Phi) \wedge \text{GeneralRecursiveFunction}(h, (P, S, 1),$$

*Remark* (Negated quantified statement).

$$\begin{aligned}\exists f, h : P \to W \Big( & f(1) = c \wedge h(1) = c \\ & \wedge \forall n \in P\ (f(S(n)) = \Phi(n, f(n))) \\ & \wedge \forall n \in P\ (h(S(n)) = \Phi(n, h(n))) \wedge f \neq h \Big).\end{aligned}$$

*Remark* (Failure modes). Uniqueness of general recursive functions fails if two distinct functions satisfy the same base clause and the same stage-dependent successor clause.

*Remark* (Contrapositive quantified statement).

$$\begin{aligned} f \neq h \implies & f(1) \neq c \vee h(1) \neq c \\ & \vee \exists n \in P\ (f(S(n)) \neq \Phi(n, f(n))) \\ & \vee \exists n \in P\ (h(S(n)) \neq \Phi(n, h(n))).\end{aligned}$$

*Remark* (Interpretation). Once the initial value and stage-dependent rule are fixed, two recursive candidates cannot diverge without violating one of the defining clauses.

*Remark* (Dependencies).

- General Recursive Function

- Induction Principle for a Peano System

**Theorem (General Recursion by State Encoding)**

**Theorem 3.4.19** (General Recursion by State Encoding). *Let $(P, S, 1)$ be a Peano system, let $W$ be a set, let $c \in W$, and let $\Phi : P \times W \to W$ be a stage-dependent step rule. There exists a function $H : P \to P \times W$ such that*

$$H(1) = (1, c)$$

*and*

$$\forall n \in P \left(H(S(n)) = (S(\pi_1(H(n))), \Phi(\pi_1(H(n)), \pi_2(H(n))))\right).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\exists H : P \to P \times W \left(H(1) = (1, c) \wedge \forall n \in P\ H(S(n)) = (S(\pi_1(H(n))), \Phi(\pi_1(H(n)), \pi_2(H(n))))\right).$$

*Remark* (Predicate reading). Let $\Gamma_\Phi : P \times W \to P \times W$ be defined by

$$\Gamma_\Phi(p, w) = (S(p), \Phi(p, w)).$$

Then state encoding is the iterator assertion

$$\exists H : P \to P \times W\ \mathrm{IteratorFunction}(H, (P, S, 1), P \times W, (1, c), \Gamma_\Phi).$$

*Remark* (Negated quantified statement).

$$\forall H : P \to P \times W \Big(H(1) \neq (1, c) \\ \vee \exists n \in P\ H(S(n)) \neq (S(\pi_1(H(n))), \Phi(\pi_1(H(n)), \pi_2(H(n))))\Big).$$

*Remark* (Failure modes). State encoding fails if every candidate state function has the wrong initial state or violates the encoded successor step at some stage.

*Remark* (Interpretation). The theorem converts stage-dependent recursion into ordinary iteration by carrying the current stage together with the current value as the recursive state.

*Remark* (Dependencies).

- Peano Iterator Theorem
- Stage-Dependent Step Rule

**Theorem (General Recursion Theorem)**

**Theorem 3.4.20** (General Recursion Theorem). *Let $(P, S, 1)$ be a Peano system, let $W$ be a set, let $c \in W$, and let $\Phi : P \times W \to W$ be a stage-dependent step rule. Then there exists a unique function $f : P \to W$ such that*

$$f(1) = c \qquad \textit{and} \qquad \forall n \in P\ (f(S(n)) = \Phi(n, f(n))).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\exists! f : P \to W \ \big(f(1) = c \wedge \forall n \in P \ (f(S(n)) = \Phi(n, f(n)))\big).$$

*Remark* (Predicate reading).

$$\exists! f : P \to W \ \ \mathrm{GeneralRecursiveFunction}(f, (P, S, 1), W, c, \Phi).$$

*Remark* (Negated quantified statement).

$$\neg\exists! f : P \to W \ \big(f(1) = c \wedge \forall n \in P \ (f(S(n)) = \Phi(n, f(n)))\big).$$

*Remark* (Failure modes). General recursion fails if no function satisfies the general recursive clauses or if two distinct functions satisfy them.

*Remark* (Failure mode decomposition).

$$\begin{aligned}
&\neg\exists! f : P \to W \ \big(f(1) = c \wedge \forall n \in P \ (f(S(n)) = \Phi(n, f(n)))\big) \\
\iff &\underbrace{\neg\exists f : P \to W \ \big(f(1) = c \wedge \forall n \in P \ (f(S(n)) = \Phi(n, f(n)))\big)}_{\text{existence fails}} \\
&\vee \underbrace{\exists f, h : P \to W \ \big(f \neq h \wedge f(1) = c \wedge h(1) = c \wedge \forall n \in P \ (f(S(n)) = \Phi(n, f(n))) \wedge \forall n \in P \ (h(S(}_{\text{uniqueness fails}}
\end{aligned}$$

*Remark* (Interpretation). General recursion justifies recursive definitions whose successor rule depends on the current stage as well as the current value.

*Remark* (Historical note). This is the stage-dependent form corresponding to Feferman's Theorem 3.4' [Feferman, 1964]. It is the form used later for arithmetic definitions; those definitions are deferred to their own sections [Feferman, 1964, Mendelson, 1982].

*Remark* (Dependencies).

- Uniqueness of General Recursive Functions
- General Recursion by State Encoding

*Remark* (Deferred arithmetic definitions). Definitions of addition, multiplication, exponentiation, and later arithmetic operations are not part of this section. They begin after the recursion principles have been established.

**Theorem (Uniqueness of Peano Systems up to Isomorphism)**

**Theorem 3.4.21** (Uniqueness of Peano Systems up to Isomorphism). *Let $(P, S_P, 1_P)$ and $(Q, S_Q, 1_Q)$ be Peano systems. Then there exists a unique bijection $\theta : P \to Q$ such that*

$$\theta(1_P) = 1_Q$$

*and*

$$\forall n \in P \ (\theta(S_P(n)) = S_Q(\theta(n))).$$

*Thus any two Peano systems are isomorphic. Go to proof.*

*Remark* (Standard quantified statement).

$$\exists!\theta : P \to Q \left(\theta \text{ is bijective} \wedge \theta(1_P) = 1_Q \wedge \forall n \in P \left(\theta(S_P(n)) = S_Q(\theta(n))\right)\right).$$

*Remark* (Predicate reading). Let $A = (P, S_P, 1_P)$ and $B = (Q, S_Q, 1_Q)$. The theorem asserts

$$\text{Isomorphism}(A, B),$$

with unique witness $\theta : P \to Q$ satisfying the displayed base and successor-preservation clauses.

*Remark* (Negated quantified statement).

$$\begin{aligned}\neg\exists!\theta : P \to Q \big(&\theta \text{ is bijective} \wedge \theta(1_P) = 1_Q \\ &\wedge \forall n \in P \left(\theta(S_P(n)) = S_Q(\theta(n))\right)\big).\end{aligned}$$

*Remark* (Failure modes). Categoricity fails if no structure-preserving bijection exists between the two Peano systems or if more than one such bijection exists.

*Remark* (Interpretation). All Peano systems have the same structure up to a unique isomorphism. The elements may be named differently, but the distinguished first element and successor structure determine the translation.

*Remark* (Historical note). This is the categoricity theorem for Peano systems, following Feferman's Theorem 3.5 and Simpson's Peano-system presentation [Feferman, 1964, Simpson, 2009].

*Remark* (Dependencies).

- Peano System
- General Recursion Theorem

# Chapter 4

# Natural Numbers ($\mathbb{N}$)

**natural-numbers**

Arithmetic Operations and Relations $\to$ **Natural Numbers ($\mathbb{N}$)** $\to$ Whole Numbers ($\mathbb{W}$)

## 4.1 Axioms for $\mathbb{N}$

### The Canonical Peano System

The natural numbers are now fixed as the canonical one-based Peano system. The preceding Peano chapter developed the abstract interface $(X, e, S)$; in this chapter, $\mathbb{N}$ is the distinguished instance obtained by taking the base element to be 1. The symbol 0 is not part of this system. It is adjoined in the next chapter when the whole numbers are constructed.

**Definition (The Natural Numbers)**

**Definition 4.1.1** (The Natural Numbers). The natural numbers $\mathbb{N}$ are the canonical Peano system

$$(\mathbb{N}, 1, S),$$

where $1 \in \mathbb{N}$ is the distinguished base element and $S : \mathbb{N} \to \mathbb{N}$ is the successor operation.

*Remark* (Standing convention). Throughout this chapter, variables $m, n, k$ range over $\mathbb{N}$ unless a larger ambient set is explicitly named. The element 1 is the base natural number, and $S(n)$ is the successor of $n$.

*Remark* (Dependencies).

- Peano System

### Peano Axioms Specialized to $\mathbb{N}$

The following axioms are not new principles beyond the Peano-system interface; they are the abstract Peano axioms specialized to the canonical natural-number system. Naming them here makes later references inside the arithmetic chapter point directly to $\mathbb{N}$ rather

than to an arbitrary Peano system.

**Axiom (N1, Base)**

**Axiom 4.1.2** (N1, Base).

$$1 \in \mathbb{N}.$$

**Axiom (N2, Successor Closure)**

**Axiom 4.1.3** (N2, Successor Closure).

$$\forall n \in \mathbb{N}\ (S(n) \in \mathbb{N}).$$

**Axiom (N3, Base Has No Predecessor)**

**Axiom 4.1.4** (N3, Base Has No Predecessor).

$$\forall n \in \mathbb{N}\ (S(n) \neq 1).$$

**Axiom (N4, Successor Injectivity)**

**Axiom 4.1.5** (N4, Successor Injectivity).

$$\forall m, n \in \mathbb{N}\ (S(m) = S(n) \Longrightarrow m = n).$$

**Axiom (N5, Induction)**

**Axiom 4.1.6** (N5, Induction). For every predicate $\varphi$ on $\mathbb{N}$,

$$\Big(\varphi(1) \wedge \forall n \in \mathbb{N}\ (\varphi(n) \Longrightarrow \varphi(S(n)))\Big) \Longrightarrow \forall n \in \mathbb{N}\ \varphi(n).$$

## 4.2 Canonical Natural Number System

### Working in $\mathbb{N}$

The preceding chapter established the Peano-system facts needed for arithmetic: induction, successor analysis, recursion, and categoricity. From this point forward, $\mathbb{N}$ denotes the canonical natural-number Peano system, with distinguished first element 1 and successor map $S$. The arithmetic operations introduced in this chapter are defined inside that system by the recursion principles already proved for Peano systems.

*Remark* (Standing convention). Unless otherwise stated, variables $m, n, k$ range over $\mathbb{N}$, the symbol 1 denotes the distinguished first natural number, and $S(n)$ denotes the successor of $n$.

## 4.3 Addition on N

### Recursive Definition of Addition

Addition on a Peano system is defined from the successor primitive by recursion. For each fixed left input $m$, the map $n \mapsto m + n$ is generated from the initial value $S(m)$ and the successor operator on the same Peano system. This definition makes addition a derived operation rather than an independent axiom.

**Definition (Addition on a Peano System)**

**Definition 4.3.1** (Addition on a Peano System). Let $(P, S, 1)$ be a Peano system. For each $m \in P$, define the function

$$f_m : P \to P$$

to be the unique function satisfying

$$f_m(1) = S(m) \qquad \text{and} \qquad \forall n \in P\ (f_m(S(n)) = S(f_m(n))).$$

For $m, n \in P$, define

$$m + n := f_m(n).$$

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system. For each $m \in P$, let $f_m : P \to P$ satisfy $f_m(1) = S(m)$ and $\forall n \in P\ (f_m(S(n)) = S(f_m(n)))$.
Then for all $m, n \in P$, $m + n := f_m(n)$.

*Remark* (Negated quantified statement).

Let $(P, S, 1)$ be a Peano system. For each $m \in P$, let $f_m : P \to P$ satisfy $f_m(1) = S(m)$ and $\forall n \in P\ (f_m(S(n)) = S(f_m(n)))$.
Then for all $m, n, x \in P$, $x \neq m + n \iff x \neq f_m(n)$.

*Remark* (Failure modes). The definition fails only when the designated value does not agree with the unique recursively generated function attached to the fixed left input $m$. Once the recursive map $f_m$ is fixed, every incorrect candidate for $m + n$ is simply a value of $P$ different from $f_m(n)$.

*Remark* (Failure mode decomposition).

Let $(P, S, 1)$ be a Peano system. For each $m \in P$, let $f_m : P \to P$ satisfy $f_m(1) = S(m)$ and $\forall n \in P\ (f_m(S(n)) = S(f_m(n)))$.

$$x \neq m + n \iff \underbrace{x \neq f_m(n)}_{\text{candidate value disagrees with the recursive output}}.$$

*Remark* (Interpretation). Addition is introduced by fixing the left input and recursively generating the right-input map. The value $m + 1$ is set

to be the successor of $m$, and each further successor step in the right input advances the output by one more successor. This makes addition repeated succession.

*Remark* (Dependencies).

- Peano System
- Peano Iterator Theorem

**Theorem (Addition Is Well-Defined on a Peano System)**

**Theorem 4.3.2** (Addition Is Well-Defined on a Peano System). *Let $(P, S, 1)$ be a Peano system. For every $m \in P$, there exists a unique function*

$$f_m : P \to P$$

*such that*

$$f_m(1) = S(m) \qquad \textbf{and} \qquad \forall n \in P\ (f_m(S(n)) = S(f_m(n))).$$

*Consequently the rule*

$$(m, n) \longmapsto m + n := f_m(n)$$

*defines a unique binary operation on $P$. Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system.

$$\forall m \in P\ \exists! f_m : P \to P \left( f_m(1) = S(m) \wedge \forall n \in P\ (f_m(S(n)) = S(f_m(n))) \right).$$

*Remark* (Interpretation). The theorem shows that the recursive clauses for addition determine exactly one right-input map for each fixed left input. The addition operation is therefore not assumed; it is constructed from the successor map by recursion.

*Remark* (Source alignment). This is the house-style well-definedness form of addition obtained by applying the Peano recursion theorem in the manner of Feferman's construction of arithmetic operations [Feferman, 1964].

*Remark* (Dependencies).

- Addition on a Peano System
- Peano Iterator Theorem

### Recursive Clauses as Reusable Equalities

The recursive definition of addition is given by the base value and successor step for the right-input map $n \mapsto m + n$. Those clauses are part of the definition, but they become substantially more useful once isolated as named theorems. Later algebraic arguments repeatedly cite these equalities rather than reopening the recursive construction each time.

**Theorem (Addition With 1)**

**Theorem 4.3.3** (Addition With 1). *Let $(P, S, 1)$ be a Peano system. For every $m \in P$,*

$$m + 1 = S(m).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall m \in P\ (m + 1 = S(m)).$$

*Remark* (Interpretation). This theorem records the base clause of the recursive definition in the operation notation itself. Adding the distinguished first element applies one successor step to the left input.

*Remark* (Dependencies).

- Addition on a Peano System

**Theorem (Addition Successor on the Right)**

**Theorem 4.3.4** (Addition Successor on the Right). *Let $(P, S, 1)$ be a Peano system. For every $m, n \in P$,*

$$m + S(n) = S(m + n).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall m, n \in P\ (m + S(n) = S(m + n)).$$

*Remark* (Interpretation). This theorem records the recursive step of addition in operation notation. Moving one successor step forward in the right input moves the sum one successor step forward as well.

*Remark* (Dependencies).

- Addition on a Peano System

### Left Input Initialization

The recursive definition controls addition when the right input is advanced by successor. To use addition symmetrically, the first structural task is to understand what happens when the distinguished first element appears on the left. The next theorem shows that left addition by 1 also produces a single successor step.

**Theorem (One Plus $n$)**

**Theorem 4.3.5** (One Plus $n$). *Let $(P, S, 1)$ be a Peano system. For every $n \in P$,*

$$1 + n = S(n).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall n \in P\ (1 + n = S(n)).$$

*Remark* (Interpretation). The theorem shows that the distinguished first element acts on either side as a single successor step. This is the first sign that the recursively defined operation is not tied permanently to the asymmetry of its definition.

*Remark* (Dependencies).

- Addition With 1
- Addition Successor on the Right
- Induction Principle for a Peano System

### Associativity and Commutativity

Once the recursive clauses are available as reusable theorems, addition can be shown to satisfy the two fundamental algebraic symmetries of a binary operation. Associativity makes repeated addition unambiguous, and commutativity shows that the recursively defined operation does not depend on which input is treated as the evolving one.

**Theorem (Addition Is Associative)**

**Theorem 4.3.6** (Addition Is Associative). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$(a + b) + c = a + (b + c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P\ ((a + b) + c = a + (b + c)).$$

*Remark* (Interpretation). Associativity shows that successive additions can be regrouped without changing the value. This makes longer sums structurally coherent and prepares addition for later interaction with order and multiplication.

*Remark* (Dependencies).

- Addition With 1
- Addition Successor on the Right
- Induction Principle for a Peano System

**Theorem (Addition Is Commutative)**

**Theorem 4.3.7** (Addition Is Commutative). *Let $(P, S, 1)$ be a Peano system. For every $a, b \in P$,*

$$a + b = b + a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b \in P\ (a + b = b + a).$$

*Remark* (Interpretation). Although addition is defined by recursion on the right input, the resulting operation is symmetric in its two arguments. Commutativity shows that the apparent asymmetry of the defining clauses is an artifact of construction rather than a feature of the operation itself.

*Remark* (Dependencies).

- Addition With 1
- Addition Successor on the Right
- One Plus $n$
- Addition Is Associative
- Induction Principle for a Peano System

### Cancellation Laws

The next structural property shows that addition preserves enough information to allow one to remove a common summand from an equation. These cancellation laws are the first substitute for subtraction inside a Peano system and are indispensable when order is later defined through additive comparison.

**Theorem (Left Cancellation for Addition)**

**Theorem 4.3.8** (Left Cancellation for Addition). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$a + b = a + c \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \big((a + b = a + c) \Longrightarrow b = c\big).$$

*Remark* (Contrapositive quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \big(b \neq c \Longrightarrow a + b \neq a + c\big).$$

*Remark* (Interpretation). Left cancellation shows that adding the same element to two inputs cannot erase a difference between them. This is the algebraic core that later makes additive definitions of order behave well.

*Remark* (Dependencies).

- Injectivity of Successor
- Addition With 1
- Addition Successor on the Right
- One Plus $n$
- Addition Is Associative
- Addition Is Commutative
- Induction Principle for a Peano System

**Theorem (Right Cancellation for Addition)**

**Theorem 4.3.9** (Right Cancellation for Addition). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$b + a = c + a \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \big((b + a = c + a) \Longrightarrow b = c\big).$$

*Remark* (Contrapositive quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \big(b \neq c \Longrightarrow b + a \neq c + a\big).$$

*Remark* (Interpretation). Right cancellation is the companion to left cancellation. Once commutativity is known, the placement of the common summand no longer matters, so equality can be tested after adding the same element on either side.

*Remark* (Dependencies).

- Addition Is Commutative
- Left Cancellation for Addition

**Theorem (Cancellation for Addition)**

**Theorem 4.3.10** (Cancellation for Addition). *Let $(P, S, 1)$ be a Peano system. Addition on $P$ satisfies cancellation on both sides:*

$$a + b = a + c \Longrightarrow b = c \qquad \textit{and} \qquad b + a = c + a \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Interpretation). The left and right cancellation laws together say that adding a common natural number never destroys equality information, regardless of which side contains the common summand.

*Remark* (Dependencies).

- Left Cancellation for Addition
- Right Cancellation for Addition

### Additive Non-Return

Addition on $\mathbb{N}$ always moves forward by at least one successor step in its right input. The next theorem records the resulting non-return law: adding a natural number to $m$ never returns the added natural number itself.

**Theorem (No Natural Number Equals Itself Plus a Natural Number)**

**Theorem 4.3.11** (No Natural Number Equals Itself Plus a Natural Number). *Let* $(P, S, 1)$ *be a Peano system. For every* $m, n \in P$*,*

$$n \neq m + n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall m, n \in P\ (n \neq m + n).$$

*Remark* (Interpretation). Because every element of $P$ is at least one successor step, the sum $m + n$ lies strictly beyond $n$. The theorem rules out additive cycles and is a standard tool for order arguments on the natural numbers.

*Remark* (Source alignment). This is the house-style form of the additive non-return theorem in [Feferman, 1964].

*Remark* (Dependencies).

- Addition With 1
- Addition Successor on the Right
- No Object Is Its Own Successor
- Induction Principle for a Peano System

## 4.4 Multiplication on N

### Recursive Definition of Multiplication

Multiplication on a Peano system is defined from addition by recursion. For each fixed left input $m$, the map $n \mapsto m \cdot n$ is generated by starting at $m$ and adding one further copy of $m$ at each successor stage of the right input. This makes multiplication repeated addition rather than an independent primitive operation.

**Definition (Multiplication on a Peano System)**

**Definition 4.4.1** (Multiplication on a Peano System). Let $(P, S, 1)$ be a Peano system. For each $m \in P$, define the function

$$g_m : P \to P$$

to be the unique function satisfying

$$g_m(1) = m \qquad \text{and} \qquad \forall n \in P\ (g_m(S(n)) = g_m(n) + m).$$

For $m, n \in P$, define

$$m \cdot n \coloneqq g_m(n).$$

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system. For each $m \in P$, let $g_m : P \to P$ satisfy
$g_m(1) = m \qquad \text{and} \qquad \forall n \in P\ (g_m(S(n)) = g_m(n) + m).$
Then for all $m, n \in P$, $\quad m \cdot n \coloneqq g_m(n)$.

*Remark* (Negated quantified statement).

Let $(P, S, 1)$ be a Peano system. For each $m \in P$, let $g_m : P \to P$ satisfy
$g_m(1) = m \qquad \text{and} \qquad \forall n \in P\ (g_m(S(n)) = g_m(n) + m).$
Then for all $m, n, x \in P$, $\quad x \neq m \cdot n \iff x \neq g_m(n)$.

*Remark* (Failure modes). The definition fails only when a proposed value for $m \cdot n$ does not agree with the recursively generated function attached to the fixed left factor $m$. Once $g_m$ is determined, every incorrect candidate product is simply a value different from $g_m(n)$.

*Remark* (Failure mode decomposition).

Let $(P, S, 1)$ be a Peano system. For each $m \in P$, let $g_m : P \to P$ satisfy
$g_m(1) = m \qquad \text{and} \qquad \forall n \in P\ (g_m(S(n)) = g_m(n) + m).$

$$x \neq m \cdot n \iff \underbrace{x \neq g_m(n)}_{\text{candidate value disagrees with the recursive output}}.$$

*Remark* (Interpretation). Multiplication is introduced by fixing the left input and recursively generating the right-input map. The value $m \cdot 1$ is initialized at $m$, and each successor step in the right input adds one further copy of $m$. Multiplication is therefore repeated addition.

*Remark* (Dependencies).

- Peano System
- General Recursion Theorem for a Peano System
- Addition on a Peano System

**Theorem (Multiplication Is Well-Defined on a Peano System)**

**Theorem 4.4.2** (Multiplication Is Well-Defined on a Peano System). *Let $(P, S, 1)$ be a Peano system. For every $m \in P$, there exists a unique function*

$$g_m : P \to P$$

*such that*

$$g_m(1) = m \qquad \textbf{and} \qquad \forall n \in P\ (g_m(S(n)) = g_m(n) + m).$$

*Consequently the rule*

$$(m, n) \longmapsto m \cdot n := g_m(n)$$

*defines a unique binary operation on $P$. Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system.

$$\forall m \in P\ \exists! g_m : P \to P \Big(g_m(1) = m \wedge \forall n \in P\ (g_m(S(n)) = g_m(n) + m)\Big).$$

*Remark* (Interpretation). The recursive clauses for multiplication determine exactly one right-input map for each fixed left input. The multiplication operation is therefore constructed from recursion and addition rather than assumed at the outset.

*Remark* (Source alignment). This is the house-style well-definedness form of multiplication obtained by applying the Peano recursion theorem in the manner of Feferman's construction of arithmetic operations [Feferman, 1964].

*Remark* (Dependencies).

- Multiplication on a Peano System
- General Recursion Theorem for a Peano System
- Addition on a Peano System

### Recursive Clauses as Reusable Equalities

The recursive definition of multiplication is given by a base value and a successor step. As with addition, these clauses become more useful once they are isolated as named theorems that can be cited directly in later algebraic arguments.

**Theorem (Multiplication With 1)**

**Theorem 4.4.3** (Multiplication With 1). *Let $(P, S, 1)$ be a Peano system. For every $m \in P$,*

$$m \cdot 1 = m.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall m \in P\ (m \cdot 1 = m).$$

*Remark* (Interpretation). This theorem records the base clause of the recursive definition in operation notation. Multiplying by the distinguished first element leaves the left input unchanged.

*Remark* (Dependencies).

- Multiplication on a Peano System

**Theorem (Multiplication Successor on the Right)**

**Theorem 4.4.4** (Multiplication Successor on the Right). *Let $(P, S, 1)$ be a Peano system. For every $m, n \in P$,*

$$m \cdot S(n) = (m \cdot n) + m.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall m, n \in P\ (m \cdot S(n) = (m \cdot n) + m).$$

*Remark* (Interpretation). Moving one successor step forward in the right input adds one further copy of $m$ to the product. This is the recursive step of multiplication written in operation notation.

*Remark* (Dependencies).

- Multiplication on a Peano System
- Addition on a Peano System

### Left Identity

The recursive definition controls multiplication when the right input evolves. To use multiplication symmetrically, the first structural task is to understand what happens when the distinguished first element appears on the left. The next theorem identifies

that left action explicitly.

**Theorem (One Times $n$)**

**Theorem 4.4.5** (One Times $n$). *Let $(P, S, 1)$ be a Peano system. For every $n \in P$,*

$$1 \cdot n = n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall n \in P\ (1 \cdot n = n).$$

*Remark* (Interpretation). The distinguished first element acts as a multiplicative identity on the left. This is the multiplicative analogue of the theorem that $1 + n$ produces the successor of $n$ in the additive theory.

*Remark* (Dependencies).

- Multiplication With 1
- Multiplication Successor on the Right
- Induction Principle for a Peano System

### Distributivity and Associativity

Multiplication is built from repeated addition, so its first major structural law is distributivity over addition. Once distributivity is available, the operation can be shown to satisfy associativity, which makes iterated products well-defined without ambiguity of grouping.

**Theorem (Multiplication Distributes Over Addition)**

**Theorem 4.4.6** (Multiplication Distributes Over Addition). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P\ (a \cdot (b + c) = (a \cdot b) + (a \cdot c)).$$

*Remark* (Interpretation). Distributivity expresses formally that multiplying by $a$ means repeatedly adding copies of $a$, so a sum in the

second factor splits into two separate blocks of repeated addition. This is the bridge between the additive and multiplicative structures.

*Remark* (Dependencies).

- Multiplication Successor on the Right
- Addition Is Associative
- Addition Is Commutative
- Induction Principle for a Peano System

**Theorem (Left Distributivity of Multiplication Over Addition)**

**Theorem 4.4.7** (Left Distributivity of Multiplication Over Addition). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b, c \in P$,

$$(a + b) \cdot c = (a \cdot c) + (b \cdot c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \big((a + b) \cdot c = (a \cdot c) + (b \cdot c)\big).$$

*Remark* (Interpretation). The existing distributive law expands a sum in the right factor. This theorem expands a sum in the left factor. It follows from right distributivity together with commutativity of multiplication.

*Remark* (Source alignment). This is the house-style form of Feferman's left distributive law for multiplication [Feferman, 1964].

*Remark* (Dependencies).

- Multiplication Distributes Over Addition
- Multiplication Is Commutative
- Addition Is Commutative

**Theorem (Multiplication Distributes over Addition on Both Sides)**

**Theorem 4.4.8** (Multiplication Distributes over Addition on Both Sides). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b, c \in P$,

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c) \qquad \textit{and} \qquad (a + b) \cdot c = (a \cdot c) + (b \cdot c).$$

*Go to proof.*

*Remark* (Dependencies).

- Multiplication Distributes Over Addition
- Left Distributivity of Multiplication Over Addition

**Theorem (Multiplication Is Associative)**

**Theorem 4.4.9** (Multiplication Is Associative). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b, c \in P$,

$$(a \cdot b) \cdot c = a \cdot (b \cdot c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P\ ((a \cdot b) \cdot c = a \cdot (b \cdot c)).$$

*Remark* (Interpretation). Associativity shows that repeated multiplication can be regrouped without changing the value. Once this law is established, longer products can be used without parenthetical ambiguity.

*Remark* (Dependencies).

- Multiplication Distributes Over Addition
- Multiplication With 1
- Multiplication Successor on the Right
- Induction Principle for a Peano System

### Commutativity

The recursive definition of multiplication treats the right input as the evolving variable, but the resulting operation is not meant to privilege one factor over the other. The next theorem shows that the asymmetry of the construction disappears in the final operation.

**Theorem (Multiplication Is Commutative)**

**Theorem 4.4.10** (Multiplication Is Commutative). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b \in P$,

$$a \cdot b = b \cdot a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b \in P\ (a \cdot b = b \cdot a).$$

*Remark* (Interpretation). Although multiplication is defined recursively in one argument, the resulting operation is symmetric in the two factors. Commutativity shows that repeated addition of $a$ to build $a \cdot b$ agrees exactly with repeated addition of $b$ to build $b \cdot a$.

*Remark* (Dependencies).

- One Times $n$
- Multiplication Successor on the Right
- Multiplication Distributes Over Addition
- Addition Is Commutative
- Induction Principle for a Peano System

### Cancellation Laws

Multiplication by a natural number preserves enough information to allow a common nonzero factor to be removed from an equality. Since every element of the one-based natural-number system is nonzero, no extra nonzero hypothesis is needed in $\mathbb{N}$.

**Theorem (Left Cancellation for Multiplication)**

**Theorem 4.4.11** (Left Cancellation for Multiplication). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$a \cdot b = a \cdot c \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.} \quad \forall a, b, c \in P\ \big(a \cdot b = a \cdot c \Longrightarrow b = c\big).$$

*Remark* (Interpretation). Left cancellation says that multiplying two values by the same natural number cannot collapse a genuine difference between them.

*Remark* (Dependencies).

- Multiplication Preserves and Reflects Strict Order
- Trichotomy on a Peano System

**Theorem (Right Cancellation for Multiplication)**

**Theorem 4.4.12** (Right Cancellation for Multiplication). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b, c \in P$,

$$b \cdot a = c \cdot a \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system. } \forall a, b, c \in P \left(b \cdot a = c \cdot a \Longrightarrow b = c\right).$$

*Remark* (Interpretation). Right cancellation is the symmetric form of multiplicative cancellation. It follows from left cancellation together with commutativity of multiplication.

*Remark* (Dependencies).

- Left Cancellation for Multiplication
- Multiplication Is Commutative

**Theorem (Cancellation for Multiplication)**

**Theorem 4.4.13** (Cancellation for Multiplication). *Let* $(P, S, 1)$ *be a Peano system. Multiplication on* $P$ *satisfies cancellation on both sides:*

$$a \cdot b = a \cdot c \Longrightarrow b = c \qquad \textit{and} \qquad b \cdot a = c \cdot a \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Dependencies).

- Left Cancellation for Multiplication
- Right Cancellation for Multiplication

## 4.5 Order on N

### Strict and Non-Strict Order

Order on a Peano system is extracted from addition. Since the chapter is using the distinguished first element 1 rather than an additive zero, the strict order is the cleaner primitive notion: $a < b$ means that $b$ is reached from $a$ by adding a nontrivial amount. The non-strict order is then obtained by allowing equality in addition to strict increase.

**Definition (Strict Less Than on a Peano System)**

**Definition 4.5.1** (Strict Less Than on a Peano System). Let $(P, S, 1)$ be a Peano system, and let $a, b \in P$. One writes

$$a < b$$

exactly when there exists $c \in P$ such that

$$b = a + c.$$

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } a, b \in P.\\ &a < b \iff \exists c \in P\ (b = a + c).\end{aligned}$$

*Remark* (Negated quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } a, b \in P.\\ &a \not< b \iff \forall c \in P\ (b \neq a + c).\end{aligned}$$

*Remark* (Failure modes). Strict inequality fails exactly when no additive witness in $P$ carries $a$ to $b$. In that case, every nontrivial additive step from $a$ misses $b$.

*Remark* (Failure mode decomposition).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } a, b \in P.\\ &a \not< b \iff \underbrace{\forall c \in P\ (b \neq a + c)}_{\text{every proposed additive witness fails}}.\end{aligned}$$

*Remark* (Interpretation). The strict order records whether $b$ lies genuinely ahead of $a$ in the counting structure. The witness $c$ is the amount of additive displacement needed to move from $a$ to $b$.

*Remark* (Dependencies).

- Addition on a Peano System

**Definition (Less Than or Equal on a Peano System)**

**Definition 4.5.2** (Less Than or Equal on a Peano System). Let $(P, S, 1)$ be a Peano system, and let $a, b \in P$. One writes

$$a \leq b$$

exactly when

$$a < b \quad \text{or} \quad a = b.$$

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system and let } a, b \in P.\\ &a \leq b \iff \big(a < b \vee a = b\big).\end{aligned}$$

*Remark* (Negated quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system and let } a, b \in P.$$
$$a \not\leq b \iff \big(a \not< b \wedge a \neq b\big).$$

*Remark* (Failure modes). The non-strict order fails in exactly two simultaneous ways: $a$ is not strictly below $b$, and $a$ is not equal to $b$. Thus $b$ neither lies ahead of $a$ nor coincides with it.

*Remark* (Failure mode decomposition).

$$\text{Let } (P, S, 1) \text{ be a Peano system and let } a, b \in P.$$
$$a \not\leq b \iff \underbrace{a \not< b}_{\text{no strict additive witness}} \wedge \underbrace{a \neq b}_{\text{equality also fails}}.$$

*Remark* (Interpretation). The relation $a \leq b$ allows either equality or a genuine forward step from $a$ to $b$. It is the additive order relation derived from the strict witness definition.

*Remark* (Dependencies).

- Strict Less Than on a Peano System

### Basic Order Laws

Once the order relations have been defined, the first task is to show that they satisfy the expected structural laws: reflexivity, transitivity, and antisymmetry. These are the properties that make the additive comparison relation behave as an honest order rather than a merely suggestive notation.

**Theorem (Order Is Reflexive on a Peano System)**

**Theorem 4.5.3** (Order Is Reflexive on a Peano System). *Let $(P, S, 1)$ be a Peano system. For every $a \in P$,*

$$a \leq a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a \in P\ (a \leq a).$$

*Remark* (Interpretation). Every element is comparable to itself. Reflexivity supplies the equality case inside the non-strict order.

*Remark* (Dependencies).

- Less Than or Equal on a Peano System

**Theorem (Order Is Transitive on a Peano System)**

**Theorem 4.5.4** (Order Is Transitive on a Peano System). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b, c \in P$,

$$\big(a \leq b \wedge b \leq c\big) \Longrightarrow a \leq c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\ &\forall a, b, c \in P\ \big((a \leq b \wedge b \leq c) \Longrightarrow a \leq c\big).\end{aligned}$$

*Remark* (Interpretation). If $b$ lies at or beyond $a$, and $c$ lies at or beyond $b$, then $c$ also lies at or beyond $a$. Transitivity expresses the consistency of additive comparison along a chain.

*Remark* (Dependencies).

- Strict Less Than on a Peano System
- Less Than or Equal on a Peano System
- Addition Is Associative

**Theorem (Order Is Antisymmetric on a Peano System)**

**Theorem 4.5.5** (Order Is Antisymmetric on a Peano System). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b \in P$,

$$\big(a \leq b \wedge b \leq a\big) \Longrightarrow a = b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\ &\forall a, b \in P\ \big((a \leq b \wedge b \leq a) \Longrightarrow a = b\big).\end{aligned}$$

*Remark* (Contrapositive quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\ &\forall a, b \in P\ \big(a \neq b \Longrightarrow \neg(a \leq b \wedge b \leq a)\big).\end{aligned}$$

*Remark* (Interpretation). Two distinct elements cannot each lie at or beyond the other. Antisymmetry separates genuine equality from bidirectional comparability.

*Remark* (Dependencies).

- Strict Less Than on a Peano System

- Less Than or Equal on a Peano System
- Left Cancellation for Addition
- Addition Is Associative
- No Object Is Its Own Successor

### Compatibility with Addition

The order on a Peano system is built from addition, so it must interact coherently with addition. The next theorems show that adding the same element to both sides preserves the comparison relation, regardless of whether the common summand is placed on the right or on the left.

**Theorem (Addition Preserves Order on the Right)**

**Theorem 4.5.6** (Addition Preserves Order on the Right). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$a \leq b \Longrightarrow a + c \leq b + c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \left(a \leq b \Longrightarrow a + c \leq b + c\right).$$

*Remark* (Interpretation). Adding the same quantity to two comparable elements does not disturb their order. The additive witness for $a \leq b$ survives after both sides are shifted by the same summand.

*Remark* (Dependencies).

- Strict Less Than on a Peano System
- Less Than or Equal on a Peano System
- Addition Is Associative

**Theorem (Addition Preserves Order on the Left)**

**Theorem 4.5.7** (Addition Preserves Order on the Left). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$a \leq b \Longrightarrow c + a \leq c + b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, c \in P \left(a \leq b \Longrightarrow c + a \leq c + b\right).$$

*Remark* (Interpretation). The same order preservation holds when the common summand is added on the left. This is the left-handed form of the additive compatibility law.

*Remark* (Dependencies).

- Addition Preserves Order on the Right
- Addition Is Commutative

**Theorem (Addition Preserves Order)**

**Theorem 4.5.8** (Addition Preserves Order). *Let $(P, S, 1)$ be a Peano system. For every $a, b, c \in P$,*

$$a \leq b \Longrightarrow a + c \leq b + c \quad \textit{and} \quad a \leq b \Longrightarrow c + a \leq c + b.$$

*Go to proof.*

*Remark* (Dependencies).

- Addition Preserves Order on the Right
- Addition Preserves Order on the Left

### Compatibility with Multiplication

Multiplication by a natural number is repeated addition by a positive amount. It therefore preserves strict comparisons, and because every multiplier lies at or beyond 1, it also reflects strict comparisons.

**Theorem (Multiplication Preserves and Reflects Strict Order)**

**Theorem 4.5.9** (Multiplication Preserves and Reflects Strict Order). *Let $(P, S, 1)$ be a Peano system. For every $k, m, n \in P$,*

$$m < n \Longleftrightarrow k \cdot m < k \cdot n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall k, m, n \in P \left(m < n \Longleftrightarrow k \cdot m < k \cdot n\right).$$

*Remark* (Interpretation). Multiplication by a fixed natural number is an order embedding of the natural numbers into themselves. It shifts strict inequalities forward without collapsing distinct values.

*Remark* (Source alignment). This is the house-style multiplication-order comparison theorem corresponding to Feferman's order laws for products [Feferman, 1964].

*Remark* (Dependencies).

- Strict Less Than on a Peano System
- Multiplication on a Peano System
- Multiplication Distributes Over Addition
- Multiplication Is Commutative
- Addition Preserves Order on the Right
- Left Cancellation for Addition

### Successor Characterization and Trichotomy

Strict inequality can be recast in terms of stepping one successor beyond the smaller element. This successor reformulation sharpens the order relation and leads to trichotomy, which states that any two elements are linearly comparable.

**Theorem (Strict Order Successor Characterization)**

**Theorem 4.5.10** (Strict Order Successor Characterization). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, b \in P$*,*

$$a < b \iff S(a) \leq b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b \in P \left(a < b \iff S(a) \leq b\right).$$

*Remark* (Interpretation). To say that $b$ is strictly larger than $a$ is to say that $b$ lies at or beyond the first successor step past $a$. This theorem converts strict order into a non-strict comparison involving successor.

*Remark* (Dependencies).

- Strict Less Than on a Peano System
- Less Than or Equal on a Peano System
- Addition With 1
- Addition Successor on the Right
- Addition Is Associative
- Addition Is Commutative

- Every Element Is Either 1 or a Successor

**Theorem (Trichotomy on a Peano System)**

**Theorem 4.5.11** (Trichotomy on a Peano System). *Let $(P, S, 1)$ be a Peano system. For every $a, b \in P$, exactly one of the following holds:*

$$a < b, \qquad a = b, \qquad b < a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\&\forall a, b \in P \Big((a < b \vee a = b \vee b < a) \wedge\\&\qquad \neg(a < b \wedge a = b) \wedge \neg(a < b \wedge b < a) \wedge \neg(a = b \wedge b < a)\Big).\end{aligned}$$

*Remark* (Interpretation). Any two elements of a Peano system are linearly comparable: one lies below the other, or they coincide. Trichotomy is the theorem that upgrades the additive comparison relation from a partial-order structure to a total order.

*Remark* (Dependencies).

- Strict Less Than on a Peano System
- Less Than or Equal on a Peano System
- Order Is Reflexive on a Peano System
- Order Is Transitive on a Peano System
- Order Is Antisymmetric on a Peano System
- Strict Order Successor Characterization
- Induction Principle for a Peano System

**Theorem (Natural-Number Order Is Total)**

**Theorem 4.5.12** (Natural-Number Order Is Total). *Let $(P, S, 1)$ be a Peano system. For every $a, b \in P$,*

$$a \leq b \quad \text{or} \quad b \leq a.$$

*Consequently $\leq$ is a total order on $P$. Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.} \quad \forall a, b \in P\ (a \leq b \vee b \leq a).$$

*Remark* (Interpretation). Trichotomy says exactly one of $a < b$, $a = b$, or $b < a$ holds. Since the non-strict order allows either strict comparison or equality, trichotomy immediately yields total comparability for $\leq$.

*Remark* (Dependencies).

- Less Than or Equal on a Peano System
- Trichotomy on a Peano System
- Order Is Reflexive on a Peano System
- Order Is Transitive on a Peano System
- Order Is Antisymmetric on a Peano System

## 4.6 Powers and Growth

### Exponentiation by Recursion

Exponentiation is defined on a Peano system by recursion on the exponent: $a^1 = a$ and $a^{S(n)} = a^n \cdot a$. This makes exponentiation a derived operation built from multiplication in exactly the way multiplication is built from addition. The recursive clauses are immediately isolated as named theorems so that later algebraic arguments can cite them directly.

**Definition (Exponentiation on a Peano System)**

**Definition 4.6.1** (Exponentiation on a Peano System). Let $(P, S, 1)$ be a Peano system. For each $a \in P$, define the function

$$h_a : P \to P$$

to be the unique function satisfying

$$h_a(1) = a \qquad \text{and} \qquad \forall n \in P\ (h_a(S(n)) = h_a(n) \cdot a).$$

For $a, n \in P$, define

$$a^n \coloneqq h_a(n).$$

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system. For each $a \in P$, let $h_a : P \to P$ satisfy $h_a(1) = a \qquad \text{and} \qquad \forall n \in P\ (h_a(S(n)) = h_a(n) \cdot a)$.
Then for all $a, n \in P$, $\quad a^n \coloneqq h_a(n)$.

*Remark* (Negated quantified statement).

Let $(P, S, 1)$ be a Peano system. For each $a \in P$, let $h_a : P \to P$ satisfy $h_a(1) = a$ and $\forall n \in P\ (h_a(S(n)) = h_a(n) \cdot a)$.
Then for all $a, n, x \in P$, $x \neq a^n \iff x \neq h_a(n)$.

*Remark* (Interpretation). Exponentiation is repeated multiplication in the same sense that multiplication is repeated addition. The base $a$ is the fixed left factor; the exponent $n$ counts how many times $a$ is multiplied together. The recursive definition propagates rightward: $a^1 = a$, $a^2 = a^1 \cdot a = a \cdot a$, and so on.

*Remark* (Dependencies).

- Peano System
- General Recursion Theorem for a Peano System
- Multiplication on a Peano System

**Theorem (Exponentiation Is Well-Defined on a Peano System)**

**Theorem 4.6.2** (Exponentiation Is Well-Defined on a Peano System). *Let $(P, S, 1)$ be a Peano system. For every $a \in P$, there exists a unique function*

$$h_a : P \to P$$

*such that*

$$h_a(1) = a \quad \textit{and} \quad \forall n \in P\ (h_a(S(n)) = h_a(n) \cdot a).$$

*Consequently the rule*

$$(a, n) \longmapsto a^n \coloneqq h_a(n)$$

*defines a unique binary operation on $P$. Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system.

$$\forall a \in P\ \exists! h_a : P \to P \Big(h_a(1) = a \wedge \forall n \in P\ (h_a(S(n)) = h_a(n) \cdot a)\Big).$$

*Remark* (Interpretation). The recursive clauses for exponentiation determine exactly one exponent map for each fixed base. Exponentiation is therefore constructed from recursion and multiplication rather than assumed as a primitive operation.

*Remark* (Source alignment). This is the house-style existence-and-uniqueness form of the recursive definition of exponentiation in [Feferman, 1964].

*Remark* (Dependencies).

- General Recursion Theorem for a Peano System
- Multiplication on a Peano System

### Exponent Laws

The recursive definition immediately yields the standard exponent laws. These are not assumed; they are proved from the recursive clauses by induction. The three primary laws are: the sum law $a^{m+n} = a^m \cdot a^n$, the product law $(a \cdot b)^n = a^n \cdot b^n$, and the power law $(a^m)^n = a^{mn}$.

**Theorem (Exponent Sum Law)**

**Theorem 4.6.3** (Exponent Sum Law). *Let $(P, S, 1)$ be a Peano system. For all $a, m, n \in P$,*

$$a^{m+n} = a^m \cdot a^n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system. $\forall a, m, n \in P\ (a^{m+n} = a^m \cdot a^n)$.

*Remark* (Interpretation). Adding exponents corresponds to concatenating the two blocks of repeated multiplication. The proof is induction on $n$ using the recursive clause $a^{S(n)} = a^n \cdot a$ and associativity of multiplication.

*Remark* (Dependencies).

- Exponentiation on a Peano System
- Multiplication Is Associative
- Induction Principle for a Peano System

**Theorem (Exponent Product Law)**

**Theorem 4.6.4** (Exponent Product Law). *Let $(P, S, 1)$ be a Peano system. For all $a, b, n \in P$,*

$$(a \cdot b)^n = a^n \cdot b^n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system. $\forall a, b, n \in P\ ((a \cdot b)^n = a^n \cdot b^n)$.

*Remark* (Interpretation). Raising a product to a power distributes over the factors. The proof is induction on $n$, using commutativity and associativity of multiplication to rearrange the interleaved factors at each successor step.

*Remark* (Dependencies).

- Exponentiation on a Peano System
- Multiplication Is Associative

- Multiplication Is Commutative
- Induction Principle for a Peano System

**Theorem (Exponent Power Law)**

**Theorem 4.6.5** (Exponent Power Law). *Let $(P, S, 1)$ be a Peano system. For all $a, m, n \in P$,*

$$(a^m)^n = a^{m \cdot n}.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system. } \forall a, m, n \in P\ ((a^m)^n = a^{m \cdot n}).$$

*Remark* (Interpretation). Raising a power to a further power multiplies the exponents. The proof is induction on $n$ using the exponent sum law and the recursive definition of multiplication.

*Remark* (Dependencies).

- Exponentiation on a Peano System
- Exponent Sum Law
- Multiplication Successor on the Right
- Induction Principle for a Peano System

### Order Compatibility of Powers

Exponentiation is order-sensitive in both its base and its exponent. For a fixed positive exponent, taking powers preserves and reflects strict order of bases. For a fixed base beyond 1, increasing the exponent strictly increases the value.

**Theorem (Powers with Common Exponent Preserve Strict Order)**

**Theorem 4.6.6** (Powers with Common Exponent Preserve Strict Order). *Let $(P, S, 1)$ be a Peano system. For every $a, b, n \in P$,*

$$a < b \iff a^n < b^n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, b, n \in P\ \big(a < b \iff a^n < b^n\big).$$

*Remark* (Interpretation). A positive exponent does not collapse the strict order between bases. Since the exponent is a natural number,

exponentiation repeatedly multiplies by positive factors and preserves the order information.

*Remark* (Source alignment). This is the house-style common-exponent comparison theorem corresponding to Feferman's natural-number exponent laws [Feferman, 1964].

*Remark* (Dependencies).

- Exponentiation on a Peano System
- Multiplication Preserves and Reflects Strict Order
- Induction Principle for a Peano System

**Theorem (Powers with Common Base Preserve Strict Order)**

**Theorem 4.6.7** (Powers with Common Base Preserve Strict Order). *Let* $(P, S, 1)$ *be a Peano system. For every* $a, m, n \in P$,

$$a^m < a^n \iff 1 < a \wedge m < n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall a, m, n \in P \left(a^m < a^n \iff (1 < a \wedge m < n)\right).$$

*Remark* (Interpretation). Strict growth in the exponent occurs exactly when the base is beyond the multiplicative identity. If the base is 1, all powers are 1; if the base is larger than 1, each successor step in the exponent multiplies by a factor that strictly increases the value.

*Remark* (Source alignment). This is the house-style common-base comparison theorem corresponding to Feferman's exponent-order theorem [Feferman, 1964].

*Remark* (Dependencies).

- Exponentiation on a Peano System
- Multiplication Preserves and Reflects Strict Order
- Exponent Sum Law
- Induction Principle for a Peano System

### Exponential Growth: $n < 2^n$

The most important growth comparison on $\mathbb{N}$ is that exponential growth outpaces linear growth: $n < 2^n$ for all $n \in P$. This single inequality captures why exponential towers

dwarf polynomial bounds and is the prototypical example of an induction argument whose inductive step uses both the hypothesis and a multiplicative amplification.

**Theorem ($n < 2^n$ for All $n \in \mathbb{N}$)**

**Theorem 4.6.8** ($n < 2^n$ for All $n \in \mathbb{N}$). *Let $(P, S, 1)$ be a Peano system, and let $2 := S(1)$. For every $n \in P$,*

$$n < 2^n.$$

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system and let $2 := S(1)$.
$\forall n \in P\ (n < 2^n)$.

*Remark* (Interpretation). The base case $1 < 2 = 2^1$ holds by definition of 2 and the strict order. For the inductive step, assume $n < 2^n$. Then $2^{S(n)} = 2^n \cdot 2$. Since $2^n > n \geq 1$, we have $2^n \cdot 2 > n \cdot 2 = n + n > n + 1 = S(n)$, where the last step uses that $n \geq 1$ so $n + n > n + 1$. Hence $S(n) < 2^{S(n)}$. The inequality $n < 2^n$ is the threshold that distinguishes the growth regimes and reappears in every comparison between polynomial and exponential bounds.

*Remark* (Dependencies).

- Exponentiation on a Peano System
- Strict Less Than on a Peano System
- Multiplication Distributes Over Addition
- Addition Preserves Order on the Right
- Induction Principle for a Peano System

## 4.7 Well Ordering Principle

### The Well-Ordering Principle

The well-ordering principle asserts that every nonempty subset of $\mathbb{N}$ contains a least element. This is not merely a consequence of the order axioms; it is a property that distinguishes $\mathbb{N}$ from dense ordered sets such as $\mathbb{Q}$, which have no smallest positive element. The well-ordering principle is equivalent to ordinary induction over $\mathbb{N}$, but it supports a distinct proof style: to establish a universal statement, assume it fails for some element and derive a contradiction by examining the least failure.

**Theorem (Well-Ordering Principle)**

**Theorem 4.7.1** (Well-Ordering Principle). *Let $(P, S, 1)$ be a Peano system. Every nonempty subset of $P$ has a least element with respect to the order $\leq$ on $P$. Go to proof.*

*Remark* (Standard quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\forall A \subseteq P \left( A \neq \varnothing \implies \exists m \in A \, \forall a \in A \, (m \leq a) \right).$$

*Remark* (Predicate reading).

$$\text{WellOrderingPrinciple} := \forall A \left( (A \subseteq P \wedge \exists a \, (a \in A)) \Rightarrow \exists m \, (m \in A \wedge \forall a \, (a \in A \Rightarrow m \leq a)) \right).$$

*Remark* (Negated quantified statement).

$$\text{Let } (P, S, 1) \text{ be a Peano system.}$$
$$\exists A \subseteq P \left( A \neq \varnothing \wedge \forall m \in A \, \exists a \in A \, (a < m) \right).$$

*Remark* (Failure modes). The well-ordering principle would fail if some nonempty subset of $P$ had no least element. That means for every candidate minimum, a strictly smaller element of the subset exists, producing an infinite strictly descending chain in $P$. Induction rules this out.

*Remark* (Failure mode decomposition).

$$\neg \, \text{WellOrderingPrinciple} \iff \exists A \subseteq P \left( \underbrace{A \neq \varnothing}_{\text{nonempty}} \wedge \underbrace{\forall m \in A \, \exists a \in A \, (a < m)}_{\text{no least element}} \right).$$

*Remark* (Interpretation). The well-ordering principle is the order-theoretic complement to induction. Induction says one can build up: start at 1 and keep stepping forward. Well-ordering says one can always descend to a minimum: any nonempty set has a floor. Proof strategies that use well-ordering typically assume a property fails somewhere, form the set of failures, extract its minimum by well-ordering, and derive a contradiction from the minimality of that element.

*Remark* (Dependencies).

- Peano System
- Less Than or Equal on a Peano System
- Induction Principle for a Peano System
- Trichotomy on a Peano System

### Equivalence of Induction, Strong Induction, and Well-Ordering

Ordinary induction, strong induction, and the well-ordering principle are logically equivalent over a Peano system. Each can be used as the primitive proof engine and the others recovered as theorems. This equivalence matters structurally: it means that induction proofs, strong-induction proofs, and least-counterexample proofs are different presentations of the same foundational mechanism.

**Theorem (Induction, Strong Induction, and Well-Ordering Are Equivalent)**

**Theorem 4.7.2** (Induction, Strong Induction, and Well-Ordering Are Equivalent). *Let $(P, S, 1)$ be a Peano system equipped with the order $\leq$. The following are equivalent:*

1. *The induction principle: for every predicate $\varphi$ on $P$, if $\varphi(1)$ holds and the successor step holds, then $\varphi(n)$ holds for all $n \in P$.*
2. *The strong induction principle: for every predicate $\varphi$ on $P$, if $\varphi(k)$ holds for every $k < n$ implies $\varphi(n)$, then $\varphi(n)$ holds for every $n \in P$.*
3. *The well-ordering principle: every nonempty subset of $P$ has a least element with respect to $\leq$.*

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system.

$$\text{WellOrderingPrinciple} \iff \text{StrongInduction} \iff \Big(\forall \varphi\ (\varphi(1) \land \forall n \in P\ (\varphi(n) \implies \varphi(S(n))) \implies \forall n \in P$$

*Remark* (Interpretation). Induction proves well-ordering by contrapositive: if a nonempty set $A$ had no least element, the set of non-members of $A$ would be inductive, forcing $A = \varnothing$, a contradiction. Well-ordering proves induction by forming the set of elements where $\varphi$ fails and, if nonempty, extracting a least failure to derive a contradiction with the successor step.

*Remark* (Dependencies).

- Induction Principle for a Peano System
- Strong Induction on a Peano System
- Well-Ordering Principle

## 4.8 Induction

### Induction from an Arbitrary Base

Ordinary and strong induction each begin at the distinguished first element 1. Many arguments require starting at a later element. Induction from an arbitrary base $m$ restricts the claim to elements at or above $m$ and starts the inductive process there.

**Theorem (Induction from an Arbitrary Base)**

**Theorem 4.8.1** (Induction from an Arbitrary Base). *Let $(P, S, 1)$ be a Peano system, let $m \in P$, and let $\varphi$ be a predicate on $P$. If*

$$\varphi(m) \quad \textit{and} \quad \forall n \in P\,\big(n \geq m \wedge \varphi(n) \Longrightarrow \varphi(S(n))\big),$$

*then*

$$\forall n \in P\,\big(n \geq m \Longrightarrow \varphi(n)\big).$$

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system, $m \in P$, and $\varphi$ a predicate on $P$.

$$\Big(\varphi(m) \wedge \forall n \in P\,\big(n \geq m \wedge \varphi(n) \Longrightarrow \varphi(S(n))\big)\Big) \Longrightarrow \forall n \in P\,\big(n \geq m \Longrightarrow \varphi(n)\big).$$

*Remark* (Interpretation). Apply ordinary induction to the shifted predicate $\psi(n) := (n \geq m \Longrightarrow \varphi(n))$, which holds vacuously below $m$ and reduces to $\varphi$ above $m$. This theorem is the formal license for all arguments of the form "for all $n \geq m$, proceed by induction."

*Remark* (Dependencies).

- Induction Principle for a Peano System
- Less Than or Equal on a Peano System

## 4.9 Algebraic Structure of N

### $\mathbb{N}$ as a Commutative Monoid Under Addition

The arithmetic established in the preceding sections can now be packaged into a named algebraic structure. Under addition, $\mathbb{N}$ is a commutative monoid with identity 0 — or, in the Peano-system setting that uses 1 as the distinguished element with no additive identity available internally, the correct statement is that $(\mathbb{N}, +)$ satisfies associativity and commutativity. This topic states the structural conclusion that follows from the individual arithmetic theorems proved above.

**Theorem ($\mathbb{N}$ Is Closed, Associative, and Commutative Under Addition)**

**Theorem 4.9.1** ($\mathbb{N}$ Is Closed, Associative, and Commutative Under Addition). *Let $(P, S, 1)$ be a Peano system. The binary operation $+$ on $P$ satisfies:*

1. ***Closure.*** *For all $a, b \in P$, $a + b \in P$.*
2. ***Associativity.*** *For all $a, b, c \in P$, $(a + b) + c = a + (b + c)$.*
3. ***Commutativity.*** *For all $a, b \in P$, $a + b = b + a$.*
4. ***Left cancellation.*** *For all $a, b, c \in P$, if $a + b = a + c$ then $b = c$.*
5. ***Right cancellation.*** *For all $a, b, c \in P$, if $b + a = c + a$ then $b = c$.*

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system.
$\forall a, b, c \in P : (a + b) + c = a + (b + c) \ \wedge \ a + b = b + a \ \wedge \ (a + b = a + c \Longrightarrow b = c)$.

*Remark* (Interpretation). This theorem is a packaging result. It names the algebraic structure that has been proved piece by piece: the three individual theorems on associativity, commutativity, and cancellation together certify that $(P, +)$ is a cancellative commutative semigroup. There is no additive identity in $P$ because $P$ has no element $0$ with $0 + n = n$ for all $n$; the smallest element under addition is $1$, not an identity.

*Remark* (Dependencies).

- Addition Is Associative
- Addition Is Commutative
- Left Cancellation for Addition
- Right Cancellation for Addition

### $\mathbb{N}$ as a Commutative Monoid Under Multiplication

Multiplication on $P$ has a two-sided multiplicative identity, namely $1$, the distinguished first element. Under multiplication, $P$ is therefore a commutative monoid. This topic states the structural packaging for multiplication that parallels the additive packaging above.

**Theorem ($\mathbb{N}$ Is a Commutative Monoid Under Multiplication)**

**Theorem 4.9.2** ($\mathbb{N}$ Is a Commutative Monoid Under Multiplication). *Let $(P, S, 1)$ be a Peano system. The binary operation $\cdot$ on $P$ satisfies:*

1. ***Closure.*** *For all $a, b \in P$, $a \cdot b \in P$.*
2. ***Associativity.*** *For all $a, b, c \in P$, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.*
3. ***Commutativity.*** *For all $a, b \in P$, $a \cdot b = b \cdot a$.*
4. ***Identity.*** *For all $a \in P$, $1 \cdot a = a = a \cdot 1$.*

*Consequently $(P, \cdot, 1)$ is a commutative monoid. Go to proof.*

*Remark* (Standard quantified statement).

$$\begin{aligned}&\text{Let } (P, S, 1) \text{ be a Peano system.}\\&\forall a, b, c \in P : \ (a \cdot b) \cdot c = a \cdot (b \cdot c) \ \wedge \ a \cdot b = b \cdot a \ \wedge \ 1 \cdot a = a.\end{aligned}$$

*Remark* (Interpretation). The distinguished first element $1$ is a multiplicative identity on both sides, so multiplication promotes $P$ from a semigroup to a monoid. The absence of multiplicative inverses (other than $1^{-1} = 1$, which does not lie in $P$ in the usual sense) is what prevents $(P, \cdot)$ from being a group. The structural limitation is intrinsic: no element of $P$ other than $1$ has a reciprocal in $P$.

*Remark* (Dependencies).

- Multiplication Is Associative
- Multiplication Is Commutative
- Multiplication With 1
- One Times $n$

### $\mathbb{N}$ as a Semiring and the Absence of Subtraction

Together, addition and multiplication on $P$ form a semiring: addition is a commutative semigroup, multiplication is a commutative monoid, and multiplication distributes over addition on both sides. The critical structural limitation is that subtraction is not a total operation on $P$. Since $P$ has no additive identity and no additive inverses, the equation $a + x = b$ has a solution in $P$ only when $b > a$; it has no solution when $b \leq a$. This failure of internal subtraction is the reason $\mathbb{Z}$ is introduced immediately after $\mathbb{N}$.

**Theorem ($\mathbb{N}$ Is a Semiring)**

**Theorem 4.9.3** ($\mathbb{N}$ Is a Semiring). *Let $(P, S, 1)$ be a Peano system. Then $(P, +, \cdot)$ satisfies:*

1. *$(P, +)$ is a commutative semigroup.*
2. *$(P, \cdot, 1)$ is a commutative monoid.*
3. *Multiplication distributes over addition: for all $a, b, c \in P$,*

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

Let $(P, S, 1)$ be a Peano system.
$\forall a, b, c \in P: \; a \cdot (b + c) = (a \cdot b) + (a \cdot c) \;\wedge\; (a + b) + c = a + (b + c) \;\wedge\; a + b = b + a \;\wedge\; 1 \cdot a = a.$

*Remark* (Interpretation). A semiring packages the interaction of two operations without requiring additive inverses. The natural numbers are the canonical example of a semiring that is not a ring: they satisfy every ring axiom except the existence of additive inverses, which fails because there is no element $-1$ in $P$. The integers are the smallest ring extending $P$ that repairs this failure.

*Remark* (Dependencies).

- $\mathbb{N}$ Is Closed, Associative, and Commutative Under Addition
- $\mathbb{N}$ Is a Commutative Monoid Under Multiplication
- Multiplication Distributes Over Addition

# Chapter 5

# Constructing the Whole Numbers

**whole-numbers**

Natural Numbers → **Constructing the Whole Numbers** → integers

## 5.1 Why Whole Numbers Are Introduced

### Adjoining an Additive Identity

The natural-number system developed in the preceding chapter is one-based: its distinguished first element is 1. This is sufficient for counting and recursive arithmetic, but addition on $\mathbb{N}$ has no additive identity inside $\mathbb{N}$. The whole numbers are introduced by adjoining a new element before constructing the integers.

### The Missing Identity

*Remark* (Exposition). In the one-based system, every element of $\mathbb{N}$ is positive. The operation $+$ is defined and behaves associatively and commutatively, but there is no element $e \in \mathbb{N}$ satisfying $e+n = n$ for every $n \in \mathbb{N}$. The element needed for that identity is not another successor-stage element; it is a new element placed before 1.

*Remark* (Interpretation). The passage from $\mathbb{N}$ to $\mathbb{W}$ is therefore not a change in the positive arithmetic already constructed. It is an extension that adds the missing additive identity and leaves the positive part intact.

*Remark* (Dependencies).

- Addition on a Peano System
- Addition Is Associative
- Addition Is Commutative

*Remark* (Exposition). Once 0 is available, addition can be packaged as a genuine commutative monoid. That structure is the natural input for the integer construction where ordered pairs are interpreted as formal differences and subtraction is made available by a quotient construction.

## 5.2 Definition of the Whole Numbers

### The Positive Part with a New Zero

The whole numbers are obtained by adjoining a single new element to the one-based natural numbers. The notation records two facts at once: the positive natural numbers remain present by inclusion, and the new element 0 is not one of those positive natural numbers.

### The Underlying Set

**Definition (Whole Numbers)**

**Definition 5.2.1** (Whole Numbers). Let $\mathbb{N}$ be the one-based natural-number system developed in the preceding chapter. Let $0$ be a new element with $0 \notin \mathbb{N}$. Define

$$\mathbb{W} := \mathbb{N} \cup \{0\}.$$

*Remark* (Standard quantified statement).

$$\mathbb{W} = \mathbb{N} \cup \{0\} \quad \text{and} \quad 0 \notin \mathbb{N}.$$

*Remark* (Interpretation). Elements of $\mathbb{W}$ are either the new element $0$ or positive natural numbers. The natural numbers embed into $\mathbb{W}$ by the inclusion map $n \mapsto n$. The element $0$ is not a successor-stage element of the original one-based Peano system.

*Remark* (Dependencies).

- Peano System

**Theorem** ($0 \notin \mathbb{N}$)

**Theorem 5.2.2** ($0 \notin \mathbb{N}$). *The adjoined element $0$ is not an element of the one-based natural-number system:*

$$0 \notin \mathbb{N}.$$

*Go to proof.*

*Remark* (Interpretation). This theorem records the disjointness condition built into the construction of $\mathbb{W}$. The element $0$ is new; it is not a renamed natural number.

*Remark* (Dependencies).

- Whole Numbers

## 5.3 Extending Successor

### Successor After Adjoining Zero

The new successor map on $\mathbb{W}$ agrees with the old successor on the positive part and sends the new initial element 0 to 1. This makes $\mathbb{W}$ a zero-based successor chain while preserving the one-based successor behavior already proved on $\mathbb{N}$.

### The Extended Successor

**Definition (Successor on $\mathbb{W}$)**

**Definition 5.3.1** (Successor on $\mathbb{W}$). Define $S_{\mathbb{W}} : \mathbb{W} \to \mathbb{W}$ by

$$S_{\mathbb{W}}(0) = 1 \qquad \text{and} \qquad S_{\mathbb{W}}(n) = S_{\mathbb{N}}(n) \quad (n \in \mathbb{N}).$$

*Remark* (Standard quantified statement).

$$\begin{aligned} &S_{\mathbb{W}} : \mathbb{W} \to \mathbb{W}, \\ &S_{\mathbb{W}}(0) = 1, \\ &\forall n \in \mathbb{N} \left(S_{\mathbb{W}}(n) = S_{\mathbb{N}}(n)\right). \end{aligned}$$

*Remark* (Interpretation). The definition inserts 0 immediately before the positive chain. After the first step, successor is exactly the successor already defined on $\mathbb{N}$.

*Remark* (Dependencies).

- Whole Numbers
- Peano System

**Theorem (Successor on $\mathbb{W}$ Is Well-Defined)**

**Theorem 5.3.2** (Successor on $\mathbb{W}$ Is Well-Defined). *The rule defining $S_{\mathbb{W}}$ determines a function $S_{\mathbb{W}} : \mathbb{W} \to \mathbb{W}$. Go to proof.*

*Remark* (Standard quantified statement).

$$S_{\mathbb{W}} \text{ is a well-defined function from } \mathbb{W} \text{ to } \mathbb{W}.$$

*Remark* (Interpretation). The cases are exclusive because $0 \notin \mathbb{N}$, and the positive case lands in $\mathbb{N} \subseteq \mathbb{W}$.

*Remark* (Dependencies).

- Whole Numbers
- Successor on $\mathbb{W}$

**Theorem (0 Is Not a Successor in $\mathbb{W}$)**

**Theorem 5.3.3** (0 Is Not a Successor in $\mathbb{W}$). *There is no $a \in \mathbb{W}$ such that $S_{\mathbb{W}}(a) = 0$. Go to proof.*

*Remark* (Standard quantified statement).

$$\neg\exists a \in \mathbb{W}\ \big(S_{\mathbb{W}}(a) = 0\big).$$

*Remark* (Interpretation). The new element $0$ is the first element of the extended chain. It is placed before $1$, not obtained by applying successor to some earlier element.

*Remark* (Dependencies).

- Successor on $\mathbb{W}$
- 1 Is Not a Successor

**Theorem (Every Nonzero Element of $\mathbb{W}$ Is a Successor)**

**Theorem 5.3.4** (Every Nonzero Element of $\mathbb{W}$ Is a Successor). *If $a \in \mathbb{W}$ and $a \neq 0$, then there exists $b \in \mathbb{W}$ such that $a = S_{\mathbb{W}}(b)$. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{W}\ \big(a \neq 0 \Longrightarrow \exists b \in \mathbb{W}\ (a = S_{\mathbb{W}}(b))\big).$$

*Remark* (Interpretation). The element $1$ is the successor of $0$, and every positive natural number beyond $1$ remains a successor in the original Peano chain.

*Remark* (Dependencies).

- Whole Numbers
- Successor on $\mathbb{W}$
- Induction Axiom

**Theorem (Successor on $\mathbb{W}$ Is Injective)**

**Theorem 5.3.5** (Successor on $\mathbb{W}$ Is Injective). *For all $a, b \in \mathbb{W}$,*

$$S_{\mathbb{W}}(a) = S_{\mathbb{W}}(b) \Longrightarrow a = b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W}\ \big(S_{\mathbb{W}}(a) = S_{\mathbb{W}}(b) \Longrightarrow a = b\big).$$

*Remark* (Interpretation). The extended successor still never merges two inputs. The only new case is the comparison between $0$ and a positive natural number, and that case is ruled out by the original non-return condition for the first element.

*Remark* (Dependencies).

- Successor on $\mathbb{W}$

- Injectivity of Successor
- 1 Is Not a Successor

**Theorem ($(\mathbb{W}, 0, S_{\mathbb{W}})$ Satisfies the Successor Properties)**

**Theorem 5.3.6** ($(\mathbb{W}, 0, S_{\mathbb{W}})$ Satisfies the Successor Properties). *The triple $(\mathbb{W}, 0, S_{\mathbb{W}})$ satisfies the successor-side properties:*

$$0 \in \mathbb{W}, \qquad S_{\mathbb{W}} : \mathbb{W} \to \mathbb{W},$$

$$\neg\exists a \in \mathbb{W}\,(S_{\mathbb{W}}(a) = 0), \qquad \forall a, b \in \mathbb{W}\,(S_{\mathbb{W}}(a) = S_{\mathbb{W}}(b) \Longrightarrow a = b).$$

*Go to proof.*

*Remark* (Interpretation). Adjoining 0 shifts the base of the successor chain from 1 to 0. The resulting successor structure has a base element, closes under successor, has no predecessor for the base, and keeps successor injective.

*Remark* (Dependencies).

- Whole Numbers
- Successor on $\mathbb{W}$
- Successor on $\mathbb{W}$ Is Well-Defined
- 0 Is Not a Successor in $\mathbb{W}$
- Successor on $\mathbb{W}$ Is Injective

## 5.4 Extending Addition to $\mathbb{W}$

### Addition with Zero

Addition on $\mathbb{W}$ keeps the old natural-number addition whenever both inputs are positive and adds identity clauses for the new element 0. The case split is exclusive because $0 \notin \mathbb{N}$.

### The Extended Addition Operation

**Definition (Addition on $\mathbb{W}$)**

**Definition 5.4.1** (Addition on $\mathbb{W}$). Define $+_{\mathbb{W}} : \mathbb{W} \times \mathbb{W} \to \mathbb{W}$ by

$$0 +_{\mathbb{W}} a = a, \qquad a +_{\mathbb{W}} 0 = a,$$

and, for $a, b \in \mathbb{N}$,

$$a +_{\mathbb{W}} b = a +_{\mathbb{N}} b.$$

*Remark* (Standard quantified statement).

$$\begin{aligned}
&+_{\mathbb{W}} : \mathbb{W} \times \mathbb{W} \to \mathbb{W}, \\
&\forall a \in \mathbb{W}\ (0 +_{\mathbb{W}} a = a \ \ \wedge \ \ a +_{\mathbb{W}} 0 = a), \\
&\forall a, b \in \mathbb{N}\ (a +_{\mathbb{W}} b = a +_{\mathbb{N}} b).
\end{aligned}$$

*Remark* (Interpretation). If either input is $0$, the new identity clause applies. If both inputs lie in $\mathbb{N}$, the old operation is used. Since $0 \notin \mathbb{N}$, the positive-positive clause does not overlap with the zero clauses.

*Remark* (Dependencies).

- Whole Numbers
- Addition on a Peano System

**Theorem (Addition on $\mathbb{W}$ Is Well-Defined)**

**Theorem 5.4.2** (Addition on $\mathbb{W}$ Is Well-Defined). *The case definition of $+_{\mathbb{W}}$ determines a binary operation on $\mathbb{W}$. Go to proof.*

*Remark* (Standard quantified statement).

$$+_{\mathbb{W}} \text{ is a well-defined binary operation on } \mathbb{W}.$$

*Remark* (Interpretation). Well-definedness is a case check. The zero cases are new, the positive-positive case is inherited from $\mathbb{N}$, and compatibility is automatic because the positive-positive case cannot contain $0$.

*Remark* (Dependencies).

- Whole Numbers
- Addition on $\mathbb{W}$
- Addition Is Well-Defined on a Peano System

**Theorem (Zero Is an Additive Identity on $\mathbb{W}$)**

**Theorem 5.4.3** (Zero Is an Additive Identity on $\mathbb{W}$). *For every $a \in \mathbb{W}$,*

$$0 +_{\mathbb{W}} a = a \qquad \textit{and} \qquad a +_{\mathbb{W}} 0 = a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{W}\ \big(0 +_{\mathbb{W}} a = a \ \ \wedge \ \ a +_{\mathbb{W}} 0 = a\big).$$

*Remark* (Interpretation). This is the structural feature missing from the one-based natural numbers. After adjoining $0$, addition has a genuine identity element.

*Remark* (Dependencies).

- Addition on $\mathbb{W}$

**Theorem (Left Additive Identity on $\mathbb{W}$)**

**Theorem 5.4.4** (Left Additive Identity on $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$0 +_{\mathbb{W}} a = a.$$

*Go to proof.*

*Remark* (Dependencies).

- Zero Is an Additive Identity on $\mathbb{W}$

**Theorem (Right Additive Identity on $\mathbb{W}$)**

**Theorem 5.4.5** (Right Additive Identity on $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$a +_{\mathbb{W}} 0 = a.$$

*Go to proof.*

*Remark* (Dependencies).

- Zero Is an Additive Identity on $\mathbb{W}$

**Theorem (Addition on $\mathbb{W}$ Extends Addition on $\mathbb{N}$)**

**Theorem 5.4.6** (Addition on $\mathbb{W}$ Extends Addition on $\mathbb{N}$). *For all* $a, b \in \mathbb{N}$,

$$a +_{\mathbb{W}} b = a +_{\mathbb{N}} b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{N} \left(a +_{\mathbb{W}} b = a +_{\mathbb{N}} b\right).$$

*Remark* (Interpretation). The extension does not alter sums of positive natural numbers.

*Remark* (Dependencies).

- Addition on $\mathbb{W}$
- Addition on a Peano System

**Theorem (Addition on $\mathbb{W}$ Is Associative)**

**Theorem 5.4.7** (Addition on $\mathbb{W}$ Is Associative). *For all* $a, b, c \in \mathbb{W}$,

$$(a +_{\mathbb{W}} b) +_{\mathbb{W}} c = a +_{\mathbb{W}} (b +_{\mathbb{W}} c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b, c \in \mathbb{W}\ \big((a +_{\mathbb{W}} b) +_{\mathbb{W}} c = a +_{\mathbb{W}} (b +_{\mathbb{W}} c)\big).$$

*Remark* (Interpretation). Associativity follows by reducing zero cases to the identity law and reducing the positive-positive-positive case to associativity on $\mathbb{N}$.

*Remark* (Dependencies).

- Zero Is an Additive Identity on $\mathbb{W}$
- Addition Is Associative

**Theorem (Addition on $\mathbb{W}$ Is Commutative)**

**Theorem 5.4.8** (Addition on $\mathbb{W}$ Is Commutative). *For all* $a, b \in \mathbb{W}$,

$$a +_{\mathbb{W}} b = b +_{\mathbb{W}} a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W}\ \big(a +_{\mathbb{W}} b = b +_{\mathbb{W}} a\big).$$

*Remark* (Interpretation). The zero cases are symmetric by definition, and the positive-positive case is the commutativity theorem already proved for $\mathbb{N}$.

*Remark* (Dependencies).

- Addition on $\mathbb{W}$
- Addition Is Commutative

**Theorem (Addition on $\mathbb{W}$ Satisfies Cancellation)**

**Theorem 5.4.9** (Addition on $\mathbb{W}$ Satisfies Cancellation). *For all* $a, b, c \in \mathbb{W}$,

$$a +_{\mathbb{W}} b = a +_{\mathbb{W}} c \Longrightarrow b = c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b, c \in \mathbb{W}\ \big(a +_{\mathbb{W}} b = a +_{\mathbb{W}} c \Longrightarrow b = c\big).$$

*Remark* (Interpretation). When $a = 0$, cancellation is the identity law. When all relevant entries are positive, it is the natural-number cancellation theorem

*Remark* (Dependencies).

- Zero Is an Additive Identity on $\mathbb{W}$
- Left Cancellation for Addition

## 5.5 Extending Multiplication to $\mathbb{W}$

### Multiplication with Zero

Multiplication on $\mathbb{W}$ keeps the old natural-number multiplication on positive inputs and adds the absorbing behavior of the new element 0. The cases are compatible because $0 \notin \mathbb{N}$.

## The Extended Multiplication Operation

**Definition (Multiplication on $\mathbb{W}$)**

**Definition 5.5.1** (Multiplication on $\mathbb{W}$). Define $\cdot_{\mathbb{W}} : \mathbb{W} \times \mathbb{W} \to \mathbb{W}$ by

$$0 \cdot_{\mathbb{W}} a = 0, \qquad a \cdot_{\mathbb{W}} 0 = 0,$$

and, for $a, b \in \mathbb{N}$,

$$a \cdot_{\mathbb{W}} b = a \cdot_{\mathbb{N}} b.$$

*Remark* (Standard quantified statement).

$$\begin{aligned}
&\cdot_{\mathbb{W}} : \mathbb{W} \times \mathbb{W} \to \mathbb{W}, \\
&\forall a \in \mathbb{W}\ (0 \cdot_{\mathbb{W}} a = 0 \ \ \wedge \ \ a \cdot_{\mathbb{W}} 0 = 0), \\
&\forall a, b \in \mathbb{N}\ (a \cdot_{\mathbb{W}} b = a \cdot_{\mathbb{N}} b).
\end{aligned}$$

*Remark* (Interpretation). The zero cases are new. The positive-positive case is inherited from $\mathbb{N}$. Since $0 \notin \mathbb{N}$, these clauses do not compete.

*Remark* (Dependencies).

- Whole Numbers
- Multiplication on a Peano System

**Theorem (Multiplication on $\mathbb{W}$ Is Well-Defined)**

**Theorem 5.5.2** (Multiplication on $\mathbb{W}$ Is Well-Defined). *The case definition of $\cdot_{\mathbb{W}}$ determines a binary operation on $\mathbb{W}$. Go to proof.*

*Remark* (Standard quantified statement).

$$\cdot_{\mathbb{W}} \text{ is a well-defined binary operation on } \mathbb{W}.$$

*Remark* (Interpretation). The inherited positive case is already well-defined on $\mathbb{N}$, and the new zero cases land in the distinguished element $0 \in \mathbb{W}$.

*Remark* (Dependencies).

- Multiplication on $\mathbb{W}$
- Multiplication Is Well-Defined on a Peano System

**Theorem (Zero Is Absorbing for Multiplication on $\mathbb{W}$)**

**Theorem 5.5.3** (Zero Is Absorbing for Multiplication on $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$0 \cdot_{\mathbb{W}} a = 0 \qquad \textit{and} \qquad a \cdot_{\mathbb{W}} 0 = 0.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{W} \left(0 \cdot_{\mathbb{W}} a = 0 \ \wedge \ a \cdot_{\mathbb{W}} 0 = 0\right).$$

*Remark* (Interpretation). The adjoined element $0$ is absorbing for multiplication on both sides.

*Remark* (Dependencies).

- Multiplication on $\mathbb{W}$

**Theorem (Left Zero Absorption on $\mathbb{W}$)**

**Theorem 5.5.4** (Left Zero Absorption on $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$0 \cdot_{\mathbb{W}} a = 0.$$

*Go to proof.*

*Remark* (Dependencies).

- Zero Is Absorbing for Multiplication on $\mathbb{W}$

**Theorem (Right Zero Absorption on $\mathbb{W}$)**

**Theorem 5.5.5** (Right Zero Absorption on $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$a \cdot_{\mathbb{W}} 0 = 0.$$

*Go to proof.*

*Remark* (Dependencies).

- Zero Is Absorbing for Multiplication on $\mathbb{W}$

**Theorem (One Is a Multiplicative Identity on $\mathbb{W}$)**

**Theorem 5.5.6** (One Is a Multiplicative Identity on $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$1 \cdot_{\mathbb{W}} a = a \qquad \textit{and} \qquad a \cdot_{\mathbb{W}} 1 = a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{W} \left(1 \cdot_{\mathbb{W}} a = a \ \wedge \ a \cdot_{\mathbb{W}} 1 = a\right).$$

*Remark* (Interpretation). The old multiplicative identity remains an identity after adjoining $0$.

*Remark* (Dependencies).

- Multiplication on $\mathbb{W}$
- Multiplication With 1
- One Times $n$

**Theorem (Multiplication on $\mathbb{W}$ Extends Multiplication on $\mathbb{N}$)**

**Theorem 5.5.7** (Multiplication on $\mathbb{W}$ Extends Multiplication on $\mathbb{N}$). *For all* $a, b \in \mathbb{N}$,

$$a \cdot_{\mathbb{W}} b = a \cdot_{\mathbb{N}} b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{N} \left(a \cdot_{\mathbb{W}} b = a \cdot_{\mathbb{N}} b\right).$$

*Remark* (Interpretation). The extension does not alter products of positive natural numbers.

*Remark* (Dependencies).

- Multiplication on $\mathbb{W}$
- Multiplication on a Peano System

**Theorem (Multiplication on $\mathbb{W}$ Is Associative)**

**Theorem 5.5.8** (Multiplication on $\mathbb{W}$ Is Associative). *For all* $a, b, c \in \mathbb{W}$,

$$(a \cdot_{\mathbb{W}} b) \cdot_{\mathbb{W}} c = a \cdot_{\mathbb{W}} (b \cdot_{\mathbb{W}} c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b, c \in \mathbb{W} \left((a \cdot_{\mathbb{W}} b) \cdot_{\mathbb{W}} c = a \cdot_{\mathbb{W}} (b \cdot_{\mathbb{W}} c)\right).$$

*Remark* (Interpretation). Associativity follows from the absorbing zero cases and the inherited associativity of multiplication on positive natural numbers.

*Remark* (Dependencies).

- Zero Is Absorbing for Multiplication on $\mathbb{W}$
- Multiplication Is Associative

**Theorem (Multiplication on $\mathbb{W}$ Is Commutative)**

**Theorem 5.5.9** (Multiplication on $\mathbb{W}$ Is Commutative). *For all* $a, b \in \mathbb{W}$,

$$a \cdot_{\mathbb{W}} b = b \cdot_{\mathbb{W}} a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W} \left(a \cdot_{\mathbb{W}} b = b \cdot_{\mathbb{W}} a\right).$$

*Remark* (Interpretation). The zero cases are symmetric, and the positive-positive case is the inherited commutativity theorem for $\mathbb{N}$.

*Remark* (Dependencies).

- Multiplication on $\mathbb{W}$
- Multiplication Is Commutative

**Theorem (Multiplication Distributes over Addition on $\mathbb{W}$)**

**Theorem 5.5.10** (Multiplication Distributes over Addition on $\mathbb{W}$). *For all* $a, b, c \in \mathbb{W}$,

$$a \cdot_{\mathbb{W}} (b +_{\mathbb{W}} c) = (a \cdot_{\mathbb{W}} b) +_{\mathbb{W}} (a \cdot_{\mathbb{W}} c).$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b, c \in \mathbb{W} \left(a \cdot_{\mathbb{W}} (b +_{\mathbb{W}} c) = (a \cdot_{\mathbb{W}} b) +_{\mathbb{W}} (a \cdot_{\mathbb{W}} c)\right).$$

*Remark* (Interpretation). Distributivity reduces zero cases to the absorbing law and positive cases to the distributive law already proved on $\mathbb{N}$.

*Remark* (Dependencies).

- Addition on $\mathbb{W}$
- Multiplication on $\mathbb{W}$
- Multiplication Distributes Over Addition

## 5.6 Order on $\mathbb{W}$

### Zero as the Least Element

The order on $\mathbb{W}$ extends the order on $\mathbb{N}$ and places the new element 0 below every whole number. Thus the positive part keeps its old order, while the extended system gains a least element.

## The Extended Order

**Definition (Order on $\mathbb{W}$)**

**Definition 5.6.1** (Order on $\mathbb{W}$). Define $\leq_{\mathbb{W}}$ on $\mathbb{W}$ by declaring

$$0 \leq_{\mathbb{W}} a \quad (a \in \mathbb{W}),$$

and, for $a, b \in \mathbb{N}$,

$$a \leq_{\mathbb{W}} b \quad \Longleftrightarrow \quad a \leq_{\mathbb{N}} b.$$

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W} \left(a \leq_{\mathbb{W}} b \Longleftrightarrow (a = 0 \ \vee \ (a, b \in \mathbb{N} \ \wedge \ a \leq_{\mathbb{N}} b))\right).$$

*Remark* (Interpretation). The non-strict order makes $0$ least and otherwise preserves the order already defined on $\mathbb{N}$.

*Remark* (Dependencies).

- Whole Numbers
- Less Than or Equal on a Peano System

**Definition (Strict Order on $\mathbb{W}$)**

**Definition 5.6.2** (Strict Order on $\mathbb{W}$). Define $<_{\mathbb{W}}$ on $\mathbb{W}$ by declaring

$$0 <_{\mathbb{W}} n \quad (n \in \mathbb{N}),$$

and, for $a, b \in \mathbb{N}$,

$$a <_{\mathbb{W}} b \quad \Longleftrightarrow \quad a <_{\mathbb{N}} b.$$

No element of $\mathbb{W}$ is strictly less than $0$.

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W} \left(a <_{\mathbb{W}} b \Longleftrightarrow (a = 0 \ \wedge \ b \in \mathbb{N}) \vee (a, b \in \mathbb{N} \ \wedge \ a <_{\mathbb{N}} b)\right).$$

*Remark* (Interpretation). The non-strict order makes $0$ least. The strict order says exactly that the positive natural numbers lie after $0$ and keep their old strict ordering.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Strict Less Than on a Peano System

**Theorem (Order on $\mathbb{W}$ Extends Order on $\mathbb{N}$)**

**Theorem 5.6.3** (Order on $\mathbb{W}$ Extends Order on $\mathbb{N}$). *For all* $a, b \in \mathbb{N}$,

$$a \leq_{\mathbb{W}} b \Longleftrightarrow a \leq_{\mathbb{N}} b, \qquad a <_{\mathbb{W}} b \Longleftrightarrow a <_{\mathbb{N}} b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{N} \left( (a \leq_{\mathbb{W}} b \iff a \leq_{\mathbb{N}} b) \wedge (a <_{\mathbb{W}} b \iff a <_{\mathbb{N}} b) \right).$$

*Remark* (Interpretation). The extended order changes only comparisons involving the new element $0$.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Strict Order on $\mathbb{W}$

**Theorem (Zero Is the Least Element of $\mathbb{W}$)**

**Theorem 5.6.4** (Zero Is the Least Element of $\mathbb{W}$). *For every* $a \in \mathbb{W}$,

$$0 \leq_{\mathbb{W}} a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{W} \, (0 \leq_{\mathbb{W}} a).$$

*Remark* (Interpretation). The adjoined element $0$ is the first element of the whole-number order.

*Remark* (Dependencies).

- Order on $\mathbb{W}$

**Theorem (Order on $\mathbb{W}$ Is Reflexive)**

**Theorem 5.6.5** (Order on $\mathbb{W}$ Is Reflexive). *For every* $a \in \mathbb{W}$, $a \leq_{\mathbb{W}} a$. *Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{W} \, (a \leq_{\mathbb{W}} a).$$

*Remark* (Interpretation). Reflexivity is immediate for $0$ and inherited from $\mathbb{N}$ for positive elements.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Order Is Reflexive on a Peano System

**Theorem (Order on $\mathbb{W}$ Is Antisymmetric)**

**Theorem 5.6.6** (Order on $\mathbb{W}$ Is Antisymmetric). *For all* $a, b \in \mathbb{W}$,

$$a \leq_{\mathbb{W}} b \text{ and } b \leq_{\mathbb{W}} a \implies a = b.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W} \left(a \leq_{\mathbb{W}} b \;\; \wedge \;\; b \leq_{\mathbb{W}} a \Longrightarrow a = b\right).$$

*Remark* (Interpretation). Antisymmetry follows from the least-element behavior of $0$ and the inherited antisymmetry on positive natural numbers.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Order Is Antisymmetric on a Peano System

**Theorem (Order on $\mathbb{W}$ Is Transitive)**

**Theorem 5.6.7** (Order on $\mathbb{W}$ Is Transitive). *For all* $a, b, c \in \mathbb{W}$,

$$a \leq_{\mathbb{W}} b \;\; \textit{and} \;\; b \leq_{\mathbb{W}} c \Longrightarrow a \leq_{\mathbb{W}} c.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b, c \in \mathbb{W} \left(a \leq_{\mathbb{W}} b \;\; \wedge \;\; b \leq_{\mathbb{W}} c \Longrightarrow a \leq_{\mathbb{W}} c\right).$$

*Remark* (Interpretation). Transitivity reduces to the least-element clause when $0$ occurs and to the transitivity theorem on $\mathbb{N}$ otherwise.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Order Is Transitive on a Peano System

**Theorem (Order on $\mathbb{W}$ Is Total)**

**Theorem 5.6.8** (Order on $\mathbb{W}$ Is Total). *For all* $a, b \in \mathbb{W}$,

$$a \leq_{\mathbb{W}} b \quad \textit{or} \quad b \leq_{\mathbb{W}} a.$$

*Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W} \left(a \leq_{\mathbb{W}} b \;\; \vee \;\; b \leq_{\mathbb{W}} a\right).$$

*Remark* (Interpretation). Any comparison involving $0$ is decided by leastness, while comparisons between positive elements are decided by totality on $\mathbb{N}$.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Trichotomy on a Peano System

**Theorem (Well-Ordering Principle for $\mathbb{W}$)**

**Theorem 5.6.9** (Well-Ordering Principle for $\mathbb{W}$). *Every nonempty subset of $\mathbb{W}$ has a least element with respect to $\leq_{\mathbb{W}}$. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall A \subseteq \mathbb{W} \left(A \neq \varnothing \Longrightarrow \exists m \in A \, \forall a \in A \, (m \leq_{\mathbb{W}} a)\right).$$

*Remark* (Interpretation). A nonempty subset either contains $0$, which is least, or lies in the positive part where the well-ordering principle for $\mathbb{N}$ applies.

*Remark* (Dependencies).

- Order on $\mathbb{W}$
- Zero Is the Least Element of $\mathbb{W}$
- Well-Ordering Principle

## 5.7 Algebraic Structure of $\mathbb{W}$

### Packaging the Extended Operations

After addition, multiplication, and order have been extended, the main point of $\mathbb{W}$ can be recorded structurally. The whole numbers form the zero-based arithmetic system used as the bridge from positive natural numbers to the integer construction.

### Algebraic Packages

**Theorem ($(\mathbb{W}, +, 0)$ Is a Commutative Monoid)**

**Theorem 5.7.1** ($(\mathbb{W}, +, 0)$ Is a Commutative Monoid). *The structure $(\mathbb{W}, +_{\mathbb{W}}, 0)$ is a commutative monoid. Go to proof.*

*Remark* (Standard quantified statement).

$$(\mathbb{W}, +_{\mathbb{W}}, 0) \text{ is a commutative monoid.}$$

*Remark* (Interpretation). Addition on $\mathbb{W}$ has an identity element, is associative, and is commutative.

*Remark* (Dependencies).

- Zero Is an Additive Identity on $\mathbb{W}$
- Addition on $\mathbb{W}$ Is Associative
- Addition on $\mathbb{W}$ Is Commutative

**Theorem ($(\mathbb{W}, \cdot, 1)$ Is a Commutative Monoid)**

**Theorem 5.7.2** ($(\mathbb{W}, \cdot, 1)$ Is a Commutative Monoid). *The structure $(\mathbb{W}, \cdot_{\mathbb{W}}, 1)$ is a commutative monoid. Go to proof.*

*Remark* (Standard quantified statement).

$$(\mathbb{W}, \cdot_{\mathbb{W}}, 1) \text{ is a commutative monoid.}$$

*Remark* (Interpretation). Multiplication on $\mathbb{W}$ has identity $1$, is associative and is commutative.

*Remark* (Dependencies).

- One Is a Multiplicative Identity on $\mathbb{W}$
- Multiplication on $\mathbb{W}$ Is Associative
- Multiplication on $\mathbb{W}$ Is Commutative

**Theorem ($(\mathbb{W}, +, \cdot, 0, 1)$ Is a Commutative Semiring)**

**Theorem 5.7.3** ($(\mathbb{W}, +, \cdot, 0, 1)$ Is a Commutative Semiring). *The structure $(\mathbb{W}, +_{\mathbb{W}}, \cdot_{\mathbb{W}}, 0, 1)$ is a commutative semiring. Go to proof.*

*Remark* (Standard quantified statement).

$$(\mathbb{W}, +_{\mathbb{W}}, \cdot_{\mathbb{W}}, 0, 1) \text{ is a commutative semiring.}$$

*Remark* (Interpretation). The whole numbers have the additive and multiplicative structure needed for ordinary arithmetic, but no subtraction operation is included.

*Remark* (Dependencies).

- $(\mathbb{W}, +, 0)$ Is a Commutative Monoid
- $(\mathbb{W}, \cdot, 1)$ Is a Commutative Monoid
- Multiplication Distributes over Addition on $\mathbb{W}$
- Zero Is Absorbing for Multiplication on $\mathbb{W}$

**Theorem ($\mathbb{W}$ Is a Commutative Ordered Semiring)**

**Theorem 5.7.4** ($\mathbb{W}$ Is a Commutative Ordered Semiring). *With the operations $+_{\mathbb{W}}$ and $\cdot_{\mathbb{W}}$ and the order $\leq_{\mathbb{W}}$, the whole numbers form a commutative ordered semiring. Go to proof.*

*Remark* (Standard quantified statement).

$$(\mathbb{W}, +_{\mathbb{W}}, \cdot_{\mathbb{W}}, 0, 1, \leq_{\mathbb{W}}) \text{ is a commutative ordered semiring.}$$

*Remark* (Interpretation). This packages the algebraic semiring laws together with the order facts: $\mathbb{W}$ is totally ordered, $0$ is least, and the inherited arithmetic is compatible with that order.

*Remark* (Dependencies).

- $(\mathbb{W}, +, \cdot, 0, 1)$ Is a Commutative Semiring

- Order on $\mathbb{W}$ Is Total
- Zero Is the Least Element of $\mathbb{W}$
- Order on $\mathbb{W}$ Extends Order on $\mathbb{N}$

**Theorem ($\mathbb{N}$ Embeds into $\mathbb{W}$ as the Positive Part)**

**Theorem 5.7.5** ($\mathbb{N}$ Embeds into $\mathbb{W}$ as the Positive Part). *The inclusion map identifies $\mathbb{N}$ with the positive part $\mathbb{W} \setminus \{0\}$ and preserves successor, $1$, addition, multiplication, and order. Go to proof.*

*Remark* (Standard quantified statement).

$$\iota : \mathbb{N} \hookrightarrow \mathbb{W}, \qquad \iota(n) = n,$$

identifies $\mathbb{N}$ with $\mathbb{W}\setminus\{0\}$ and preserves successor, $1$, addition, multiplication, and order.

*Remark* (Interpretation). The positive part of $\mathbb{W}$ is exactly the previously constructed natural-number system, with the same arithmetic and order.

*Remark* (Dependencies).

- Whole Numbers
- Addition on $\mathbb{W}$ Extends Addition on $\mathbb{N}$
- Multiplication on $\mathbb{W}$ Extends Multiplication on $\mathbb{N}$
- Order on $\mathbb{W}$ Extends Order on $\mathbb{N}$

**Theorem ($\mathbb{W}$ Has No Additive Inverses Except $0$)**

**Theorem 5.7.6** ($\mathbb{W}$ Has No Additive Inverses Except $0$). *If $a, b \in \mathbb{W}$ and*

$$a +_{\mathbb{W}} b = 0,$$

*then $a = 0$ and $b = 0$. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{W} \left(a +_{\mathbb{W}} b = 0 \Longrightarrow (a = 0 \ \wedge \ b = 0)\right).$$

*Remark* (Interpretation). The only way to sum to zero inside $\mathbb{W}$ is for both summands already to be zero.

*Remark* (Dependencies).

- Addition on $\mathbb{W}$
- Zero Is the Least Element of $\mathbb{W}$
- No Natural Number Equals Itself Plus a Natural Number

**Theorem ($\mathbb{W}$ Is Not a Ring)**

**Theorem 5.7.7** ($\mathbb{W}$ Is Not a Ring). *With the operations $+_{\mathbb{W}}$ and $\cdot_{\mathbb{W}}$, the whole numbers do not form a ring. Go to proof.*

*Remark* (Standard quantified statement).

$$(\mathbb{W}, +_{\mathbb{W}}, \cdot_{\mathbb{W}}, 0, 1) \text{ is not a ring.}$$

*Remark* (Interpretation). The obstruction is not associativity, commutativity, or distributivity. The obstruction is the absence of additive inverses for nonzero whole numbers.

*Remark* (Dependencies).

- $(\mathbb{W}, +, \cdot, 0, 1)$ Is a Commutative Semiring
- $\mathbb{W}$ Has No Additive Inverses Except 0

## 5.8 Transition to $\mathbb{Z}$

### From Zero to Additive Inverses

The whole numbers fix the missing additive identity, but they do not make subtraction available in general. The integer construction repairs this by adjoining additive inverses formally.

### The Next Construction

*Remark* (Exposition). The passage from $\mathbb{N}$ to $\mathbb{W}$ adds the element needed for $0 + a = a$. It does not add solutions to equations such as $a + x = b$ when $b < a$. Thus $\mathbb{W}$ is still not closed under subtraction.

*Remark* (Exposition). The construction of $\mathbb{Z}$ repairs this by adjoining additive inverses formally. One standard route uses ordered pairs from $\mathbb{W} \times \mathbb{W}$, with a pair $(a, b)$ interpreted as a formal difference $a - b$. The equivalence relation then identifies different pairs that represent the same intended difference

*Remark* (Dependencies).

- $(\mathbb{W}, +, 0)$ Is a Commutative Monoid
- $\mathbb{W}$ Has No Additive Inverses Except 0
- $\mathbb{W}$ Is Not a Ring

# Chapter 6

# Integers ($\mathbb{Z}$)

**integers**

Whole Numbers $\to$ **Integers** $\to$ Rational Numbers

## 6.1 Construction of $\mathbb{Z}$

### Integer Prepairs

**Definition**

**Definition 6.1.1** (Integer Prepair). An *integer prepair* is an ordered pair $(a, b) \in \mathbb{W} \times \mathbb{W}$, read as the formal difference $a - b$.

*Remark* (Dependencies).

- Ordered Pair
- Whole Numbers

**Definition**

**Definition 6.1.2** (Integer Prepair Set). The integer prepair set is

$$\operatorname{Pre}\mathbb{Z} := \mathbb{W} \times \mathbb{W}.$$

*Remark* (Dependencies).

- Cartesian Product
- Whole Numbers

### Formal-Difference Equivalence

**Definition**

**Definition 6.1.3** (Formal-Difference Equivalence). For $(a, b), (c, d) \in \operatorname{Pre}\mathbb{Z}$, define

$$(a, b) \sim_{\mathbb{Z}} (c, d) \quad :\Longleftrightarrow \quad a + d = c + b.$$

*Remark* (Dependencies).

- Relation
- Integer Prepair Set
- Addition on $\mathbb{W}$

**Lemma**

**Lemma 6.1.4** (Formal-Difference Equivalence Is Reflexive). *For every* $(a, b) \in \operatorname{Pre} \mathbb{Z}$,

$$(a, b) \sim_{\mathbb{Z}} (a, b).$$

*Remark* (Dependencies).

- Formal-Difference Equivalence
- Addition on $\mathbb{W}$

**Lemma**

**Lemma 6.1.5** (Formal-Difference Equivalence Is Symmetric). *For all prepairs* $(a, b), (c, d)$,

$$(a, b) \sim_{\mathbb{Z}} (c, d) \Longrightarrow (c, d) \sim_{\mathbb{Z}} (a, b).$$

*Remark* (Dependencies).

- Formal-Difference Equivalence
- Addition on $\mathbb{W}$

**Lemma**

**Lemma 6.1.6** (Formal-Difference Equivalence Is Transitive). *For all prepairs* $(a, b), (c, d), (e, f)$,

$$(a, b) \sim_{\mathbb{Z}} (c, d) \wedge (c, d) \sim_{\mathbb{Z}} (e, f) \Longrightarrow (a, b) \sim_{\mathbb{Z}} (e, f).$$

*Remark* (Dependencies).

- Addition on $\mathbb{W}$ Satisfies Cancellation

**Theorem**

**Theorem 6.1.7** (Formal-Difference Equivalence Is an Equivalence Relation). *The relation* $\sim_{\mathbb{Z}}$ *is an equivalence relation on* $\operatorname{Pre} \mathbb{Z}$.

*Remark* (Dependencies).

- Equivalence Relation
- Formal-Difference Equivalence
- Formal-Difference Equivalence Is Reflexive
- Formal-Difference Equivalence Is Symmetric
- Formal-Difference Equivalence Is Transitive

## Definition of $\mathbb{Z}$

**Definition**

**Definition 6.1.8** (Integers). The integers are the quotient

$$\mathbb{Z} := \operatorname{Pre}\mathbb{Z}/\sim_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Quotient Set
- Integer Prepair Set
- Formal-Difference Equivalence

**Definition**

**Definition 6.1.9** (Integer Class Notation). The notation $[a, b]$ denotes the $\sim_{\mathbb{Z}}$-equivalence class of the prepair $(a, b)$.

*Remark* (Dependencies).

- Integers
- Formal-Difference Equivalence

## Zero and One in $\mathbb{Z}$

**Definition**

**Definition 6.1.10** (Zero in $\mathbb{Z}$). Define

$$0_{\mathbb{Z}} := [0, 0].$$

*Remark* (Dependencies).

- Integer Class Notation
- Whole Numbers

**Definition**

**Definition 6.1.11** (One in $\mathbb{Z}$). Define

$$1_{\mathbb{Z}} := [1, 0].$$

*Remark* (Dependencies).

- Integer Class Notation
- Whole Numbers

**Theorem**

**Theorem 6.1.12** (Zero Is an Integer). *The class $0_{\mathbb{Z}}$ is an element of $\mathbb{Z}$.*

*Remark* (Dependencies).

- Integers
- Zero in $\mathbb{Z}$
- Integer Class Notation

**Theorem**

**Theorem 6.1.13** (One Is an Integer). *The class $1_{\mathbb{Z}}$ is an element of $\mathbb{Z}$.*

*Remark* (Dependencies).

- Integers
- One in $\mathbb{Z}$
- Integer Class Notation

**Theorem**

**Theorem 6.1.14** (Zero Is Not One in $\mathbb{Z}$). *In $\mathbb{Z}$,*

$$0_{\mathbb{Z}} \neq 1_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- $0 \notin \mathbb{N}$

## 6.2 Operations on $\mathbb{Z}$-Classes

### Addition on $\mathbb{Z}$-Classes

**Definition**

**Definition 6.2.1** (Addition on $\mathbb{Z}$-Classes). Define addition on representatives by

$$[a, b] + [c, d] := [a + c, b + d].$$

*Remark* (Dependencies).

- Integer Class Notation
- Addition on $\mathbb{W}$

**Lemma**

**Lemma 6.2.2** (Addition Respects Formal-Difference Equivalence on the Left). *If $[a, b] = [a', b']$, then*

$$[a, b] + [c, d] = [a', b'] + [c, d].$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$

- Formal-Difference Equivalence
- Addition on $\mathbb{W}$ Satisfies Cancellation

**Lemma**

**Lemma 6.2.3** (Addition Respects Formal-Difference Equivalence on the Right). *If* $[c, d] = [c', d']$, *then*

$$[a, b] + [c, d] = [a, b] + [c', d'].$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Formal-Difference Equivalence
- Addition on $\mathbb{W}$ Satisfies Cancellation

**Lemma**

**Lemma 6.2.4** (Addition Respects Formal-Difference Equivalence). *If* $[a, b] = [a', b']$ *and* $[c, d] = [c', d']$, *then*

$$[a, b] + [c, d] = [a', b'] + [c', d'].$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Addition Respects Formal-Difference Equivalence on the Left
- Addition Respects Formal-Difference Equivalence on the Right

**Theorem**

**Theorem 6.2.5** (Addition on $\mathbb{Z}$ Is Well-Defined). *Addition determines a well-defined binary operation* $\mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$.

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Addition Respects Formal-Difference Equivalence

**Theorem**

**Theorem 6.2.6** (Addition on $\mathbb{Z}$ Is Associative). *For all* $x, y, z \in \mathbb{Z}$,

$$(x + y) + z = x + (y + z).$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined

- Addition on $\mathbb{W}$ Is Associative

**Theorem**

**Theorem 6.2.7** (Addition on $\mathbb{Z}$ Is Commutative). *For all* $x, y \in \mathbb{Z}$,

$$x + y = y + x.$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined
- Addition on $\mathbb{W}$ Is Commutative

**Theorem**

**Theorem 6.2.8** (Zero Is a Left Additive Identity on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$0_{\mathbb{Z}} + x = x.$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Zero in $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined
- Zero Is an Additive Identity on $\mathbb{W}$

**Corollary**

**Corollary 6.2.9** (Zero Is a Right Additive Identity on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$x + 0_{\mathbb{Z}} = x.$$

*Remark* (Dependencies).

- Zero Is a Left Additive Identity on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.2.10** (Zero Is an Additive Identity on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$0_{\mathbb{Z}} + x = x \qquad \textit{and} \qquad x + 0_{\mathbb{Z}} = x.$$

*Remark* (Dependencies).

- Zero Is a Left Additive Identity on $\mathbb{Z}$
- Zero Is a Right Additive Identity on $\mathbb{Z}$

### Negation on $\mathbb{Z}$-Classes

**Definition**

**Definition 6.2.11** (Negation on $\mathbb{Z}$-Classes). Define negation on representatives by

$$-[a, b] := [b, a].$$

*Remark* (Dependencies).

- Integer Class Notation

**Lemma**

**Lemma 6.2.12** (Negation Respects Formal-Difference Equivalence). *If* $[a, b] = [a', b']$, *then*

$$-[a, b] = -[a', b'].$$

*Remark* (Dependencies).

- Negation on $\mathbb{Z}$
- Formal-Difference Equivalence

**Theorem**

**Theorem 6.2.13** (Negation on $\mathbb{Z}$ Is Well-Defined). *Negation determines a well-defined unary operation* $\mathbb{Z} \to \mathbb{Z}$.

*Remark* (Dependencies).

- Negation on $\mathbb{Z}$
- Negation Respects Formal-Difference Equivalence

**Theorem**

**Theorem 6.2.14** (Negation Gives a Left Additive Inverse on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$(-x) + x = 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Negation on $\mathbb{Z}$
- Addition on $\mathbb{Z}$
- Zero in $\mathbb{Z}$
- Negation on $\mathbb{Z}$ Is Well-Defined
- Addition on $\mathbb{Z}$ Is Well-Defined

**Corollary**

**Corollary 6.2.15** (Negation Gives a Right Additive Inverse on $\mathbb{Z}$). *For every $x \in \mathbb{Z}$,*

$$x + (-x) = 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Negation Gives a Left Additive Inverse on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.2.16** (Every Integer Has an Additive Inverse). *For every $x \in \mathbb{Z}$ there exists $y \in \mathbb{Z}$ such that*

$$x + y = 0_{\mathbb{Z}} \qquad \textit{and} \qquad y + x = 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Negation Gives a Left Additive Inverse on $\mathbb{Z}$
- Negation Gives a Right Additive Inverse on $\mathbb{Z}$

## Subtraction on $\mathbb{Z}$-Classes

**Definition**

**Definition 6.2.17** (Subtraction on $\mathbb{Z}$). For $x, y \in \mathbb{Z}$, define

$$x - y := x + (-y).$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Negation on $\mathbb{Z}$

**Theorem**

**Theorem 6.2.18** (Subtraction on $\mathbb{Z}$ Is Well-Defined). *Subtraction determines a well-defined binary operation $\mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$.*

*Remark* (Dependencies).

- Subtraction on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined
- Negation on $\mathbb{Z}$ Is Well-Defined

### Multiplication on $\mathbb{Z}$-Classes

**Definition**

**Definition 6.2.19** (Multiplication on $\mathbb{Z}$-Classes). Define multiplication on representatives by

$$[a, b] \cdot [c, d] := [ac + bd, ad + bc].$$

*Remark* (Dependencies).

- Integer Class Notation
- Addition on $\mathbb{W}$
- Multiplication on $\mathbb{W}$

**Lemma**

**Lemma 6.2.20** (Multiplication Respects Formal-Difference Equivalence on the Left). *If* $[a, b] = [a', b']$, *then*

$$[a, b] \cdot [c, d] = [a', b'] \cdot [c, d].$$

*Remark* (Dependencies).

- Addition on $\mathbb{W}$ Satisfies Cancellation
- Order on $\mathbb{W}$ Is Total

**Lemma**

**Lemma 6.2.21** (Multiplication Respects Formal-Difference Equivalence on the Right). *If* $[c, d] = [c', d']$, *then*

$$[a, b] \cdot [c, d] = [a, b] \cdot [c', d'].$$

*Remark* (Dependencies).

- Addition on $\mathbb{W}$ Satisfies Cancellation
- Order on $\mathbb{W}$ Is Total

**Lemma**

**Lemma 6.2.22** (Multiplication Respects Formal-Difference Equivalence). *If* $[a, b] = [a', b']$ *and* $[c, d] = [c', d']$, *then*

$$[a, b] \cdot [c, d] = [a', b'] \cdot [c', d'].$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Z}$
- Multiplication Respects Formal-Difference Equivalence on the Left

- Multiplication Respects Formal-Difference Equivalence on the Right

**Theorem**

**Theorem 6.2.23** (Multiplication on $\mathbb{Z}$ Is Well-Defined). *Multiplication determines a well-defined binary operation $\mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$.*

*Remark* (Dependencies).

- Multiplication on $\mathbb{Z}$
- Multiplication Respects Formal-Difference Equivalence

**Theorem**

**Theorem 6.2.24** (Multiplication on $\mathbb{Z}$ Is Associative). *For all $x, y, z \in \mathbb{Z}$,*

$$(x \cdot y) \cdot z = x \cdot (y \cdot z).$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Multiplication on $\mathbb{W}$ Is Associative

**Theorem**

**Theorem 6.2.25** (Multiplication on $\mathbb{Z}$ Is Commutative). *For all $x, y \in \mathbb{Z}$,*

$$x \cdot y = y \cdot x.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Multiplication on $\mathbb{W}$ Is Commutative

**Theorem**

**Theorem 6.2.26** (One Is a Left Multiplicative Identity on $\mathbb{Z}$). *For every $x \in \mathbb{Z}$,*

$$1_{\mathbb{Z}} \cdot x = x.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Z}$
- One in $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined

- One Is a Multiplicative Identity on $\mathbb{W}$

**Corollary**

**Corollary 6.2.27** (One Is a Right Multiplicative Identity on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$x \cdot 1_{\mathbb{Z}} = x.$$

*Remark* (Dependencies).

- One Is a Left Multiplicative Identity on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.2.28** (One Is a Multiplicative Identity on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$1_{\mathbb{Z}} \cdot x = x \qquad \textit{and} \qquad x \cdot 1_{\mathbb{Z}} = x.$$

*Remark* (Dependencies).

- One Is a Left Multiplicative Identity on $\mathbb{Z}$
- One Is a Right Multiplicative Identity on $\mathbb{Z}$

## Distributivity on $\mathbb{Z}$-Classes

**Theorem**

**Theorem 6.2.29** (Left Distributivity on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$x \cdot (y + z) = x \cdot y + x \cdot z.$$

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$
- Multiplication Distributes over Addition on $\mathbb{W}$

**Corollary**

**Corollary 6.2.30** (Right Distributivity on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$(y + z) \cdot x = y \cdot x + z \cdot x.$$

*Remark* (Dependencies).

- Left Distributivity on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.2.31** (Multiplication Distributes over Addition on $\mathbb{Z}$). *Multiplication distributes over addition on both sides in $\mathbb{Z}$.*

*Remark* (Dependencies).

- Addition on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$
- Left Distributivity on $\mathbb{Z}$
- Right Distributivity on $\mathbb{Z}$

## Class-Level Ring Summary

**Theorem**

**Theorem 6.2.32** ($\mathbb{Z}$ Is an Additive Abelian Group). *The structure $(\mathbb{Z}, +, 0_{\mathbb{Z}})$ is an abelian group.*

*Remark* (Dependencies).

- Group
- Addition on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Associative
- Addition on $\mathbb{Z}$ Is Commutative
- Zero Is an Additive Identity on $\mathbb{Z}$
- Every Integer Has an Additive Inverse

**Theorem**

**Theorem 6.2.33** ($\mathbb{Z}$ Is a Multiplicative Commutative Monoid). *The structure $(\mathbb{Z}, \cdot, 1_{\mathbb{Z}})$ is a commutative monoid.*

*Remark* (Dependencies).

- Commutative Monoid
- Multiplication on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Associative
- Multiplication on $\mathbb{Z}$ Is Commutative
- One Is a Multiplicative Identity on $\mathbb{Z}$

**Theorem**

**Theorem 6.2.34** ($\mathbb{Z}$ Is a Commutative Ring). *The structure* $(\mathbb{Z}, +, \cdot, 0_{\mathbb{Z}}, 1_{\mathbb{Z}})$ *is a commutative ring.*

*Remark* (Dependencies).

- Commutative Ring
- $\mathbb{Z}$ Is an Additive Abelian Group
- $\mathbb{Z}$ Is a Multiplicative Commutative Monoid
- Multiplication Distributes over Addition on $\mathbb{Z}$

## 6.3 Algebraic Structure of $\mathbb{Z}$

### Integral Domain Structure

**Theorem**

**Theorem 6.3.1** ($\mathbb{Z}$ Has No Zero Divisors). *For all* $x, y \in \mathbb{Z}$,

$$x \cdot y = 0 \Longrightarrow x = 0 \text{ or } y = 0.$$

*Remark* (Dependencies).

- Order on $\mathbb{W}$ Is Total
- Cancellation for Multiplication

**Theorem**

**Theorem 6.3.2** ($\mathbb{Z}$ Is an Integral Domain). *The integers form an integral domain:* $\mathbb{Z}$ *is a commutative ring,* $0 \neq 1$*, and* $\mathbb{Z}$ *has no zero divisors.*

*Remark* (Dependencies).

- $\mathbb{Z}$ Is a Commutative Ring
- Zero Is Not One in $\mathbb{Z}$
- $\mathbb{Z}$ Has No Zero Divisors

**Corollary**

**Corollary 6.3.3** (Multiplicative Cancellation on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$z \neq 0 \wedge z \cdot x = z \cdot y \Longrightarrow x = y.$$

*Remark* (Dependencies).

- $\mathbb{Z}$ Has No Zero Divisors
- Multiplication on $\mathbb{Z}$ Is Commutative

### Derived Laws on $\mathbb{Z}$

**Corollary**

**Corollary 6.3.4** (Zero Absorption on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$0 \cdot x = 0 \qquad \textit{and} \qquad x \cdot 0 = 0.$$

*Remark* (Dependencies).

- $\mathbb{Z}$ Is a Commutative Ring
- Zero in $\mathbb{Z}$

**Corollary**

**Corollary 6.3.5** (Sign Rule on $\mathbb{Z}$). *For all* $x, y \in \mathbb{Z}$,

$$(-x) \cdot y = -(x \cdot y) \qquad \textit{and} \qquad x \cdot (-y) = -(x \cdot y).$$

*Remark* (Dependencies).

- $\mathbb{Z}$ Is a Commutative Ring
- Negation on $\mathbb{Z}$ Is Well-Defined
- Multiplication Distributes over Addition on $\mathbb{Z}$

**Corollary**

**Corollary 6.3.6** (Negative One Rule on $\mathbb{Z}$). *For every* $x \in \mathbb{Z}$,

$$(-1) \cdot x = -x.$$

*Remark* (Dependencies).

- Sign Rule on $\mathbb{Z}$
- One in $\mathbb{Z}$

**Corollary**

**Corollary 6.3.7** (Product of Negatives on $\mathbb{Z}$). *For all* $x, y \in \mathbb{Z}$,

$$(-x) \cdot (-y) = x \cdot y.$$

*Remark* (Dependencies).

- Sign Rule on $\mathbb{Z}$
- Negative One Rule on $\mathbb{Z}$

## 6.4 Order on $\mathbb{Z}$

### Positivity on $\mathbb{Z}$

**Definition**

**Definition 6.4.1** (Positive Integer). An integer $x \in \mathbb{Z}$ is *positive* if $x = [a, b]$ for some $a, b \in \mathbb{W}$ with $b < a$.

*Remark* (Dependencies).

- Integer Class Notation
- Whole Numbers
- Order on $\mathbb{W}$

**Lemma**

**Lemma 6.4.2** (Positivity Respects Formal-Difference Equivalence). *If* $[a, b] = [a', b']$, *then*

$$b < a \iff b' < a'.$$

*Remark* (Dependencies).

- Addition on $\mathbb{W}$ Satisfies Cancellation
- Order on $\mathbb{W}$ Is Total

**Theorem**

**Theorem 6.4.3** (Positivity on $\mathbb{Z}$ Is Well-Defined). *The predicate "$x$ is positive" is well-defined on equivalence classes in $\mathbb{Z}$.*

*Remark* (Dependencies).

- Positive Integer
- Formal-Difference Equivalence
- Positivity Respects Formal-Difference Equivalence

**Definition**

**Definition 6.4.4** (Strict Order on $\mathbb{Z}$). For $x, y \in \mathbb{Z}$, define

$$x < y \quad :\iff \quad y - x \text{ is positive.}$$

*Remark* (Dependencies).

- Subtraction on $\mathbb{Z}$
- Positive Integer

**Definition**

**Definition 6.4.5** (Order on $\mathbb{Z}$). For $x, y \in \mathbb{Z}$, define

$$x \leq y \quad :\Longleftrightarrow \quad x < y \text{ or } x = y.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$

**Proposition**

**Proposition 6.4.6** (Product of Positives Is Positive). *For all* $x, y \in \mathbb{Z}$,

$$0 < x \wedge 0 < y \Longrightarrow 0 < x \cdot y.$$

*Remark* (Dependencies).

- Positive Integer
- Multiplication on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Order on $\mathbb{W}$

## Canonical Form and Sign

**Theorem**

**Theorem 6.4.7** (Canonical Form of Integers). *For every* $x \in \mathbb{Z}$, *exactly one of the following holds:*

$$x = [n, 0] \text{ for some nonzero } n \in \mathbb{W}, \qquad x = [0, 0], \qquad x = [0, n] \text{ for some nonzero } n \in \mathbb{W}.$$

*Remark* (Dependencies).

- Integers
- Integer Class Notation
- Formal-Difference Equivalence
- Whole Numbers
- Order on $\mathbb{W}$ Is Total

**Corollary**

**Corollary 6.4.8** (Integers Split by Sign). *Every integer is nonnegative or the negative of a nonnegative integer, and the two descriptions overlap only at* $0_{\mathbb{Z}}$.

*Remark* (Dependencies).

- Canonical Form of Integers
- Positive Integer
- Zero in $\mathbb{Z}$

**Corollary**

**Corollary 6.4.9** (Image of the Embedding Is the Nonnegative Integers). *The image of the canonical embedding $\mathbb{W} \to \mathbb{Z}$ is exactly the set of nonnegative integers.*

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- Integers Split by Sign
- Embedding $\mathbb{W}$ into $\mathbb{Z}$ Is Injective

**Theorem**

**Theorem 6.4.10** (Sign Trichotomy on $\mathbb{Z}$). *For every $x \in \mathbb{Z}$, exactly one of*

$$x < 0, \qquad x = 0, \qquad 0 < x$$

*holds.*

*Remark* (Dependencies).

- Canonical Form of Integers
- Order on $\mathbb{W}$ Is Total

**Theorem**

**Theorem 6.4.11** (Trichotomy on $\mathbb{Z}$). *For all $x, y \in \mathbb{Z}$, exactly one of*

$$x < y, \qquad x = y, \qquad y < x$$

*holds.*

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$
- Subtraction on $\mathbb{Z}$
- Sign Trichotomy on $\mathbb{Z}$

### Total Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.12** (Strict Order on $\mathbb{Z}$ Is Irreflexive). *For every $x \in \mathbb{Z}$, not $x < x$.*

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$
- Subtraction on $\mathbb{Z}$
- Zero Is an Integer

**Theorem**

**Theorem 6.4.13** (Strict Order on $\mathbb{Z}$ Is Asymmetric). *For all $x, y \in \mathbb{Z}$,*

$$x < y \Longrightarrow \neg(y < x).$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$
- Strict Order on $\mathbb{Z}$ Is Irreflexive
- Trichotomy on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.14** (Strict Order on $\mathbb{Z}$ Is Transitive). *For all $x, y, z \in \mathbb{Z}$,*

$$x < y \wedge y < z \Longrightarrow x < z.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$
- Subtraction on $\mathbb{Z}$
- Positive Integer
- Addition Preserves Strict Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.15** (Order on $\mathbb{Z}$ Is Reflexive). *For every $x \in \mathbb{Z}$, $x \leq x$.*

*Remark* (Dependencies).

- Order on $\mathbb{Z}$
- Strict Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.16** (Order on $\mathbb{Z}$ Is Antisymmetric). *For all* $x, y \in \mathbb{Z}$,

$$x \leq y \wedge y \leq x \Longrightarrow x = y.$$

*Remark* (Dependencies).

- Order on $\mathbb{Z}$
- Strict Order on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.17** (Order on $\mathbb{Z}$ Is Transitive). *For all* $x, y, z \in \mathbb{Z}$,

$$x \leq y \wedge y \leq z \Longrightarrow x \leq z.$$

*Remark* (Dependencies).

- Order on $\mathbb{Z}$
- Strict Order on $\mathbb{Z}$
- Strict Order on $\mathbb{Z}$ Is Transitive
- Trichotomy on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.18** (Order on $\mathbb{Z}$ Is Total). *For all* $x, y \in \mathbb{Z}$,

$$x \leq y \text{ or } y \leq x.$$

*Remark* (Dependencies).

- Order on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.19** ($\mathbb{Z}$ Is a Totally Ordered Set). *The ordered pair* $(\mathbb{Z}, \leq)$ *is a totally ordered set.*

*Remark* (Dependencies).

- Ordered Set
- Total Order
- Order on $\mathbb{Z}$
- Order on $\mathbb{Z}$ Is Reflexive

- Order on $\mathbb{Z}$ Is Antisymmetric
- Order on $\mathbb{Z}$ Is Transitive
- Order on $\mathbb{Z}$ Is Total

### Addition Order Compatibility on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.20** (Left Addition Preserves Strict Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$x < y \implies z + x < z + y.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$
- Addition on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined

**Corollary**

**Corollary 6.4.21** (Right Addition Preserves Strict Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$x < y \implies x + z < y + z.$$

*Remark* (Dependencies).

- Left Addition Preserves Strict Order on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.4.22** (Addition Preserves Strict Order on $\mathbb{Z}$). *Addition preserves strict order on both sides in* $\mathbb{Z}$.

*Remark* (Dependencies).

- Left Addition Preserves Strict Order on $\mathbb{Z}$
- Right Addition Preserves Strict Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.23** (Left Addition Preserves Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$x \leq y \implies z + x \leq z + y.$$

*Remark* (Dependencies).

- Order on $\mathbb{Z}$

- Left Addition Preserves Strict Order on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined

**Corollary**

**Corollary 6.4.24** (Right Addition Preserves Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$x \leq y \implies x + z \leq y + z.$$

*Remark* (Dependencies).

- Left Addition Preserves Order on $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.4.25** (Addition Preserves Order on $\mathbb{Z}$). *Addition preserves non-strict order on both sides in* $\mathbb{Z}$.

*Remark* (Dependencies).

- Left Addition Preserves Order on $\mathbb{Z}$
- Right Addition Preserves Order on $\mathbb{Z}$

**Corollary**

**Corollary 6.4.26** (Addition Reflects Order on $\mathbb{Z}$). *Addition by a fixed integer reflects both strict and non-strict order.*

*Remark* (Dependencies).

- Addition Preserves Strict Order on $\mathbb{Z}$
- Addition Preserves Order on $\mathbb{Z}$
- Negation Is a Left Additive Inverse on $\mathbb{Z}$

### Multiplication Order Compatibility on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.27** (Left Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$0 < z \wedge x < y \implies z \cdot x < z \cdot y.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Z}$
- Product of Positives Is Positive

- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Multiplication Distributes over Addition on $\mathbb{Z}$

**Corollary**

**Corollary 6.4.28** (Right Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$0 < z \wedge x < y \Longrightarrow x \cdot z < y \cdot z.$$

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.4.29** (Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$). *Multiplication by a positive integer preserves strict order on both sides.*

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$
- Right Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.30** (Left Multiplication by Positives Preserves Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$0 < z \wedge x \leq y \Longrightarrow z \cdot x \leq z \cdot y.$$

*Remark* (Dependencies).

- Order on $\mathbb{Z}$
- Left Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$

**Corollary**

**Corollary 6.4.31** (Right Multiplication by Positives Preserves Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$0 < z \wedge x \leq y \Longrightarrow x \cdot z \leq y \cdot z.$$

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Order on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 6.4.32** (Multiplication by Positives Preserves Order on $\mathbb{Z}$). *Multiplication by a positive integer preserves non-strict order on both sides.*

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Order on $\mathbb{Z}$
- Right Multiplication by Positives Preserves Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.33** (Multiplication by Negatives Reverses Strict Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$z < 0 \wedge x < y \Longrightarrow z \cdot y < z \cdot x.$$

*Remark* (Dependencies).

- Multiplication by Positives Preserves Strict Order on $\mathbb{Z}$
- Sign Rule on $\mathbb{Z}$
- Strict Order on $\mathbb{Z}$ Is Asymmetric

**Corollary**

**Corollary 6.4.34** (Multiplication by Negatives Reverses Order on $\mathbb{Z}$). *For all* $x, y, z \in \mathbb{Z}$,

$$z < 0 \wedge x \leq y \Longrightarrow z \cdot y \leq z \cdot x.$$

*Remark* (Dependencies).

- Multiplication by Negatives Reverses Strict Order on $\mathbb{Z}$
- Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.35** (Multiplication by Zero Collapses Order on $\mathbb{Z}$). *For all* $x, y \in \mathbb{Z}$,

$$0 \cdot x = 0 \cdot y.$$

*Remark* (Dependencies).

- Zero Absorption on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined

### Absolute Value on $\mathbb{Z}$

**Definition**

**Definition 6.4.36** (Absolute Value on $\mathbb{Z}$). Define

$$|x| := \begin{cases} x, & 0 \leq x, \\ -x, & x < 0. \end{cases}$$

*Remark* (Dependencies).

- Order on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$
- Negation on $\mathbb{Z}$ Is Well-Defined

**Theorem**

**Theorem 6.4.37** (Absolute Value Is Nonnegative). *For every* $x \in \mathbb{Z}$,

$$0 \leq |x|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.38** (Absolute Value Vanishes Only at Zero). *For every* $x \in \mathbb{Z}$,

$$|x| = 0 \iff x = 0.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.39** (Absolute Value Is Symmetric under Negation). *For every* $x \in \mathbb{Z}$,

$$|-x| = |x|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$
- Sign Rule on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.40** (Absolute Value Is Multiplicative). *For all* $x, y \in \mathbb{Z}$,

$$|x \cdot y| = |x| \cdot |y|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Z}$
- Product of Positives Is Positive
- Product of Negatives on $\mathbb{Z}$
- Multiplication by Positives Preserves Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.41** (Bounding by Absolute Value). *For every* $x \in \mathbb{Z}$,

$$-|x| \leq x \leq |x|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$
- Absolute Value Is Symmetric under Negation

**Theorem**

**Theorem 6.4.42** (Triangle Inequality on $\mathbb{Z}$). *For all* $x, y \in \mathbb{Z}$,

$$|x + y| \leq |x| + |y|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Z}$
- Bounding by Absolute Value
- Addition Preserves Order on $\mathbb{Z}$

## Well-Ordering Fails on $\mathbb{Z}$

**Theorem**

**Theorem 6.4.43** ($\mathbb{Z}$ Is Not Well-Ordered). *The ordered set* $(\mathbb{Z}, \leq)$ *is not a well-order.*

*Remark* (Dependencies).

- Well-Ordered Set

- Order on $\mathbb{Z}$
- Zero in $\mathbb{Z}$
- One in $\mathbb{Z}$
- Trichotomy on $\mathbb{Z}$

## 6.5 Embedding $\mathbb{W}$ into $\mathbb{Z}$

### Embedding $\mathbb{W}$ into $\mathbb{Z}$

**Definition**

**Definition 6.5.1** (Embedding of $\mathbb{W}$ into $\mathbb{Z}$). Define

$$\iota : \mathbb{W} \to \mathbb{Z}, \qquad \iota(n) := [n, 0].$$

*Remark* (Dependencies).

- Whole Numbers
- Integers
- Integer Class Notation
- Function

**Theorem**

**Theorem 6.5.2** (Embedding Preserves Zero). *The canonical embedding satisfies*

$$\iota(0_{\mathbb{W}}) = 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- Zero in $\mathbb{Z}$
- Whole Numbers

**Theorem**

**Theorem 6.5.3** (Embedding Preserves One). *The canonical embedding satisfies*

$$\iota(1_{\mathbb{W}}) = 1_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- One in $\mathbb{Z}$

- Whole Numbers

**Theorem**

**Theorem 6.5.4** (Embedding $\mathbb{W}$ into $\mathbb{Z}$ Is Injective). *For all* $a, b \in \mathbb{W}$,

$$\iota(a) = \iota(b) \implies a = b.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- Injective Function
- Formal-Difference Equivalence
- Addition on $\mathbb{W}$ Satisfies Cancellation

**Theorem**

**Theorem 6.5.5** (Embedding Preserves Addition). *For all* $a, b \in \mathbb{W}$,

$$\iota(a + b) = \iota(a) + \iota(b).$$

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- Addition on $\mathbb{W}$
- Addition on $\mathbb{Z}$ Is Well-Defined

**Theorem**

**Theorem 6.5.6** (Embedding Preserves Multiplication). *For all* $a, b \in \mathbb{W}$,

$$\iota(a \cdot b) = \iota(a) \cdot \iota(b).$$

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- Multiplication on $\mathbb{W}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined

**Theorem**

**Theorem 6.5.7** (Embedding Preserves Order). *For all* $a, b \in \mathbb{W}$,

$$a \leq b \iff \iota(a) \leq \iota(b).$$

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$

- Order on $\mathbb{W}$
- Order on $\mathbb{Z}$

**Theorem**

**Theorem 6.5.8** ($\mathbb{W}$ Embeds into $\mathbb{Z}$). *The map $\iota : \mathbb{W} \to \mathbb{Z}$ is an injective order-preserving semiring homomorphism.*

*Remark* (Dependencies).

- Embedding of $\mathbb{W}$ into $\mathbb{Z}$
- Embedding $\mathbb{W}$ into $\mathbb{Z}$ Is Injective
- Embedding Preserves Addition
- Embedding Preserves Multiplication
- Embedding Preserves Order

## 6.6 Divisibility in $\mathbb{Z}$

### Divisibility in $\mathbb{Z}$

**Definition**

**Definition 6.6.1** (Divisibility in $\mathbb{Z}$). For $a, b \in \mathbb{Z}$, define

$$a \mid b \quad :\Longleftrightarrow \quad \exists c \in \mathbb{Z}\,(b = a \cdot c).$$

*Remark* (Dependencies).

- Integers
- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Existential Formula

### Basic Divisibility Laws

**Proposition**

**Proposition 6.6.2** (Divisibility Is Reflexive on $\mathbb{Z}$). *For every $a \in \mathbb{Z}$,*

$$a \mid a.$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- One in $\mathbb{Z}$
- $\mathbb{Z}$ Multiplicative Commutative Monoid

**Proposition**

**Proposition 6.6.3** (Divisibility Is Transitive on $\mathbb{Z}$). *For all* $a, b, c \in \mathbb{Z}$,

$$a \mid b \wedge b \mid c \Longrightarrow a \mid c.$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Associative

**Proposition**

**Proposition 6.6.4** (One and Negative One Divide Every Integer). *For every* $a \in \mathbb{Z}$,

$$1 \mid a \quad \textit{and} \quad (-1) \mid a.$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- One in $\mathbb{Z}$
- Negative One Rule on $\mathbb{Z}$

**Proposition**

**Proposition 6.6.5** (Every Integer Divides Zero). *For every* $a \in \mathbb{Z}$,

$$a \mid 0.$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Zero Absorption on $\mathbb{Z}$

**Proposition**

**Proposition 6.6.6** (Divisibility Is Compatible with Negation on the Divisor). *For all* $a, b \in \mathbb{Z}$,

$$a \mid b \Longleftrightarrow (-a) \mid b.$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Sign Rule on $\mathbb{Z}$

**Proposition**

**Proposition 6.6.7** (Divisibility Is Compatible with Negation on the Dividend). *For all* $a, b \in \mathbb{Z}$,

$$a \mid b \iff a \mid (-b).$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Sign Rule on $\mathbb{Z}$

**Proposition**

**Proposition 6.6.8** (Divisibility Is Compatible with Negation on $\mathbb{Z}$). *For all* $a, b \in \mathbb{Z}$,

$$a \mid b \iff (-a) \mid b \iff a \mid (-b).$$

*Remark* (Dependencies).

- Divisibility Is Compatible with Negation on the Divisor
- Divisibility Is Compatible with Negation on the Dividend

**Proposition**

**Proposition 6.6.9** (Divisibility Is Compatible with Addition on $\mathbb{Z}$). *For all* $a, b, c \in \mathbb{Z}$,

$$a \mid b \wedge a \mid c \implies a \mid (b + c).$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Addition on $\mathbb{Z}$ Is Well-Defined
- Multiplication Distributes over Addition on $\mathbb{Z}$

**Proposition**

**Proposition 6.6.10** (Divisibility Respects Linear Combinations on $\mathbb{Z}$). *For all* $a, b, c, x, y \in \mathbb{Z}$,

$$a \mid b \wedge a \mid c \implies a \mid (b \cdot x + c \cdot y).$$

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Divisibility Is Compatible with Addition on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Associative
- Multiplication Distributes over Addition on $\mathbb{Z}$

## Units in $\mathbb{Z}$

**Definition**

**Definition 6.6.11** (Unit in $\mathbb{Z}$). An integer $u \in \mathbb{Z}$ is a *unit* if

$$\exists v \in \mathbb{Z}\,(u \cdot v = 1).$$

*Remark* (Dependencies).

- Integers
- One in $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Existential Formula

**Theorem**

**Theorem 6.6.12** (The Units of $\mathbb{Z}$ Are Exactly $\pm 1$). *An integer $u$ is a unit if and only if*

$$u = 1 \quad \textit{or} \quad u = -1.$$

*Remark* (Dependencies).

- Unit in $\mathbb{Z}$
- $\mathbb{Z}$ Is an Integral Domain
- Trichotomy on $\mathbb{Z}$
- Negative One Rule on $\mathbb{Z}$

**Definition**

**Definition 6.6.13** (Associates in $\mathbb{Z}$). Integers $a, b \in \mathbb{Z}$ are *associates* if $a = u \cdot b$ for some unit $u \in \mathbb{Z}$.

*Remark* (Dependencies).

- Unit in $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined

## Division Algorithm

**Theorem**

**Theorem 6.6.14** (Division Algorithm on $\mathbb{Z}$). *If $a, b \in \mathbb{Z}$ and $b \neq 0$, then there exist unique $q, r \in \mathbb{Z}$ such that*

$$a = bq + r \qquad \textit{and} \qquad 0 \leq r < |b|.$$

*Remark* (Dependencies).

- Well-Ordering Principle
- Multiplicative Cancellation on $\mathbb{Z}$
- $\mathbb{Z}$ Is Not Well-Ordered

### Greatest Common Divisor

**Definition**

**Definition 6.6.15** (Greatest Common Divisor in $\mathbb{Z}$). A greatest common divisor $\gcd(a, b)$ is the positive common divisor of $a$ and $b$ divisible by every other common divisor.

*Remark* (Dependencies).

- Divisibility in $\mathbb{Z}$
- Positive Integer
- Order on $\mathbb{Z}$

### Bezout Identity

**Theorem**

**Theorem 6.6.16** (Bezout Identity on $\mathbb{Z}$). *For $a, b \in \mathbb{Z}$, there exist $x, y \in \mathbb{Z}$ such that*

$$\gcd(a, b) = a \cdot x + b \cdot y.$$

*Remark* (Dependencies).

- Well-Ordering Principle

## 6.7 Transition to $\mathbb{Q}$

*Remark* (Why Rational Numbers Are Introduced). The integers have additive inverses, but they do not have multiplicative inverses for every nonzero element. The rational numbers are introduced by adjoining those missing inverses through equivalence classes of integer pairs with nonzero second coordinate.

# Chapter 7

# Rational Numbers ($\mathbb{Q}$)

**rationals**

Integers ($\mathbb{Z}$) $\rightarrow$ **Rationals** ($\mathbb{Q}$) $\rightarrow$ Real Numbers ($\mathbb{R}$)

## 7.1 Construction of $\mathbb{Q}$

### Prefractions

**Definition**

**Definition 7.1.1** (Nonzero Integers). The set of nonzero integers is

$$\mathbb{Z}^* := \mathbb{Z} \setminus \{0_{\mathbb{Z}}\}.$$

*Remark* (Dependencies).

- Integers
- Zero in $\mathbb{Z}$

**Definition**

**Definition 7.1.2** (Prefraction). A *prefraction* is an ordered pair $(a, b) \in \mathbb{Z} \times \mathbb{Z}^*$, read as the formal quotient $a/b$.

*Remark* (Dependencies).

- Ordered Pair
- Integers
- Nonzero Integers

**Definition**

**Definition 7.1.3** (Prefraction Set). The prefraction set is

$$\operatorname{Pre}\mathbb{Q} := \mathbb{Z} \times \mathbb{Z}^*.$$

*Remark* (Dependencies).

- Cartesian Product
- Integers
- Nonzero Integers

### Fraction Equivalence

**Definition**

**Definition 7.1.4** (Fraction Equivalence). For $(a, b), (c, d) \in \operatorname{Pre} \mathbb{Q}$, define

$$(a, b) \sim_{\mathbb{Q}} (c, d) \quad :\Longleftrightarrow \quad a \cdot d = b \cdot c.$$

*Remark* (Dependencies).

- Relation
- Prefraction Set
- Multiplication on $\mathbb{Z}$

**Lemma**

**Lemma 7.1.5** (Fraction Equivalence Is Reflexive). *For every* $(a, b) \in \operatorname{Pre} \mathbb{Q}$,

$$(a, b) \sim_{\mathbb{Q}} (a, b).$$

*Remark* (Dependencies).

- Fraction Equivalence
- Multiplication on $\mathbb{Z}$ Is Commutative

**Lemma**

**Lemma 7.1.6** (Fraction Equivalence Is Symmetric). *For all prefractions* $(a, b), (c, d)$,

$$(a, b) \sim_{\mathbb{Q}} (c, d) \Longrightarrow (c, d) \sim_{\mathbb{Q}} (a, b).$$

*Remark* (Dependencies).

- Fraction Equivalence
- Multiplication on $\mathbb{Z}$ Is Commutative

**Lemma**

**Lemma 7.1.7** (Fraction Equivalence Is Transitive). *For all prefractions* $(a, b), (c, d), (e, f)$,

$$(a, b) \sim_{\mathbb{Q}} (c, d) \wedge (c, d) \sim_{\mathbb{Q}} (e, f) \Longrightarrow (a, b) \sim_{\mathbb{Q}} (e, f).$$

*Remark* (Dependencies).

- Multiplicative Cancellation on $\mathbb{Z}$

**Theorem**

**Theorem 7.1.8** (Fraction Equivalence Is an Equivalence Relation). *The relation $\sim_{\mathbb{Q}}$ is an equivalence relation on $\operatorname{Pre}\mathbb{Q}$.*

*Remark* (Dependencies).

- Equivalence Relation
- Fraction Equivalence
- Fraction Equivalence Is Reflexive
- Fraction Equivalence Is Symmetric
- Fraction Equivalence Is Transitive

## Definition of $\mathbb{Q}$

**Definition**

**Definition 7.1.9** (Rational Numbers). The rational numbers are the quotient

$$\mathbb{Q} := \operatorname{Pre}\mathbb{Q}/\sim_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Quotient Set
- Prefraction Set
- Fraction Equivalence

**Definition**

**Definition 7.1.10** (Rational Class Notation). The notation $[a, b]$ denotes the $\sim_{\mathbb{Q}}$-equivalence class of the prefraction $(a, b)$, where $b \neq 0_{\mathbb{Z}}$.

*Remark* (Dependencies).

- Rational Numbers
- Fraction Equivalence

**Proposition**

**Proposition 7.1.11** (Equality Criterion for Rational Classes). *For prefractions $(a, b), (c, d) \in \operatorname{Pre}\mathbb{Q}$,*

$$[a, b] = [c, d] \iff a \cdot d = b \cdot c.$$

*Remark* (Dependencies).

- Rational Class Notation
- Fraction Equivalence
- Quotient Set
- Equivalence Class

## Zero and One in $\mathbb{Q}$

**Definition**

**Definition 7.1.12** (Zero in $\mathbb{Q}$). Define

$$0_{\mathbb{Q}} := [0_{\mathbb{Z}}, 1_{\mathbb{Z}}].$$

*Remark* (Dependencies).

- Rational Class Notation
- Zero in $\mathbb{Z}$
- One in $\mathbb{Z}$

**Definition**

**Definition 7.1.13** (One in $\mathbb{Q}$). Define

$$1_{\mathbb{Q}} := [1_{\mathbb{Z}}, 1_{\mathbb{Z}}].$$

*Remark* (Dependencies).

- Rational Class Notation
- One in $\mathbb{Z}$

**Lemma**

**Lemma 7.1.14** (Vanishing Criterion in $\mathbb{Q}$). *For every prefraction* $(a, b) \in \operatorname{Pre}\mathbb{Q}$,

$$[a, b] = 0_{\mathbb{Q}} \iff a = 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Rational Class Notation
- Fraction Equivalence
- Zero in $\mathbb{Q}$
- Zero in $\mathbb{Z}$

**Theorem**

**Theorem 7.1.15** (Zero Is a Rational). *The class* $0_{\mathbb{Q}}$ *is an element of* $\mathbb{Q}$.

*Remark* (Dependencies).

- Rational Numbers
- Zero in $\mathbb{Q}$
- Rational Class Notation

**Theorem**

**Theorem 7.1.16** (One Is a Rational). *The class $1_{\mathbb{Q}}$ is an element of $\mathbb{Q}$.*

*Remark* (Dependencies).

- Rational Numbers
- One in $\mathbb{Q}$
- Rational Class Notation

**Theorem**

**Theorem 7.1.17** (Zero Is Not One in $\mathbb{Q}$). *In $\mathbb{Q}$,*

$$0_{\mathbb{Q}} \neq 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Zero Is Not One in $\mathbb{Z}$

## 7.2 Operations on $\mathbb{Q}$-Classes

### Addition on $\mathbb{Q}$-Classes

**Definition**

**Definition 7.2.1** (Addition on $\mathbb{Q}$-Classes). Define addition on representatives by

$$[a, b] + [c, d] := [a \cdot d + b \cdot c, \quad b \cdot d].$$

*Remark* (Dependencies).

- Rational Class Notation
- Addition on $\mathbb{Z}$
- Multiplication on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.2** (Addition Lands in $\operatorname{Pre}\mathbb{Q}$). *If $b \neq 0_{\mathbb{Z}}$ and $d \neq 0_{\mathbb{Z}}$, then*

$$b \cdot d \neq 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- $\mathbb{Z}$ Has No Zero Divisors

**Lemma**

**Lemma 7.2.3** (Addition Respects Fraction Equivalence on the Left). *If* $[a, b] = [a', b']$, *then*

$$[a, b] + [c, d] = [a', b'] + [c, d].$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Fraction Equivalence
- Equality Criterion for Rational Classes
- Multiplicative Cancellation on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.4** (Addition Respects Fraction Equivalence on the Right). *If* $[c, d] = [c', d']$, *then*

$$[a, b] + [c, d] = [a, b] + [c', d'].$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Fraction Equivalence
- Equality Criterion for Rational Classes
- Multiplicative Cancellation on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.5** (Addition Respects Fraction Equivalence). *If* $[a, b] = [a', b']$ *and* $[c, d] = [c', d']$, *then*

$$[a, b] + [c, d] = [a', b'] + [c', d'].$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Addition Respects Fraction Equivalence on the Left
- Addition Respects Fraction Equivalence on the Right

**Theorem**

**Theorem 7.2.6** (Addition on $\mathbb{Q}$ Is Well-Defined). *Addition determines a well-defined binary operation* $\mathbb{Q} \times \mathbb{Q} \to \mathbb{Q}$.

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Addition Respects Fraction Equivalence

**Theorem**

**Theorem 7.2.7** (Addition on $\mathbb{Q}$ Is Associative). *For all* $x, y, z \in \mathbb{Q}$,

$$(x + y) + z = x + (y + z).$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Well-Defined
- Addition on $\mathbb{Z}$ Is Associative
- Multiplication on $\mathbb{Z}$ Is Associative
- Multiplication Distributes over Addition on $\mathbb{Z}$

**Theorem**

**Theorem 7.2.8** (Addition on $\mathbb{Q}$ Is Commutative). *For all* $x, y \in \mathbb{Q}$,

$$x + y = y + x.$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Well-Defined
- Addition on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 7.2.9** (Zero Is a Left Additive Identity on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$0_{\mathbb{Q}} + x = x.$$

*Remark* (Dependencies).

- Zero in $\mathbb{Q}$
- Addition on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Well-Defined
- Zero Is an Additive Identity on $\mathbb{Z}$
- One Is a Multiplicative Identity on $\mathbb{Z}$

**Corollary**

**Corollary 7.2.10** (Zero Is a Right Additive Identity on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$x + 0_{\mathbb{Q}} = x.$$

*Remark* (Dependencies).

- Zero Is a Left Additive Identity on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.2.11** (Zero Is an Additive Identity on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$0_{\mathbb{Q}} + x = x \qquad \textit{and} \qquad x + 0_{\mathbb{Q}} = x.$$

*Remark* (Dependencies).

- Zero Is a Left Additive Identity on $\mathbb{Q}$
- Zero Is a Right Additive Identity on $\mathbb{Q}$

## Negation on $\mathbb{Q}$-Classes

**Definition**

**Definition 7.2.12** (Negation on $\mathbb{Q}$-Classes). Define negation on representatives by

$$-[a, b] := [-a, b].$$

*Remark* (Dependencies).

- Rational Class Notation
- Negation on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.13** (Negation Respects Fraction Equivalence). *If* $[a, b] = [a', b']$, *then*

$$-[a, b] = -[a', b'].$$

*Remark* (Dependencies).

- Negation on $\mathbb{Q}$
- Fraction Equivalence
- Sign Rule on $\mathbb{Z}$

**Theorem**

**Theorem 7.2.14** (Negation on $\mathbb{Q}$ Is Well-Defined). *Negation determines a well-defined unary operation* $\mathbb{Q} \to \mathbb{Q}$.

*Remark* (Dependencies).

- Negation on $\mathbb{Q}$
- Negation Respects Fraction Equivalence

**Theorem**

**Theorem 7.2.15** (Negation Gives a Left Additive Inverse on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$(-x) + x = 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Negation on $\mathbb{Q}$
- Addition on $\mathbb{Q}$
- Negation on $\mathbb{Q}$ Is Well-Defined
- Addition on $\mathbb{Q}$ Is Well-Defined
- Negation Gives a Left Additive Inverse on $\mathbb{Z}$

**Corollary**

**Corollary 7.2.16** (Negation Gives a Right Additive Inverse on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$x + (-x) = 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Negation Gives a Left Additive Inverse on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.2.17** (Every Rational Has an Additive Inverse). *For every* $x \in \mathbb{Q}$ *there exists* $y \in \mathbb{Q}$ *such that*

$$x + y = 0_{\mathbb{Q}} \qquad \textit{and} \qquad y + x = 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Negation on $\mathbb{Q}$ Is Well-Defined
- Negation Gives a Left Additive Inverse on $\mathbb{Q}$
- Negation Gives a Right Additive Inverse on $\mathbb{Q}$

### Subtraction on $\mathbb{Q}$

**Definition**

**Definition 7.2.18** (Subtraction on $\mathbb{Q}$). For $x, y \in \mathbb{Q}$, define

$$x - y := x + (-y).$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Negation on $\mathbb{Q}$

**Theorem**

**Theorem 7.2.19** (Subtraction on $\mathbb{Q}$ Is Well-Defined). *Subtraction determines a well-defined binary operation* $\mathbb{Q} \times \mathbb{Q} \to \mathbb{Q}$.

*Remark* (Dependencies).

- Subtraction on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Well-Defined
- Negation on $\mathbb{Q}$ Is Well-Defined

### Multiplication on $\mathbb{Q}$-Classes

**Definition**

**Definition 7.2.20** (Multiplication on $\mathbb{Q}$-Classes). Define multiplication on representatives by

$$[a, b] \cdot [c, d] := [a \cdot c, \quad b \cdot d].$$

*Remark* (Dependencies).

- Rational Class Notation
- Multiplication on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.21** (Multiplication Lands in $\operatorname{Pre}\mathbb{Q}$). *If* $b \neq 0_{\mathbb{Z}}$ *and* $d \neq 0_{\mathbb{Z}}$, *then*

$$b \cdot d \neq 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- $\mathbb{Z}$ Has No Zero Divisors

**Lemma**

**Lemma 7.2.22** (Multiplication Respects Fraction Equivalence on the Left). *If* $[a, b] = [a', b']$, *then*

$$[a, b] \cdot [c, d] = [a', b'] \cdot [c, d].$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Fraction Equivalence
- Equality Criterion for Rational Classes
- Multiplicative Cancellation on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.23** (Multiplication Respects Fraction Equivalence on the Right). *If* $[c, d] = [c', d']$, *then*

$$[a, b] \cdot [c, d] = [a, b] \cdot [c', d'].$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Fraction Equivalence
- Equality Criterion for Rational Classes
- Multiplicative Cancellation on $\mathbb{Z}$

**Lemma**

**Lemma 7.2.24** (Multiplication Respects Fraction Equivalence). *If* $[a, b] = [a', b']$ *and* $[c, d] = [c', d']$, *then*

$$[a, b] \cdot [c, d] = [a', b'] \cdot [c', d'].$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Multiplication Respects Fraction Equivalence on the Left
- Multiplication Respects Fraction Equivalence on the Right

**Theorem**

**Theorem 7.2.25** (Multiplication on $\mathbb{Q}$ Is Well-Defined). *Multiplication determines a well-defined binary operation* $\mathbb{Q} \times \mathbb{Q} \to \mathbb{Q}$.

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Multiplication Respects Fraction Equivalence

**Theorem**

**Theorem 7.2.26** (Multiplication on $\mathbb{Q}$ Is Associative). *For all* $x, y, z \in \mathbb{Q}$,

$$(x \cdot y) \cdot z = x \cdot (y \cdot z).$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Well-Defined
- Multiplication on $\mathbb{Z}$ Is Associative

**Theorem**

**Theorem 7.2.27** (Multiplication on $\mathbb{Q}$ Is Commutative). *For all* $x, y \in \mathbb{Q}$,

$$x \cdot y = y \cdot x.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Well-Defined
- Multiplication on $\mathbb{Z}$ Is Commutative

**Theorem**

**Theorem 7.2.28** (One Is a Left Multiplicative Identity on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$1_{\mathbb{Q}} \cdot x = x.$$

*Remark* (Dependencies).

- One in $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Well-Defined
- One Is a Multiplicative Identity on $\mathbb{Z}$

**Corollary**

**Corollary 7.2.29** (One Is a Right Multiplicative Identity on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$x \cdot 1_{\mathbb{Q}} = x.$$

*Remark* (Dependencies).

- One Is a Left Multiplicative Identity on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.2.30** (One Is a Multiplicative Identity on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$1_{\mathbb{Q}} \cdot x = x \qquad \textit{and} \qquad x \cdot 1_{\mathbb{Q}} = x.$$

*Remark* (Dependencies).

- One Is a Left Multiplicative Identity on $\mathbb{Q}$
- One Is a Right Multiplicative Identity on $\mathbb{Q}$

## Reciprocal on Nonzero $\mathbb{Q}$-Classes

**Lemma**

**Lemma 7.2.31** (Nonzero Predicate Is Well-Defined on $\mathbb{Q}$). *For every prefraction* $(a, b) \in \operatorname{Pre}\mathbb{Q}$,

$$[a, b] \neq 0_{\mathbb{Q}} \iff a \neq 0_{\mathbb{Z}}.$$

*Thus nonzeroness is independent of the representative.*

*Remark* (Dependencies).

- Zero in $\mathbb{Q}$
- Vanishing Criterion in $\mathbb{Q}$

**Definition**

**Definition 7.2.32** (Reciprocal on Nonzero $\mathbb{Q}$-Classes). For $[a, b] \neq 0_{\mathbb{Q}}$, define

$$[a, b]^{-1} := [b, a].$$

*Remark* (Dependencies).

- Rational Class Notation
- Vanishing Criterion in $\mathbb{Q}$

**Lemma**

**Lemma 7.2.33** (Reciprocal Lands in $\operatorname{Pre}\mathbb{Q}$). *If* $[a, b] \neq 0_{\mathbb{Q}}$, *then* $a \neq 0_{\mathbb{Z}}$, *so* $(b, a)$ *is a prefraction.*

*Remark* (Dependencies).

- Prefraction

- Prefraction Set
- Vanishing Criterion in $\mathbb{Q}$
- Zero in $\mathbb{Z}$

**Lemma**

**Lemma 7.2.34** (Reciprocal Respects Fraction Equivalence). *If* $[a, b] = [a', b']$ *and* $[a, b] \neq 0_{\mathbb{Q}}$*, then*

$$[a, b]^{-1} = [a', b']^{-1}.$$

*Remark* (Dependencies).

- Reciprocal on $\mathbb{Q}$
- Fraction Equivalence
- Reciprocal Lands in Pre $\mathbb{Q}$

**Theorem**

**Theorem 7.2.35** (Reciprocal on $\mathbb{Q}$ Is Well-Defined). *Reciprocal determines a well-defined map*

$$\mathbb{Q} \setminus \{0_{\mathbb{Q}}\} \to \mathbb{Q} \setminus \{0_{\mathbb{Q}}\}.$$

*Remark* (Dependencies).

- Reciprocal on $\mathbb{Q}$
- Reciprocal Lands in Pre $\mathbb{Q}$
- Reciprocal Respects Fraction Equivalence

**Theorem**

**Theorem 7.2.36** (Reciprocal Gives a Left Multiplicative Inverse on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$*,*

$$x^{-1} \cdot x = 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Reciprocal on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$
- One in $\mathbb{Q}$
- Reciprocal on $\mathbb{Q}$ Is Well-Defined
- Multiplication on $\mathbb{Q}$ Is Well-Defined

**Corollary**

**Corollary 7.2.37** (Reciprocal Gives a Right Multiplicative Inverse on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$,

$$x \cdot x^{-1} = 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Reciprocal Gives a Left Multiplicative Inverse on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.2.38** (Every Nonzero Rational Has a Multiplicative Inverse). *For every* $x \in \mathbb{Q}$ *with* $x \neq 0_{\mathbb{Q}}$, *there exists* $y \in \mathbb{Q}$ *such that*

$$x \cdot y = 1_{\mathbb{Q}} \qquad \textit{and} \qquad y \cdot x = 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Reciprocal Gives a Left Multiplicative Inverse on $\mathbb{Q}$
- Reciprocal Gives a Right Multiplicative Inverse on $\mathbb{Q}$

## Division on $\mathbb{Q}$

**Definition**

**Definition 7.2.39** (Division on $\mathbb{Q}$). For $x, y \in \mathbb{Q}$ with $y \neq 0_{\mathbb{Q}}$, define

$$x/y := x \cdot y^{-1}.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{Q}$
- Reciprocal on $\mathbb{Q}$

**Theorem**

**Theorem 7.2.40** (Division on $\mathbb{Q}$ Is Well-Defined). *Division determines a well-defined map*

$$\mathbb{Q} \times (\mathbb{Q} \setminus \{0_{\mathbb{Q}}\}) \to \mathbb{Q}.$$

*Remark* (Dependencies).

- Division on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Well-Defined
- Reciprocal on $\mathbb{Q}$ Is Well-Defined

### Distributivity on $\mathbb{Q}$-Classes

**Theorem**

**Theorem 7.2.41** (Left Distributivity on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$x \cdot (y + z) = x \cdot y + x \cdot z.$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Well-Defined
- Multiplication on $\mathbb{Q}$ Is Well-Defined
- Multiplication Distributes over Addition on $\mathbb{Z}$

**Corollary**

**Corollary 7.2.42** (Right Distributivity on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$(y + z) \cdot x = y \cdot x + z \cdot x.$$

*Remark* (Dependencies).

- Left Distributivity on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.2.43** (Multiplication Distributes over Addition on $\mathbb{Q}$). *Multiplication distributes over addition on both sides in* $\mathbb{Q}$.

*Remark* (Dependencies).

- Left Distributivity on $\mathbb{Q}$
- Right Distributivity on $\mathbb{Q}$

### Class-Level Field Summary

**Theorem**

**Theorem 7.2.44** ($\mathbb{Q}$ Is an Additive Abelian Group). *The structure* $(\mathbb{Q}, +, 0_{\mathbb{Q}})$ *is an abelian group.*

*Remark* (Dependencies).

- Group
- Addition on $\mathbb{Q}$

- Addition on $\mathbb{Q}$ Is Associative
- Addition on $\mathbb{Q}$ Is Commutative
- Zero Is an Additive Identity on $\mathbb{Q}$
- Every Rational Has an Additive Inverse

**Theorem**

**Theorem 7.2.45** (Nonzero $\mathbb{Q}$ Is a Multiplicative Abelian Group). *The structure* $(\mathbb{Q} \setminus \{0_{\mathbb{Q}}\}, \cdot, 1_{\mathbb{Q}})$ *is an abelian group.*

*Remark* (Dependencies).

- Group
- Multiplication on $\mathbb{Q}$
- Nonzero Predicate Is Well-Defined on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Associative
- Multiplication on $\mathbb{Q}$ Is Commutative
- One Is a Multiplicative Identity on $\mathbb{Q}$
- Every Nonzero Rational Has a Multiplicative Inverse

**Theorem**

**Theorem 7.2.46** ($\mathbb{Q}$ Is a Field). *The structure* $(\mathbb{Q}, +, \cdot, 0_{\mathbb{Q}}, 1_{\mathbb{Q}})$ *is a field.*

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Additive Abelian Group
- Nonzero $\mathbb{Q}$ Is a Multiplicative Abelian Group
- Multiplication Distributes over Addition on $\mathbb{Q}$
- Zero Is Not One in $\mathbb{Q}$

## 7.3 Field Structure of Q

### Derived Field Laws on $\mathbb{Q}$

**Corollary**

**Corollary 7.3.1** (Zero Product on $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$x \cdot y = 0_{\mathbb{Q}} \iff x = 0_{\mathbb{Q}} \text{ or } y = 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- Field Zero Product

**Corollary**

**Corollary 7.3.2** (Double Negation on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$-(-x) = x.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- Generic Field Laws

**Corollary**

**Corollary 7.3.3** (Negation of a Product on $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$(-x) \cdot y = -(x \cdot y) \qquad \textit{and} \qquad x \cdot (-y) = -(x \cdot y).$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- Generic Field Laws

**Corollary**

**Corollary 7.3.4** (Product of Negatives on $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$(-x) \cdot (-y) = x \cdot y.$$

*Remark* (Dependencies).

- Negation of a Product on $\mathbb{Q}$
- Double Negation on $\mathbb{Q}$

**Corollary**

**Corollary 7.3.5** (Additive Cancellation on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$x + z = y + z \Longrightarrow x = y.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Additive Abelian Group
- Generic Field Laws

Corollary

**Corollary 7.3.6** (Multiplicative Cancellation on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$z \neq 0_{\mathbb{Q}} \wedge x \cdot z = y \cdot z \Longrightarrow x = y.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- Generic Field Laws

Corollary

**Corollary 7.3.7** (Uniqueness of Additive Inverses on $\mathbb{Q}$). *Every additive inverse in* $\mathbb{Q}$ *is unique.*

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Additive Abelian Group
- Generic Field Laws

Corollary

**Corollary 7.3.8** (Uniqueness of Multiplicative Inverses on $\mathbb{Q}$). *Every multiplicative inverse of a nonzero rational number is unique.*

*Remark* (Dependencies).

- Nonzero $\mathbb{Q}$ Is a Multiplicative Abelian Group
- Generic Field Laws

Corollary

**Corollary 7.3.9** (The Inverse of One Is One on $\mathbb{Q}$). *In* $\mathbb{Q}$,

$$1_{\mathbb{Q}}^{-1} = 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- One in $\mathbb{Q}$
- Reciprocal Gives a Left Multiplicative Inverse on $\mathbb{Q}$
- One Is a Multiplicative Identity on $\mathbb{Q}$

Corollary

**Corollary 7.3.10** (Inverse of a Product on $\mathbb{Q}$). *For all nonzero* $x, y \in \mathbb{Q}$,

$$(x \cdot y)^{-1} = x^{-1} \cdot y^{-1}.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- Reciprocal on $\mathbb{Q}$ Is Well-Defined
- Uniqueness of Multiplicative Inverses on $\mathbb{Q}$

**Corollary**

**Corollary 7.3.11** (Inverse of an Inverse on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$,

$$(x^{-1})^{-1} = x.$$

*Remark* (Dependencies).

- Generic Field Laws

## Integer Powers on $\mathbb{Q}$

**Definition**

**Definition 7.3.12** (Integer Power on $\mathbb{Q}$). Let $x \in \mathbb{Q}$. For $n \in \mathbb{W}$, define $x^n$ by the usual recursive power operation. If $x \neq 0_{\mathbb{Q}}$ and $n \in \mathbb{Z}$ is negative, define

$$x^n := (x^{-1})^{-n}.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- Every Nonzero Rational Has a Multiplicative Inverse
- Field Integer Power

**Corollary**

**Corollary 7.3.13** (Exponent Addition Law on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$ *and all* $m, n \in \mathbb{Z}$,

$$x^{m+n} = x^m \cdot x^n.$$

*Remark* (Dependencies).

- Integer Power on $\mathbb{Q}$
- Field Exponent Addition

**Corollary**

**Corollary 7.3.14** (Power of a Power on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$ *and all* $m, n \in \mathbb{Z}$,

$$(x^m)^n = x^{m \cdot n}.$$

*Remark* (Dependencies).

- Integer Power on $\mathbb{Q}$

- Field Power of Power

**Corollary**

**Corollary 7.3.15** (Power of a Product on $\mathbb{Q}$). *For all nonzero $x, y \in \mathbb{Q}$ and every $n \in \mathbb{Z}$,*

$$(x \cdot y)^n = x^n \cdot y^n.$$

*Remark* (Dependencies).

- Field Exponent Addition
- Field Power of Power
- Field Power of Product

## 7.4 Order on $\mathbb{Q}$

### Positivity on $\mathbb{Q}$

**Definition**

**Definition 7.4.1** (Positive Rational). A rational number $[a, b] \in \mathbb{Q}$ is *positive* if

$$a \cdot b > 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- Rational Class Notation
- Multiplication on $\mathbb{Z}$
- Order on $\mathbb{Z}$

**Lemma**

**Lemma 7.4.2** (Positivity Respects Fraction Equivalence on $\mathbb{Q}$). *If $[a, b] = [c, d]$, then*

$$a \cdot b > 0_{\mathbb{Z}} \iff c \cdot d > 0_{\mathbb{Z}}.$$

*Remark* (Dependencies).

- $\mathbb{Z}$ Is a Totally Ordered Set
- $\mathbb{Z}$ Has No Zero Divisors

**Theorem**

**Theorem 7.4.3** (Positivity on $\mathbb{Q}$ Is Well-Defined). *The predicate "$x$ is positive" is well-defined on equivalence classes in $\mathbb{Q}$.*

*Remark* (Dependencies).

- Positive Rational

- Positivity Respects Fraction Equivalence on $\mathbb{Q}$

**Definition**

**Definition 7.4.4** (Strict Order on $\mathbb{Q}$). For $x, y \in \mathbb{Q}$, define

$$x < y \quad :\Longleftrightarrow \quad y - x \text{ is positive.}$$

*Remark* (Dependencies).

- Positive Rational
- Subtraction on $\mathbb{Q}$

**Definition**

**Definition 7.4.5** (Order on $\mathbb{Q}$). For $x, y \in \mathbb{Q}$, define

$$x \leq y \quad :\Longleftrightarrow \quad x < y \text{ or } x = y.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$

**Proposition**

**Proposition 7.4.6** (Product of Positives Is Positive on $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < x \wedge 0_{\mathbb{Q}} < y \Longrightarrow 0_{\mathbb{Q}} < x \cdot y.$$

*Remark* (Dependencies).

- Positive Rational
- Multiplication on $\mathbb{Q}$
- $\mathbb{Z}$ Has No Zero Divisors

## Trichotomy on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.7** (Sign Trichotomy on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$, *exactly one of*

$$x < 0_{\mathbb{Q}}, \qquad x = 0_{\mathbb{Q}}, \qquad 0_{\mathbb{Q}} < x$$

*holds.*

*Remark* (Dependencies).

- Positive Rational
- Strict Order on $\mathbb{Q}$
- Zero in $\mathbb{Q}$

**Theorem**

**Theorem 7.4.8** (Trichotomy on $\mathbb{Q}$). *For all $x, y \in \mathbb{Q}$, exactly one of*

$$x < y, \qquad x = y, \qquad y < x$$

*holds.*

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Subtraction on $\mathbb{Q}$
- Sign Trichotomy on $\mathbb{Q}$

## Total Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.9** (Strict Order on $\mathbb{Q}$ Is Irreflexive). *For every $x \in \mathbb{Q}$, not $x < x$.*

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Trichotomy on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.10** (Strict Order on $\mathbb{Q}$ Is Asymmetric). *For all $x, y \in \mathbb{Q}$,*

$$x < y \Longrightarrow \neg(y < x).$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Trichotomy on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.11** (Strict Order on $\mathbb{Q}$ Is Transitive). *For all $x, y, z \in \mathbb{Q}$,*

$$x < y \wedge y < z \Longrightarrow x < z.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Addition Preserves Strict Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.12** (Order on $\mathbb{Q}$ Is Reflexive). *For every* $x \in \mathbb{Q}$, $x \leq x$.

*Remark* (Dependencies).

- Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.13** (Order on $\mathbb{Q}$ Is Antisymmetric). *For all* $x, y \in \mathbb{Q}$,

$$x \leq y \wedge y \leq x \Longrightarrow x = y.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Strict Order on $\mathbb{Q}$ Is Asymmetric

**Theorem**

**Theorem 7.4.14** (Order on $\mathbb{Q}$ Is Transitive). *For all* $x, y, z \in \mathbb{Q}$,

$$x \leq y \wedge y \leq z \Longrightarrow x \leq z.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Strict Order on $\mathbb{Q}$ Is Transitive

**Theorem**

**Theorem 7.4.15** (Order on $\mathbb{Q}$ Is Total). *For all* $x, y \in \mathbb{Q}$,

$$x \leq y \text{ or } y \leq x.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Trichotomy on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.16** ($\mathbb{Q}$ Is a Totally Ordered Set). *The ordered pair* $(\mathbb{Q}, \leq)$ *is a totally ordered set.*

*Remark* (Dependencies).

- Ordered Set
- Total Order
- Order on $\mathbb{Q}$

- Order on $\mathbb{Q}$ Is Reflexive
- Order on $\mathbb{Q}$ Is Antisymmetric
- Order on $\mathbb{Q}$ Is Transitive
- Order on $\mathbb{Q}$ Is Total

### Addition Order Compatibility on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.17** (Left Addition Preserves Strict Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$x < y \implies z + x < z + y.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Addition on $\mathbb{Q}$
- Positive Rational

**Corollary**

**Corollary 7.4.18** (Right Addition Preserves Strict Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$x < y \implies x + z < y + z.$$

*Remark* (Dependencies).

- Left Addition Preserves Strict Order on $\mathbb{Q}$
- Addition on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.4.19** (Addition Preserves Strict Order on $\mathbb{Q}$). *Addition preserves strict order on both sides in* $\mathbb{Q}$.

*Remark* (Dependencies).

- Left Addition Preserves Strict Order on $\mathbb{Q}$
- Right Addition Preserves Strict Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.20** (Left Addition Preserves Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$x \leq y \implies z + x \leq z + y.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Left Addition Preserves Strict Order on $\mathbb{Q}$

**Corollary**

**Corollary 7.4.21** (Right Addition Preserves Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$x \leq y \Longrightarrow x + z \leq y + z.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Right Addition Preserves Strict Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.22** (Addition Preserves Order on $\mathbb{Q}$). *Addition preserves non-strict order on both sides in* $\mathbb{Q}$.

*Remark* (Dependencies).

- Left Addition Preserves Order on $\mathbb{Q}$
- Right Addition Preserves Order on $\mathbb{Q}$

**Corollary**

**Corollary 7.4.23** (Addition Reflects Order on $\mathbb{Q}$). *Addition by a fixed rational reflects both strict and non-strict order.*

*Remark* (Dependencies).

- Addition Preserves Strict Order on $\mathbb{Q}$
- Addition Preserves Order on $\mathbb{Q}$
- Negation Gives a Left Additive Inverse on $\mathbb{Q}$

### Multiplication Order Compatibility on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.24** (Left Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < z \wedge x < y \Longrightarrow z \cdot x < z \cdot y.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Product of Positives Is Positive on $\mathbb{Q}$

- Multiplication on $\mathbb{Q}$

**Corollary**

**Corollary 7.4.25** (Right Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < z \wedge x < y \Longrightarrow x \cdot z < y \cdot z.$$

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 7.4.26** (Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$). *Multiplication by a positive rational preserves strict order on both sides.*

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$
- Right Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.27** (Left Multiplication by Positives Preserves Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < z \wedge x \leq y \Longrightarrow z \cdot x \leq z \cdot y.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Left Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$

**Corollary**

**Corollary 7.4.28** (Right Multiplication by Positives Preserves Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < z \wedge x \leq y \Longrightarrow x \cdot z \leq y \cdot z.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Right Multiplication by Positives Preserves Strict Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.29** (Multiplication by Positives Preserves Order on $\mathbb{Q}$). *Multiplication by a positive rational preserves non-strict order on both sides.*

*Remark* (Dependencies).

- Left Multiplication by Positives Preserves Order on $\mathbb{Q}$
- Right Multiplication by Positives Preserves Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.30** (Multiplication by Negatives Reverses Strict Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$z < 0_{\mathbb{Q}} \wedge x < y \Longrightarrow z \cdot y < z \cdot x.$$

*Remark* (Dependencies).

- Strict Order on $\mathbb{Q}$
- Negation of a Product on $\mathbb{Q}$
- Positive Multiplication Preserves Strict Order on $\mathbb{Q}$

**Corollary**

**Corollary 7.4.31** (Multiplication by Negatives Reverses Order on $\mathbb{Q}$). *For all* $x, y, z \in \mathbb{Q}$,

$$z < 0_{\mathbb{Q}} \wedge x \leq y \Longrightarrow z \cdot y \leq z \cdot x.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Multiplication by Negatives Reverses Strict Order on $\mathbb{Q}$

## Reciprocal Order Laws on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.32** (Sign of a Reciprocal on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < x \Longrightarrow 0_{\mathbb{Q}} < x^{-1}, \qquad x < 0_{\mathbb{Q}} \Longrightarrow x^{-1} < 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Reciprocal on $\mathbb{Q}$
- Order on $\mathbb{Q}$

- Product of Positives Is Positive on $\mathbb{Q}$
- Reciprocal Is a Left Multiplicative Inverse on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.33** (Reciprocal Reverses Order on Positives in $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < x < y \Longrightarrow 0_{\mathbb{Q}} < y^{-1} < x^{-1}.$$

*Remark* (Dependencies).

- Reciprocal on $\mathbb{Q}$
- Order on $\mathbb{Q}$
- Sign of a Reciprocal on $\mathbb{Q}$
- Positive Multiplication Preserves Strict Order on $\mathbb{Q}$

## Absolute Value on $\mathbb{Q}$

**Definition**

**Definition 7.4.34** (Absolute Value on $\mathbb{Q}$). For $x \in \mathbb{Q}$, define

$$|x| := \begin{cases} x, & 0_{\mathbb{Q}} \leq x, \\ -x, & x < 0_{\mathbb{Q}}. \end{cases}$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Zero in $\mathbb{Q}$
- Negation on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.35** (Absolute Value Is Nonnegative on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$0_{\mathbb{Q}} \leq |x|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Q}$
- Order on $\mathbb{Q}$
- Trichotomy on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.36** (Absolute Value Vanishes Only at Zero on $\mathbb{Q}$). *For every* $x \in \mathbb{Q}$,

$$|x| = 0_{\mathbb{Q}} \iff x = 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Q}$
- Absolute Value Is Nonnegative on $\mathbb{Q}$
- Trichotomy on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.37** (Absolute Value Is Multiplicative on $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$|x \cdot y| = |x| \cdot |y|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Q}$
- Positive Multiplication Preserves Order on $\mathbb{Q}$
- Product of Negatives on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.38** (Triangle Inequality on $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$|x + y| \leq |x| + |y|.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Q}$
- Addition on $\mathbb{Q}$
- Order on $\mathbb{Q}$
- $\mathbb{Q}$ Is a Totally Ordered Set
- Addition Preserves Order on $\mathbb{Q}$

**Theorem**

**Theorem 7.4.39** (Absolute Value of a Reciprocal on $\mathbb{Q}$). *For every nonzero* $x \in \mathbb{Q}$,

$$|x^{-1}| = |x|^{-1}.$$

*Remark* (Dependencies).

- Absolute Value on $\mathbb{Q}$

- Reciprocal on $\mathbb{Q}$ Is Well-Defined
- Sign of a Reciprocal on $\mathbb{Q}$
- Absolute Value Is Multiplicative on $\mathbb{Q}$

## 7.5 Ordered Field Structure of Q

### Ordered Field Instantiation

**Theorem**

**Theorem 7.5.1** ($\mathbb{Q}$ Is an Ordered Field). *The structure*

$$(\mathbb{Q}, +, \cdot, 0_{\mathbb{Q}}, 1_{\mathbb{Q}}, \leq)$$

*is an ordered field.*

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- $\mathbb{Q}$ Is a Totally Ordered Set
- Addition Preserves Order on $\mathbb{Q}$
- Multiplication by Positives Preserves Order on $\mathbb{Q}$

**Corollary**

**Corollary 7.5.2** (Ordered Field Laws Hold on $\mathbb{Q}$). *Every theorem that depends only on the ordered-field axioms applies to $\mathbb{Q}$ with its constructed operations and order.*

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Generic Ordered-Field Laws

### Standard Ordered-Field Consequences

**Corollary**

**Corollary 7.5.3** (Squares Are Nonnegative on $\mathbb{Q}$). *For every $x \in \mathbb{Q}$,*

$$0_{\mathbb{Q}} \leq x^2.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Ordered Field Squares Are Nonnegative

**Corollary**

**Corollary 7.5.4** (One Is Positive in $\mathbb{Q}$). *In $\mathbb{Q}$,*

$$0_{\mathbb{Q}} < 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Ordered Field One Is Positive

**Corollary**

**Corollary 7.5.5** (Natural Numbers Are Positive in $\mathbb{Q}$). *Every natural number embedded in $\mathbb{Q}$ is positive.*

*Remark* (Dependencies).

- One Is Positive in $\mathbb{Q}$

**Corollary**

**Corollary 7.5.6** (Positive Rationals Have Positive Inverses). *For every $x \in \mathbb{Q}$,*

$$0_{\mathbb{Q}} < x \Longrightarrow 0_{\mathbb{Q}} < x^{-1}.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Ordered Field Positive Inverses

**Corollary**

**Corollary 7.5.7** (Fraction Comparison in $\mathbb{Q}$). *For rational classes with positive denominator product,*

$$[a, b] < [c, d] \quad \Longleftrightarrow \quad a \cdot d < c \cdot b.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Ordered Field Squares Are Nonnegative
- Ordered Field One Is Positive
- Ordered Field Positive Inverses
- Ordered Field Fraction Comparison
- Reciprocal Reverses Order on Positives in $\mathbb{Q}$

## 7.6 Arbitrary Closeness in $\mathbb{Q}$

### Density and the Archimedean Property

**Definition**

**Definition 7.6.1** (Two in $\mathbb{Q}$). Define

$$2_{\mathbb{Q}} := 1_{\mathbb{Q}} + 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- One in $\mathbb{Q}$
- Addition on $\mathbb{Q}$

**Lemma**

**Lemma 7.6.2** (Two Is Nonzero in $\mathbb{Q}$). *In* $\mathbb{Q}$,

$$2_{\mathbb{Q}} \neq 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- One Is Positive in $\mathbb{Q}$
- Trichotomy on $\mathbb{Q}$

**Lemma**

**Lemma 7.6.3** (Midpoint Lies Strictly Between Rationals). *For all* $x, y \in \mathbb{Q}$,

$$x < y \Longrightarrow x < (x + y) \cdot 2_{\mathbb{Q}}^{-1} < y.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Two Is Nonzero in $\mathbb{Q}$

**Theorem**

**Theorem 7.6.4** (Density of $\mathbb{Q}$ in Itself). *For all* $x, y \in \mathbb{Q}$,

$$x < y \Longrightarrow \exists r \in \mathbb{Q} \ (x < r < y).$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Midpoint Lies Strictly Between Rationals

**Theorem**

**Theorem 7.6.5** (Archimedean Property of $\mathbb{Q}$). *For all* $x, y \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < x \Longrightarrow \exists n \in \mathbb{N} \ (y < n \cdot x).$$

*Remark* (Dependencies).

- Well-Ordering Principle
- Division Algorithm on $\mathbb{Z}$
- $\mathbb{Z}$ Is a Totally Ordered Set

## From Discreteness to Arbitrary Closeness

*Remark* 7.6.6 (Discrete Systems Before $\mathbb{Q}$). The systems $\mathbb{N}$, $\mathbb{W}$, and $\mathbb{Z}$ are discrete in the order sense: consecutive integer points have no same-system point between them. The rational numbers are the first number system in this construction chain where intervals can be subdivided internally without leaving the system.

**Definition**

**Definition 7.6.7** (Arbitrarily Close Distinct Points). A number system has arbitrarily close distinct points if for every point $x$ and every positive tolerance $\varepsilon$, there is a point $y \neq x$ such that

$$|x - y| < \varepsilon.$$

*Remark* (Dependencies).

- Order on $\mathbb{Q}$
- Absolute Value on $\mathbb{Q}$

**Theorem**

**Theorem 7.6.8** ($\mathbb{Q}$ Has Arbitrarily Close Distinct Points). *For every* $x \in \mathbb{Q}$ *and every* $\varepsilon \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < \varepsilon \Longrightarrow \exists y \in \mathbb{Q} \ (y \neq x \ \textit{and} \ |x - y| < \varepsilon).$$

*Remark* (Dependencies).

- Reciprocals Become Arbitrarily Small in $\mathbb{Q}$
- Addition Preserves Strict Order on $\mathbb{Q}$

**Corollary**

**Corollary 7.6.9** ($\mathbb{Q}$ Has No Adjacent Points). *There do not exist rational numbers* $x < y$ *with no rational number strictly between them.*

*Remark* (Dependencies).

- Density of $\mathbb{Q}$ in Itself

**Proposition**

**Proposition 7.6.10** ($\mathbb{Q}$ Has No Least Positive Rational). *For every* $x \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < x \Longrightarrow \exists y \in \mathbb{Q} \ (0_{\mathbb{Q}} < y < x).$$

*Remark* (Dependencies).

- Density of $\mathbb{Q}$ in Itself

**Corollary**

**Corollary 7.6.11** (Reciprocals Become Arbitrarily Small in $\mathbb{Q}$). *For every* $\varepsilon \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < \varepsilon \Longrightarrow \exists n \in \mathbb{N} \ \left(0_{\mathbb{Q}} < n^{-1} < \varepsilon\right).$$

*Remark* (Dependencies).

- Archimedean Property of $\mathbb{Q}$
- Sign of a Reciprocal on $\mathbb{Q}$

## Limit-Like Behavior Before Limits

**Proposition**

**Proposition 7.6.12** (One-Sided Rational Approximation). *For every* $x \in \mathbb{Q}$ *and every* $\varepsilon \in \mathbb{Q}$,

$$0_{\mathbb{Q}} < \varepsilon \Longrightarrow \exists y, z \in \mathbb{Q} \ (x - \varepsilon < y < x < z < x + \varepsilon).$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Has Arbitrarily Close Distinct Points
- Density of $\mathbb{Q}$ in Itself

*Remark* 7.6.13 (Refinement and Bounds). The Archimedean property supplies arbitrarily small rational increments, so a rational point can be approached from one side at any prescribed tolerance. Mediants supply a complementary order-refinement mechanism: once two rational bounds are known, their mediant is a new rational bound strictly between them. The following approximation topics make these two mechanisms explicit through dyadic grids, mediant refinement, Stern-Brocot paths, and rational Cauchy sequences.

## 7.7 Rational Gaps

### No Rational Square Root of Two

**Theorem**

**Theorem 7.7.1** (No Rational Square Root of Two). *There is no rational number $q \in \mathbb{Q}$ such that*

$$q \cdot q = 2_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- $\mathbb{Q}$ Is a Field
- $\mathbb{Z}$ Is an Integral Domain
- parity or infinite descent facts in $\mathbb{Z}$

### The Square-Two Gap

**Definition**

**Definition 7.7.2** (The Gap Set in $\mathbb{Q}$). Define

$$S := \{q \in \mathbb{Q} : 0_{\mathbb{Q}} < q \text{ and } q \cdot q < 2_{\mathbb{Q}}\}.$$

*Remark* (Dependencies).

- Rationals
- Order on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$
- Zero in $\mathbb{Q}$

**Proposition**

**Proposition 7.7.3** (The Gap Set Is Bounded Above in $\mathbb{Q}$). *The set $S$ is nonempty and bounded above in $\mathbb{Q}$.*

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Natural Numbers Are Positive in $\mathbb{Q}$

**Theorem**

**Theorem 7.7.4** (The Square-Two Gap Has No Rational Supremum). *The set $S$ has no least upper bound in $\mathbb{Q}$.*

*Remark* (Dependencies).

- No Rational Square Root of Two
- Density of $\mathbb{Q}$ in Itself
- $\mathbb{Q}$ Is an Ordered Field

**Corollary**

**Corollary 7.7.5** ($\mathbb{Q}$ Is Order-Incomplete). *The ordered field $\mathbb{Q}$ is not order-complete: there exists a nonempty bounded-above subset of $\mathbb{Q}$ with no supremum in $\mathbb{Q}$.*

*Remark* (Dependencies).

- The Gap Set Is Bounded Above in $\mathbb{Q}$
- The Square-Two Gap Has No Rational Supremum

## Cauchy Sequences in $\mathbb{Q}$

**Definition**

**Definition 7.7.6** (Cauchy Sequence in $\mathbb{Q}$). A sequence $(x_n)$ in $\mathbb{Q}$ is Cauchy if

$$\forall \varepsilon \in \mathbb{Q}\ (0_{\mathbb{Q}} < \varepsilon \Longrightarrow \exists N \in \mathbb{N}\, \forall m, n \geq N\ |x_m - x_n| < \varepsilon)\,.$$

*Remark* (Dependencies).

- Rationals
- Order on $\mathbb{Q}$
- Absolute Value on $\mathbb{Q}$
- Subtraction on $\mathbb{Q}$

**Definition**

**Definition 7.7.7** (Null Sequence in $\mathbb{Q}$). A sequence $(x_n)$ in $\mathbb{Q}$ is null if

$$\forall \varepsilon \in \mathbb{Q}\ (0_{\mathbb{Q}} < \varepsilon \Longrightarrow \exists N \in \mathbb{N}\, \forall n \geq N\ |x_n| < \varepsilon)\,.$$

*Remark* (Dependencies).

- Rationals
- Order on $\mathbb{Q}$
- Absolute Value on $\mathbb{Q}$

**Lemma**

**Lemma 7.7.8** (Rational Limits Are Unique). *A convergent sequence in $\mathbb{Q}$ has at most one rational limit.*

*Remark* (Dependencies).

- Triangle Inequality on $\mathbb{Q}$
- $\mathbb{Q}$ Is an Ordered Field

## 7.8 Embedding $\mathbb{Z}$ into $\mathbb{Q}$

### The Integer Embedding

**Definition**

**Definition 7.8.1** (Embedding of $\mathbb{Z}$ into $\mathbb{Q}$). Define

$$\iota : \mathbb{Z} \to \mathbb{Q}, \qquad \iota(n) := [n, 1_{\mathbb{Z}}].$$

*Remark* (Dependencies).

- Rational Numbers
- One in $\mathbb{Z}$
- Zero Is Not One in $\mathbb{Z}$

**Theorem**

**Theorem 7.8.2** (Embedding Preserves Zero). *The embedding sends integer zero to rational zero:*

$$\iota(0_{\mathbb{Z}}) = 0_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Z}$ into $\mathbb{Q}$
- Zero in $\mathbb{Z}$
- Zero in $\mathbb{Q}$

**Theorem**

**Theorem 7.8.3** (Embedding Preserves One). *The embedding sends integer one to rational one:*

$$\iota(1_{\mathbb{Z}}) = 1_{\mathbb{Q}}.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Z}$ into $\mathbb{Q}$
- One in $\mathbb{Z}$
- One in $\mathbb{Q}$

**Theorem**

**Theorem 7.8.4** (Embedding $\mathbb{Z}$ into $\mathbb{Q}$ Is Injective). *For all* $m, n \in \mathbb{Z}$,

$$\iota(m) = \iota(n) \Longrightarrow m = n.$$

*Remark* (Dependencies).

- Equality Criterion for Rational Classes
- One Is a Multiplicative Identity on $\mathbb{Z}$

### Preservation of Algebra and Order

**Theorem**

**Theorem 7.8.5** (Embedding Preserves Addition). *For all* $m, n \in \mathbb{Z}$,

$$\iota(m+n) = \iota(m) + \iota(n).$$

*Remark* (Dependencies).

- Embedding $\mathbb{Z}$ into $\mathbb{Q}$
- Addition on $\mathbb{Z}$
- Addition on $\mathbb{Q}$
- Addition on $\mathbb{Z}$ Is Well-Defined
- Addition on $\mathbb{Q}$ Is Well-Defined

**Theorem**

**Theorem 7.8.6** (Embedding Preserves Multiplication). *For all* $m, n \in \mathbb{Z}$,

$$\iota(m \cdot n) = \iota(m) \cdot \iota(n).$$

*Remark* (Dependencies).

- Embedding $\mathbb{Z}$ into $\mathbb{Q}$
- Multiplication on $\mathbb{Z}$
- Multiplication on $\mathbb{Q}$
- Multiplication on $\mathbb{Z}$ Is Well-Defined
- Multiplication on $\mathbb{Q}$ Is Well-Defined

**Theorem**

**Theorem 7.8.7** (Embedding Preserves Order). *For all* $m, n \in \mathbb{Z}$,

$$m \leq n \iff \iota(m) \leq \iota(n).$$

*Remark* (Dependencies).

- Addition on $\mathbb{Q}$-Classes
- Multiplication on $\mathbb{Q}$-Classes
- Order on $\mathbb{Q}$

- $\mathbb{Z}$ Is a Totally Ordered Set

**Theorem**

**Theorem 7.8.8** ($\mathbb{Z}$ Embeds into $\mathbb{Q}$). *The map $\iota : \mathbb{Z} \to \mathbb{Q}$ is an injective order-preserving ring homomorphism.*

*Remark* (Dependencies).

- Embedding Preserves Zero
- Embedding Preserves One
- Embedding $\mathbb{Z}$ into $\mathbb{Q}$ Is Injective
- Embedding Preserves Addition
- Embedding Preserves Multiplication
- Embedding Preserves Order

## Field of Fractions

**Theorem**

**Theorem 7.8.9** ($\mathbb{Q}$ Is the Field of Fractions of $\mathbb{Z}$). *For every $x \in \mathbb{Q}$, there exist $m, n \in \mathbb{Z}$ with $n \neq 0_{\mathbb{Z}}$ such that*

$$x = \iota(m) \cdot \iota(n)^{-1}.$$

*Remark* (Dependencies).

- Rational Class Notation
- Every Nonzero Rational Has a Multiplicative Inverse
- Vanishing Criterion

**Corollary**

**Corollary 7.8.10** (The Embedding of $\mathbb{Z}$ into $\mathbb{Q}$ Is Not Surjective). *The image of $\iota : \mathbb{Z} \to \mathbb{Q}$ is a proper subset of $\mathbb{Q}$.*

*Remark* (Dependencies).

- Density of $\mathbb{Q}$ in Itself
- Embedding Preserves Order

## 7.9 Transition to $\mathbb{R}$

### Why the Real Numbers Are Introduced

*Remark* 7.9.1 (Why the Real Numbers Are Introduced). The rational numbers form an ordered field, are dense in themselves, and satisfy the Archimedean property. These facts make $\mathbb{Q}$ a rich arithmetic and order structure, but they do not make it complete.

The square-two gap shows the defect precisely: there is a nonempty bounded-above subset of $\mathbb{Q}$ with no supremum in $\mathbb{Q}$. The real numbers are introduced as a completion of $\mathbb{Q}$, so that bounded-above sets acquire suprema and Cauchy sequences acquire limits.

*Remark* (Dependencies).

- $\mathbb{Q}$ Is an Ordered Field
- Density of $\mathbb{Q}$ in Itself
- Archimedean Property of $\mathbb{Q}$
- $\mathbb{Q}$ Is Order-Incomplete
- Cauchy Sequence in $\mathbb{Q}$

## 7.10 Canonical Rational Number Statement Plan

### Architecture Rule for the Rational Numbers

The rational-number chapter is organized around a local-versus-generic separation. Local results open the quotient construction box: prefractions, fraction equivalence, well-defined operations, field axioms, order, density, gaps, and the embedding of $\mathbb{Z}$ into $\mathbb{Q}$. Generic field and ring consequences are not re-proved here. They are cited as corollaries from the general algebraic infrastructure developed earlier.

*Remark* (Partiality of reciprocal). The new structural issue relative to $\mathbb{Z}$ is that reciprocal is a partial operation. It is defined only on nonzero rational classes, so the nonzero predicate must be proved well-defined before reciprocal is introduced.

### Construction of $\mathbb{Q}$

- Def `def:nonzero-integers`: $\mathbb{Z}^* := \mathbb{Z} \setminus \{0\}$.
- Def `def:prefraction`: a prefraction is $(a, b) \in \mathbb{Z} \times \mathbb{Z}^*$, read as $a/b$.
- Def `def:prefraction-set`: $\mathrm{PreQ} := \mathbb{Z} \times \mathbb{Z}^*$.
- Def `def:fraction-equivalence`: $(a, b) \sim_Q (c, d)$ iff $ad = bc$.
- Lem `lem:fraction-equivalence-reflexive`: $(a, b) \sim_Q (a, b)$.
- Lem `lem:fraction-equivalence-symmetric`: $(a, b) \sim_Q (c, d) \Rightarrow (c, d) \sim_Q (a, b)$.

- Lem `lem:fraction-equivalence-transitive`: fraction equivalence is transitive; depends on nonzero multiplicative cancellation in $\mathbb{Z}$.
- Thm `thm:fraction-equivalence-relation`: $\sim_Q$ is an equivalence relation on PreQ.
- Def `def:rationals`: $\mathbb{Q} := \mathrm{PreQ} \,/\, \sim_Q$.
- Def `def:rational-class-notation`: $[a, b]$ denotes the $\sim_Q$-class of $(a, b)$ with $b \neq 0$.
- Def `def:zero-in-q`: $0_Q := [0, 1]$.
- Def `def:one-in-q`: $1_Q := [1, 1]$.
- Lem `lem:vanishing-criterion-q`: $[a, b] = 0_Q$ iff $a = 0$.
- Thm `thm:zero-is-a-rational`: $0_Q \in \mathbb{Q}$.
- Thm `thm:one-is-a-rational`: $1_Q \in \mathbb{Q}$.
- Thm `thm:zero-not-one-in-q`: $0_Q \neq 1_Q$; depends on $0_Z \neq 1_Z$.

**Operations on Rational Classes**

- Def `def:addition-on-q-classes`: $[a, b] + [c, d] := [ad + bc, bd]$.
- Lem `lem:addition-denominator-nonzero-q`: $b \neq 0$ and $d \neq 0$ imply $bd \neq 0$.
- Lem `lem:addition-on-q-respects-left-equivalence`: addition respects fraction equivalence in the left argument.
- Lem `lem:addition-on-q-respects-right-equivalence`: addition respects fraction equivalence in the right argument.
- Lem `lem:addition-on-q-respects-equivalence`: addition respects fraction equivalence in both arguments.
- Thm `thm:addition-on-q-well-defined`: addition is a well-defined operation $\mathbb{Q} \times \mathbb{Q} \to \mathbb{Q}$.
- Thm `thm:addition-on-q-associative`: $(x + y) + z = x + (y + z)$.
- Thm `thm:addition-on-q-commutative`: $x + y = y + x$.
- Thm `thm:zero-left-additive-identity-on-q`: $0_Q + x = x$.
- Cor `cor:zero-right-additive-identity-on-q`: $x + 0_Q = x$.
- Def `def:negation-on-q-classes`: $-[a, b] := [-a, b]$.
- Lem `lem:negation-on-q-respects-equivalence`: negation respects fraction equivalence.

- Thm `thm:negation-on-q-well-defined`: negation is well-defined on $\mathbb{Q}$.
- Thm `thm:negation-left-additive-inverse-on-q`: $(-x) + x = 0_Q$.
- Cor `cor:negation-right-additive-inverse-on-q`: $x + (-x) = 0_Q$.
- Thm `thm:every-rational-has-additive-inverse`: every rational has a two-sided additive inverse.
- Def `def:subtraction-on-q`: $x - y := x + (-y)$.
- Thm `thm:subtraction-on-q-well-defined`: subtraction is well-defined.
- Def `def:multiplication-on-q-classes`: $[a, b] \cdot [c, d] := [ac, bd]$.
- Lem `lem:multiplication-denominator-nonzero-q`: $b \neq 0$ and $d \neq 0$ imply $bd \neq 0$.
- Lem `lem:multiplication-on-q-respects-left-equivalence`: multiplication respects equivalence in the left argument.
- Lem `lem:multiplication-on-q-respects-right-equivalence`: multiplication respects equivalence in the right argument.
- Lem `lem:multiplication-on-q-respects-equivalence`: multiplication respects equivalence in both arguments.
- Thm `thm:multiplication-on-q-well-defined`: multiplication is well-defined on $\mathbb{Q}$.
- Thm `thm:multiplication-on-q-associative`: $(xy)z = x(yz)$.
- Thm `thm:multiplication-on-q-commutative`: $xy = yx$.
- Thm `thm:one-left-multiplicative-identity-on-q`: $1_Q x = x$.
- Cor `cor:one-right-multiplicative-identity-on-q`: $x 1_Q = x$.
- Lem `lem:nonzero-predicate-well-defined-q`: $[a, b] \neq 0_Q$ iff $a \neq 0$.
- Def `def:reciprocal-on-q-classes`: for $a \neq 0$, $[a, b]^{-1} := [b, a]$.
- Lem `lem:reciprocal-denominator-nonzero-q`: $a \neq 0$ puts $(b, a)$ in PreQ.
- Lem `lem:reciprocal-on-q-respects-equivalence`: reciprocal respects equivalence on nonzero classes.
- Thm `thm:reciprocal-on-q-well-defined`: reciprocal is a well-defined map $\mathbb{Q} \setminus \{0_Q\} \to \mathbb{Q} \setminus \{0_Q\}$.
- Thm `thm:reciprocal-left-multiplicative-inverse-on-q`: $x \neq 0_Q \Rightarrow x^{-1}x = 1_Q$.

- Cor `cor:reciprocal-right-multiplicative-inverse-on-q`: $x \neq 0_Q \Rightarrow xx^{-1} = 1_Q$.
- Thm `thm:every-nonzero-rational-has-multiplicative-inverse`: every nonzero rational has a two-sided multiplicative inverse.
- Def `def:division-on-q`: for $y \neq 0_Q$, $x/y := x \cdot y^{-1}$.
- Thm `thm:division-on-q-well-defined`: division is well-defined on $\mathbb{Q} \times (\mathbb{Q} \setminus \{0_Q\})$.
- Thm `thm:left-distributivity-on-q`: $x(y+z) = xy + xz$.
- Cor `cor:right-distributivity-on-q`: $(y+z)x = yx + zx$.

**Field Structure and Generic Field Laws**

- Thm `thm:q-additive-abelian-group`: $(\mathbb{Q}, +, 0_Q)$ is an abelian group.
- Thm `thm:q-nonzero-multiplicative-abelian-group`: $(\mathbb{Q} \setminus \{0_Q\}, \cdot, 1_Q)$ is an abelian group.
- Thm `thm:q-is-a-field`: $\mathbb{Q}$ is a field.
- Cor `cor:zero-product-on-q`: $xy = 0_Q$ iff $x = 0_Q$ or $y = 0_Q$; cites the generic field zero-product theorem.
- Cor `cor:double-negation-on-q`: $-(-x) = x$; cites the generic ring result.
- Cor `cor:negation-of-product-on-q`: $(-x)y = -(xy) = x(-y)$; cites the generic ring result.
- Cor `cor:product-of-negatives-on-q`: $(-x)(-y) = xy$; cites the generic ring result.
- Cor `cor:additive-cancellation-on-q`: $x + z = y + z \Rightarrow x = y$; cites generic group cancellation.
- Cor `cor:multiplicative-cancellation-on-q`: $z \neq 0_Q$ and $xz = yz$ imply $x = y$; cites generic field cancellation.
- Cor `cor:uniqueness-additive-inverse-on-q`: additive inverses are unique.
- Cor `cor:uniqueness-multiplicative-inverse-on-q`: multiplicative inverses of nonzero elements are unique.
- Cor `cor:inverse-of-one-is-one-on-q`: $1_Q^{-1} = 1_Q$.
- Cor `cor:inverse-of-product-on-q`: if $x, y \neq 0$, then $(xy)^{-1} = x^{-1}y^{-1}$.
- Cor `cor:inverse-of-inverse-on-q`: if $x \neq 0$, then $(x^{-1})^{-1} = x$.
- Def `def:integer-power-on-q`: $x^n$ for $n \in \mathbb{Z}$, with $x \neq 0_Q$ when $n < 0$.

- Cor `cor:exponent-addition-on-q`: $x^{m+n} = x^m x^n$.
- Cor `cor:power-of-power-on-q`: $(x^m)^n = x^{mn}$.
- Cor `cor:power-of-product-on-q`: $(xy)^n = x^n y^n$.

**Order on $\mathbb{Q}$**

- Def `def:positive-rational`: $[a, b]$ is positive iff $ab > 0$ in $\mathbb{Z}$.
- Lem `lem:positivity-respects-equivalence-q`: positivity respects fraction equivalence.
- Thm `thm:positivity-on-q-well-defined`: positivity is well-defined on $\mathbb{Q}$.
- Def `def:strict-order-on-q`: $x < y$ iff $y - x$ is positive.
- Def `def:order-on-q`: $x \leq y$ iff $x < y$ or $x = y$.
- Prop `prop:product-of-positives-is-positive-q`: positive rationals have positive product.
- Thm `thm:sign-trichotomy-on-q`: exactly one of $x < 0_Q$, $x = 0_Q$, $0_Q < x$ holds.
- Thm `thm:trichotomy-on-q`: exactly one of $x < y$, $x = y$, $y < x$ holds.
- Thm `thm:strict-order-on-q-irreflexive`: not $x < x$.
- Thm `thm:strict-order-on-q-asymmetric`: $x < y \Rightarrow \neg(y < x)$.
- Thm `thm:strict-order-on-q-transitive`: $x < y$ and $y < z$ imply $x < z$.
- Thm `thm:order-on-q-reflexive`: $x \leq x$.
- Thm `thm:order-on-q-antisymmetric`: $x \leq y$ and $y \leq x$ imply $x = y$.
- Thm `thm:order-on-q-transitive`: $x \leq y$ and $y \leq z$ imply $x \leq z$.
- Thm `thm:order-on-q-total`: $x \leq y$ or $y \leq x$.
- Thm `thm:q-totally-ordered-set`: $(\mathbb{Q}, \leq)$ is a total order.
- Thm `thm:left-addition-preserves-strict-order-on-q`: $x < y \Rightarrow z + x < z + y$.
- Cor `cor:right-addition-preserves-strict-order-on-q`: $x < y \Rightarrow x + z < y + z$.
- Thm `thm:left-addition-preserves-order-on-q`: $x \leq y \Rightarrow z + x \leq z + y$.
- Cor `cor:right-addition-preserves-order-on-q`: $x \leq y \Rightarrow x + z \leq y + z$.
- Cor `cor:addition-reflects-order-on-q`: addition by a fixed rational reflects strict and non-strict order.

- Thm `thm:left-positive-multiplication-preserves-strict-order-on-q`: positive left multiplication preserves strict order.
- Cor `cor:right-positive-multiplication-preserves-strict-order-on-q`: positive right multiplication preserves strict order.
- Thm `thm:positive-multiplication-preserves-order-on-q`: positive multiplication preserves non-strict order.
- Thm `thm:negative-multiplication-reverses-strict-order-on-q`: negative multiplication reverses strict order.
- Thm `thm:reciprocal-reverses-order-on-positives-q`: $0_Q < x < y \Rightarrow 0_Q < y^{-1} < x^{-1}$.
- Thm `thm:sign-of-reciprocal-q`: reciprocals preserve positive and negative sign.
- Def `def:absolute-value-on-q`: $|x| := x$ if $0_Q \leq x$, and $|x| := -x$ otherwise.
- Thm `thm:absolute-value-nonnegative-on-q`: $0_Q \leq |x|$.
- Thm `thm:absolute-value-zero-iff-on-q`: $|x| = 0_Q$ iff $x = 0_Q$.
- Thm `thm:absolute-value-multiplicative-on-q`: $|xy| = |x||y|$.
- Thm `thm:triangle-inequality-on-q`: $|x + y| \leq |x| + |y|$.
- Thm `thm:absolute-value-of-reciprocal-q`: $x \neq 0_Q \Rightarrow |x^{-1}| = |x|^{-1}$.
- Thm `thm:q-is-an-ordered-field`: $\mathbb{Q}$ is an ordered field.

### Density, Gaps, and Embedding

- Lem `lem:midpoint-between-q`: $x < y \Rightarrow x < (x + y)2_Q^{-1} < y$.
- Thm `thm:density-of-q`: between any two rationals lies a rational.
- Prop `prop:no-least-positive-rational`: every positive rational has a smaller positive rational.
- Thm `thm:archimedean-property-of-q`: for $0_Q < x$, some $n \in \mathbb{N}$ has $y < nx$.
- Cor `cor:reciprocals-arbitrarily-small-q`: reciprocals of naturals become arbitrarily small.
- Thm `thm:no-rational-square-root-of-two`: no rational square is $2_Q$.
- Def `def:gap-set-q`: $S := \{q \in \mathbb{Q} : 0_Q < q \wedge q^2 < 2_Q\}$.
- Prop `prop:gap-set-bounded-above-q`: the gap set is nonempty and bounded above.
- Thm `thm:gap-set-no-rational-supremum-q`: the gap set has no rational supremum.

- Cor `cor:q-order-incomplete`: $\mathbb{Q}$ is not order-complete.
- Def `def:cauchy-sequence-in-q`: rational Cauchy sequence.
- Def `def:null-sequence-in-q`: rational null sequence.
- Lem `lem:rational-limit-unique`: a convergent rational sequence has at most one rational limit.
- Def `def:embedding-z-into-q`: $\iota : \mathbb{Z} \to \mathbb{Q}$, $\iota(n) := [n, 1]$.
- Thm `thm:embedding-z-into-q-preserves-zero`: $\iota(0_Z) = 0_Q$.
- Thm `thm:embedding-z-into-q-preserves-one`: $\iota(1_Z) = 1_Q$.
- Thm `thm:embedding-z-into-q-injective`: $\iota(m) = \iota(n) \Rightarrow m = n$.
- Thm `thm:embedding-z-into-q-preserves-addition`: $\iota(m + n) = \iota(m) + \iota(n)$.
- Thm `thm:embedding-z-into-q-preserves-multiplication`: $\iota(mn) = \iota(m)\iota(n)$.
- Thm `thm:embedding-z-into-q-preserves-order`: $m \leq n$ iff $\iota(m) \leq \iota(n)$.
- Thm `thm:z-embeds-into-q`: $\iota$ is an injective order-preserving ring homomorphism.
- Thm `thm:q-is-field-of-fractions-of-z`: every rational is a quotient of integer images.
- Cor `cor:embedding-z-into-q-not-surjective`: $\iota(\mathbb{Z}) \subsetneq \mathbb{Q}$.
- Rem `rem:why-real-numbers-are-introduced`: $\mathbb{Q}$ is an ordered field but order-incomplete, motivating $\mathbb{R}$.

# Chapter 8

# Real Numbers ($\mathbb{R}$)

**reals**

Rational Numbers $\rightarrow$ **Real Numbers** $\rightarrow$ Structure of the Real Line

## 8.1 Construction of $\mathbb{R}$

### Dedekind Cuts

**Definition**

**Definition 8.1.1** (Dedekind Cut). A ***Dedekind cut*** is a set $\alpha \subseteq \mathbb{Q}$ satisfying:

$$\alpha \neq \emptyset, \qquad \alpha \neq \mathbb{Q},$$

$$p \in \alpha \wedge q < p \Longrightarrow q \in \alpha,$$

and

$$p \in \alpha \Longrightarrow \exists r \in \alpha\,(p < r).$$

*Remark* (Dependencies).

- Density of $\mathbb{Q}$

**Definition**

**Definition 8.1.2** (The Real Numbers). The real numbers are the set

$$\mathbb{R} := \{\alpha : \alpha \text{ is a Dedekind cut in } \mathbb{Q}\}.$$

*Remark* (Dependencies).

- Dedekind Cut

### Rational Cuts

**Definition**

**Definition 8.1.3** (Rational Cut). For $q \in \mathbb{Q}$, define the rational cut

$$q^* := \{r \in \mathbb{Q} : r < q\}.$$

*Remark* (Dependencies).

- Rational Numbers
- Order on $\mathbb{Q}$

**Lemma**

**Lemma 8.1.4** (Rational Cuts Are Cuts). *For every $q \in \mathbb{Q}$, the set $q^*$ is a Dedekind cut.*

*Remark* (Dependencies).

- Density of $\mathbb{Q}$
- No Least Positive Rational

## Zero and One in $\mathbb{R}$

**Definition**

**Definition 8.1.5** (Zero in $\mathbb{R}$). Define

$$0_{\mathbb{R}} := (0_{\mathbb{Q}})^* = \{r \in \mathbb{Q} : r < 0_{\mathbb{Q}}\}.$$

*Remark* (Dependencies).

- Rational Cut
- Zero in $\mathbb{Q}$

**Definition**

**Definition 8.1.6** (One in $\mathbb{R}$). Define

$$1_{\mathbb{R}} := (1_{\mathbb{Q}})^* = \{r \in \mathbb{Q} : r < 1_{\mathbb{Q}}\}.$$

*Remark* (Dependencies).

- Rational Cut
- One in $\mathbb{Q}$

**Theorem**

**Theorem 8.1.7** (Zero and One Are Reals). *The cuts $0_{\mathbb{R}}$ and $1_{\mathbb{R}}$ are elements of $\mathbb{R}$.*

*Remark* (Dependencies).

- The Real Numbers
- Zero in $\mathbb{R}$
- One in $\mathbb{R}$

- Rational Cuts Are Cuts

**Theorem**

**Theorem 8.1.8** (Zero Is Not One in $\mathbb{R}$). *In* $\mathbb{R}$,

$$0_{\mathbb{R}} \neq 1_{\mathbb{R}}.$$

*Remark* (Dependencies).

- Zero in $\mathbb{R}$
- One in $\mathbb{R}$
- One Is Positive in $\mathbb{Q}$

## 8.2 Order on $\mathbb{R}$

### Order by Inclusion

**Definition**

**Definition 8.2.1** (Order on $\mathbb{R}$). For cuts $\alpha, \beta \in \mathbb{R}$, define

$$\alpha \leq \beta \quad :\Longleftrightarrow \quad \alpha \subseteq \beta.$$

*Remark* (Dependencies).

- The Real Numbers
- Subset

**Definition**

**Definition 8.2.2** (Strict Order on $\mathbb{R}$). For cuts $\alpha, \beta \in \mathbb{R}$, define

$$\alpha < \beta \quad :\Longleftrightarrow \quad \alpha \subsetneq \beta.$$

*Remark* (Dependencies).

- The Real Numbers
- Order on $\mathbb{R}$
- Proper Subset

**Theorem**

**Theorem 8.2.3** (Order on $\mathbb{R}$ Is Reflexive). *For every* $\alpha \in \mathbb{R}$, $\alpha \leq \alpha$.

*Remark* (Dependencies).

- Order on $\mathbb{R}$
- Subset

**Theorem**

**Theorem 8.2.4** (Order on $\mathbb{R}$ Is Antisymmetric). *For all* $\alpha, \beta \in \mathbb{R}$,

$$\alpha \leq \beta \wedge \beta \leq \alpha \Longrightarrow \alpha = \beta.$$

*Remark* (Dependencies).

- Order on $\mathbb{R}$
- Set Equality
- Subset

**Theorem**

**Theorem 8.2.5** (Order on $\mathbb{R}$ Is Transitive). *For all* $\alpha, \beta, \gamma \in \mathbb{R}$,

$$\alpha \leq \beta \wedge \beta \leq \gamma \Longrightarrow \alpha \leq \gamma.$$

*Remark* (Dependencies).

- Order on $\mathbb{R}$
- Subset

**Theorem**

**Theorem 8.2.6** (Comparability of Cuts). *For all cuts* $\alpha, \beta \in \mathbb{R}$,

$$\alpha \subseteq \beta \vee \beta \subseteq \alpha.$$

*Remark* (Dependencies).

- Dedekind Cut
- $\mathbb{Q}$ Is a Totally Ordered Set

**Theorem**

**Theorem 8.2.7** (Order on $\mathbb{R}$ Is Total). *For all* $\alpha, \beta \in \mathbb{R}$,

$$\alpha \leq \beta \vee \beta \leq \alpha.$$

*Remark* (Dependencies).

- Order on $\mathbb{R}$
- Comparability of Cuts

**Theorem**

**Theorem 8.2.8** (Trichotomy on $\mathbb{R}$). *For all* $\alpha, \beta \in \mathbb{R}$, *exactly one of*

$$\alpha < \beta, \qquad \alpha = \beta, \qquad \beta < \alpha$$

*holds.*

*Remark* (Dependencies).

- Strict Order on $\mathbb{R}$
- Order on $\mathbb{R}$ Is Antisymmetric
- Order on $\mathbb{R}$ Is Total

**Theorem**

**Theorem 8.2.9** ($\mathbb{R}$ Is a Totally Ordered Set). *The ordered pair* $(\mathbb{R}, \leq)$ *is a total order.*

*Remark* (Dependencies).

- Ordered Set
- Total Order
- Order on $\mathbb{R}$
- Order on $\mathbb{R}$ Is Reflexive
- Order on $\mathbb{R}$ Is Antisymmetric
- Order on $\mathbb{R}$ Is Transitive
- Order on $\mathbb{R}$ Is Total

## 8.3 Completeness of $\mathbb{R}$

**Definition**

**Definition 8.3.1** (Upper Bound in $\mathbb{R}$). Let $S \subseteq \mathbb{R}$. A cut $\gamma \in \mathbb{R}$ is an upper bound of $S$ if

$$\forall \alpha \in S\,(\alpha \leq \gamma).$$

The set $S$ is bounded above if it has an upper bound in $\mathbb{R}$.

*Remark* (Dependencies).

- The Real Numbers
- Order on $\mathbb{R}$
- Subset

**Definition**

**Definition 8.3.2** (Supremum in $\mathbb{R}$). For $S \subseteq \mathbb{R}$, a cut $\sigma$ is the supremum of $S$ if $\sigma$ is an upper bound of $S$ and every upper bound $\gamma$ of $S$ satisfies $\sigma \leq \gamma$.

*Remark* (Dependencies).

- Upper Bound in $\mathbb{R}$

- Order on $\mathbb{R}$

**Definition**

**Definition 8.3.3** (Least-Upper-Bound Property on $\mathbb{R}$). The ordered set $\mathbb{R}$ has the *least-upper-bound property* if every nonempty subset of $\mathbb{R}$ that is bounded above has a supremum in $\mathbb{R}$.

*Remark* (Dependencies).

- The Real Numbers
- Order on $\mathbb{R}$
- Upper Bound in $\mathbb{R}$
- Supremum in $\mathbb{R}$

**Lemma**

**Lemma 8.3.4** (Union of a Bounded Nonempty Family of Cuts Is a Cut). *If $S \subseteq \mathbb{R}$ is nonempty and bounded above, then*

$$\bigcup_{\alpha \in S} \alpha$$

*is a Dedekind cut.*

*Remark* (Dependencies).

- Dedekind Cut
- Upper Bound in $\mathbb{R}$
- Union

**Theorem**

**Theorem 8.3.5** (Least-Upper-Bound Property). *Every nonempty subset $S \subseteq \mathbb{R}$ that is bounded above has a least upper bound, namely*

$$\sup S = \bigcup_{\alpha \in S} \alpha.$$

*Remark* (Dependencies).

- Least-Upper-Bound Property on $\mathbb{R}$
- Union of a Bounded Nonempty Family of Cuts Is a Cut
- Supremum in $\mathbb{R}$

**Corollary**

**Corollary 8.3.6** (Greatest-Lower-Bound Property). *Every nonempty subset $S \subseteq \mathbb{R}$ that is bounded below has a greatest lower bound.*

*Remark* (Dependencies).

- Least-Upper-Bound Property

## 8.4 Addition on $\mathbb{R}$

**Definition**

**Definition 8.4.1** (Addition of Cuts). For cuts $\alpha, \beta \in \mathbb{R}$, define

$$\alpha + \beta := \{p + q : p \in \alpha, \ \ q \in \beta\}.$$

*Remark* (Dependencies).

- The Real Numbers
- Addition on $\mathbb{Q}$

**Lemma**

**Lemma 8.4.2** (Sum of Cuts Is a Cut). *If $\alpha, \beta \in \mathbb{R}$, then $\alpha + \beta$ is a Dedekind cut.*

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Dedekind Cut
- $\mathbb{Q}$ Is an Ordered Field

**Theorem**

**Theorem 8.4.3** (Addition on $\mathbb{R}$ Is Associative). *For all $\alpha, \beta, \gamma \in \mathbb{R}$,*

$$(\alpha + \beta) + \gamma = \alpha + (\beta + \gamma).$$

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Sum of Cuts Is a Cut
- Addition on $\mathbb{Q}$ Is Associative

**Theorem**

**Theorem 8.4.4** (Addition on $\mathbb{R}$ Is Commutative). *For all $\alpha, \beta \in \mathbb{R}$,*

$$\alpha + \beta = \beta + \alpha.$$

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Sum of Cuts Is a Cut
- Addition on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 8.4.5** (Zero Is an Additive Identity on $\mathbb{R}$). *For every* $\alpha \in \mathbb{R}$,

$$\alpha + 0_{\mathbb{R}} = \alpha.$$

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Zero in $\mathbb{R}$
- Zero Is an Additive Identity on $\mathbb{Q}$

**Definition**

**Definition 8.4.6** (Negation of a Cut). For $\alpha \in \mathbb{R}$, define

$$-\alpha := \{p \in \mathbb{Q} : \exists r \in \mathbb{Q}\,(r > 0 \wedge -p - r \notin \alpha)\}.$$

*Remark* (Dependencies).

- The Real Numbers
- Rational Numbers
- Positive Rational
- Order on $\mathbb{Q}$

**Lemma**

**Lemma 8.4.7** (Negation of a Cut Is a Cut). *If* $\alpha \in \mathbb{R}$, *then* $-\alpha$ *is a Dedekind cut.*

*Remark* (Dependencies).

- Negation on $\mathbb{R}$
- Dedekind Cut
- $\mathbb{Q}$ Is an Ordered Field

**Theorem**

**Theorem 8.4.8** (Negation Is an Additive Inverse on $\mathbb{R}$). *For every* $\alpha \in \mathbb{R}$,

$$\alpha + (-\alpha) = 0_{\mathbb{R}}.$$

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Negation on $\mathbb{R}$
- Negation of a Cut Is a Cut

- Zero in $\mathbb{R}$

**Theorem**

**Theorem 8.4.9** ($\mathbb{R}$ Is an Additive Abelian Group). *The structure* $(\mathbb{R}, +, 0_{\mathbb{R}})$ *is an abelian group.*

*Remark* (Dependencies).

- Group
- Addition on $\mathbb{R}$
- Addition on $\mathbb{R}$ Is Associative
- Addition on $\mathbb{R}$ Is Commutative
- Zero Is an Additive Identity on $\mathbb{R}$
- Negation Is an Additive Inverse on $\mathbb{R}$

**Definition**

**Definition 8.4.10** (Subtraction on $\mathbb{R}$). For cuts $\alpha, \beta \in \mathbb{R}$, define

$$\alpha - \beta := \alpha + (-\beta).$$

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Negation on $\mathbb{R}$

## 8.5 Multiplication on $\mathbb{R}$

**Definition**

**Definition 8.5.1** (Nonnegative Cut). A cut $\alpha \in \mathbb{R}$ is nonnegative if $0_{\mathbb{R}} \leq \alpha$ and positive if $0_{\mathbb{R}} < \alpha$.

*Remark* (Dependencies).

- The Real Numbers
- Zero in $\mathbb{R}$
- Order on $\mathbb{R}$

**Definition**

**Definition 8.5.2** (Product of Nonnegative Cuts). For nonnegative cuts $\alpha, \beta \in \mathbb{R}$, define

$$\alpha \cdot \beta := \{r \in \mathbb{Q} : r < 0_{\mathbb{Q}}\} \cup \{p \cdot q : p \in \alpha,\ \ q \in \beta,\ \ p \geq 0_{\mathbb{Q}},\ \ q \geq 0_{\mathbb{Q}}\}.$$

*Remark* (Dependencies).

- Nonnegative Cut
- Multiplication on $\mathbb{Q}$
- Order on $\mathbb{Q}$

**Lemma**

**Lemma 8.5.3** (Product of Nonnegative Cuts Is a Cut). *If $\alpha, \beta \geq 0_{\mathbb{R}}$, then $\alpha \cdot \beta$ is a Dedekind cut.*

*Remark* (Dependencies).

- Product of Nonnegative Cuts
- Dedekind Cut
- $\mathbb{Q}$ Is an Ordered Field

**Definition**

**Definition 8.5.4** (Absolute Value on $\mathbb{R}$). For $\alpha \in \mathbb{R}$, define

$$|\alpha| := \begin{cases} \alpha, & 0_{\mathbb{R}} \leq \alpha, \\ -\alpha, & \alpha < 0_{\mathbb{R}}. \end{cases}$$

*Remark* (Dependencies).

- Order on $\mathbb{R}$
- Strict Order on $\mathbb{R}$
- Zero in $\mathbb{R}$
- Negation on $\mathbb{R}$

**Theorem**

**Theorem 8.5.5** (Triangle Inequality). *For all $\alpha, \beta \in \mathbb{R}$,*

$$|\alpha + \beta| \leq |\alpha| + |\beta|.$$

*Go to proof.*

*Remark* (Dependencies).

- Absolute Value on $\mathbb{R}$
- $\mathbb{R}$ Is an Ordered Field

**Definition**

**Definition 8.5.6** (Multiplication of Cuts). For cuts $\alpha, \beta \in \mathbb{R}$, define $\alpha \cdot \beta$ by sign cases: use $|\alpha| \cdot |\beta|$ when $\alpha$ and $\beta$ have the same sign, use $-(|\alpha| \cdot |\beta|)$ when they have opposite signs, and use $0_{\mathbb{R}}$ when either factor is $0_{\mathbb{R}}$.

*Remark* (Dependencies).

- The Real Numbers
- Product of Nonnegative Cuts
- Absolute Value on $\mathbb{R}$
- Negation on $\mathbb{R}$
- Zero in $\mathbb{R}$

**Lemma**

**Lemma 8.5.7** (Product of Cuts Is a Cut). *If* $\alpha, \beta \in \mathbb{R}$, *then* $\alpha \cdot \beta \in \mathbb{R}$.

*Remark* (Dependencies).

- Multiplication on $\mathbb{R}$
- Product of Nonnegative Cuts
- Product of Nonnegative Cuts Is a Cut
- Negation of a Cut Is a Cut

**Theorem**

**Theorem 8.5.8** (Multiplication on $\mathbb{R}$ Is Associative). *For all* $\alpha, \beta, \gamma \in \mathbb{R}$,

$$(\alpha \cdot \beta) \cdot \gamma = \alpha \cdot (\beta \cdot \gamma).$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{R}$
- Product of Cuts Is a Cut
- Multiplication on $\mathbb{Q}$ Is Associative

**Theorem**

**Theorem 8.5.9** (Multiplication on $\mathbb{R}$ Is Commutative). *For all* $\alpha, \beta \in \mathbb{R}$,

$$\alpha \cdot \beta = \beta \cdot \alpha.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{R}$
- Product of Cuts Is a Cut
- Multiplication on $\mathbb{Q}$ Is Commutative

**Theorem**

**Theorem 8.5.10** (One Is a Multiplicative Identity on $\mathbb{R}$). *For every* $\alpha \in \mathbb{R}$,

$$\alpha \cdot 1_{\mathbb{R}} = \alpha.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{R}$
- One in $\mathbb{R}$
- One Is a Multiplicative Identity on $\mathbb{Q}$

**Definition**

**Definition 8.5.11** (Reciprocal of a Positive Cut). For $\alpha > 0_{\mathbb{R}}$, define

$$\alpha^{-1} := \{r \in \mathbb{Q} : r \leq 0_{\mathbb{Q}}\} \cup \{r \in \mathbb{Q} : r > 0_{\mathbb{Q}} \wedge \exists s \in \mathbb{Q}\,(s > r \wedge s^{-1} \notin \alpha)\}.$$

*Remark* (Dependencies).

- The Real Numbers
- Order on $\mathbb{R}$
- Order on $\mathbb{Q}$
- Reciprocal on $\mathbb{Q}$

**Definition**

**Definition 8.5.12** (Reciprocal of a Nonzero Cut). For $\alpha \neq 0_{\mathbb{R}}$, define $\alpha^{-1}$ to be $|\alpha|^{-1}$ if $\alpha > 0_{\mathbb{R}}$ and $-(|\alpha|^{-1})$ if $\alpha < 0_{\mathbb{R}}$.

*Remark* (Dependencies).

- Reciprocal of a Positive Cut
- Absolute Value on $\mathbb{R}$
- Negation on $\mathbb{R}$
- Zero in $\mathbb{R}$
- Order on $\mathbb{R}$

**Lemma**

**Lemma 8.5.13** (Reciprocal of a Nonzero Cut Is a Cut). *If* $\alpha \neq 0_{\mathbb{R}}$, *then* $\alpha^{-1} \in \mathbb{R}$.

*Remark* (Dependencies).

- Reciprocal on $\mathbb{R}$

- Reciprocal of a Positive Cut
- Dedekind Cut
- $\mathbb{Q}$ Is an Ordered Field

**Theorem**

**Theorem 8.5.14** (Reciprocal Is a Multiplicative Inverse on $\mathbb{R}$). *If* $\alpha \neq 0_{\mathbb{R}}$, *then*

$$\alpha \cdot \alpha^{-1} = 1_{\mathbb{R}}.$$

*Remark* (Dependencies).

- Reciprocal on $\mathbb{R}$
- Multiplication on $\mathbb{R}$
- Reciprocal of a Nonzero Cut Is a Cut
- One in $\mathbb{R}$

**Definition**

**Definition 8.5.15** (Division on $\mathbb{R}$). For $\beta \neq 0_{\mathbb{R}}$, define

$$\alpha/\beta := \alpha \cdot \beta^{-1}.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{R}$
- Reciprocal on $\mathbb{R}$
- Zero in $\mathbb{R}$

**Theorem**

**Theorem 8.5.16** (Distributivity on $\mathbb{R}$). *For all* $\alpha, \beta, \gamma \in \mathbb{R}$,

$$\alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma.$$

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Multiplication on $\mathbb{R}$
- Multiplication Distributes over Addition on $\mathbb{Q}$

## 8.6 Field and Ordered-Field Structure of $\mathbb{R}$

**Theorem**

**Theorem 8.6.1** (Nonzero $\mathbb{R}$ Is a Multiplicative Abelian Group). *The structure* $(\mathbb{R} \setminus \{0_{\mathbb{R}}\}, \cdot, 1_{\mathbb{R}})$ *is an abelian group.*

*Remark* (Dependencies).

- Group
- Multiplication on $\mathbb{R}$
- Multiplication on $\mathbb{R}$ Is Associative
- Multiplication on $\mathbb{R}$ Is Commutative
- One Is a Multiplicative Identity on $\mathbb{R}$
- Reciprocal Is a Multiplicative Inverse on $\mathbb{R}$

**Theorem**

**Theorem 8.6.2** ($\mathbb{R}$ Is a Field). *The structure* $(\mathbb{R}, +, \cdot, 0_{\mathbb{R}}, 1_{\mathbb{R}})$ *is a field.*

*Remark* (Dependencies).

- $\mathbb{R}$ Is an Additive Abelian Group
- Nonzero $\mathbb{R}$ Is a Multiplicative Abelian Group
- Distributivity on $\mathbb{R}$
- Zero Is Not One in $\mathbb{R}$

**Corollary**

**Corollary 8.6.3** (Field Laws Hold on $\mathbb{R}$). *Every theorem proved generically for fields holds in $\mathbb{R}$, including zero product, double negation, sign rules, cancellation, uniqueness of inverses, inverse laws, and exponent laws.*

*Remark* (Dependencies).

- Generic Field Laws

### Order Compatibility

**Theorem**

**Theorem 8.6.4** (Addition Preserves Order on $\mathbb{R}$). *For all* $\alpha, \beta, \gamma \in \mathbb{R}$,

$$\alpha < \beta \Longrightarrow \alpha + \gamma < \beta + \gamma,$$

*and the corresponding non-strict statement also holds.*

*Remark* (Dependencies).

- Addition on $\mathbb{R}$
- Order on $\mathbb{R}$
- Strict Order on $\mathbb{R}$
- Addition Preserves Order on $\mathbb{Q}$

**Theorem**

**Theorem 8.6.5** (Multiplication by Positives Preserves Order on $\mathbb{R}$). *For all* $\alpha, \beta, \gamma \in \mathbb{R}$,

$$0_{\mathbb{R}} < \gamma \wedge \alpha < \beta \Longrightarrow \gamma \cdot \alpha < \gamma \cdot \beta.$$

*Remark* (Dependencies).

- Multiplication on $\mathbb{R}$
- Order on $\mathbb{R}$
- Strict Order on $\mathbb{R}$
- Positive Multiplication Preserves Order on $\mathbb{Q}$

**Theorem**

**Theorem 8.6.6** ($\mathbb{R}$ Is an Ordered Field). *The structure* $(\mathbb{R}, +, \cdot, 0_{\mathbb{R}}, 1_{\mathbb{R}}, \leq)$ *is an ordered field.*

*Remark* (Dependencies).

- $\mathbb{R}$ Is a Field
- $\mathbb{R}$ Is a Totally Ordered Set
- Addition Preserves Order on $\mathbb{R}$
- Multiplication by Positives Preserves Order on $\mathbb{R}$
- Ordered-Field Trichotomy

**Corollary**

**Corollary 8.6.7** (Ordered-Field Laws Hold on $\mathbb{R}$). *Every theorem proved generically for ordered fields holds in* $\mathbb{R}$.

*Remark* (Dependencies).

- Generic Ordered-Field Laws

**Theorem**

**Theorem 8.6.8** ($\mathbb{R}$ Is a Complete Ordered Field). *The ordered field* $\mathbb{R}$ *satisfies the least-upper-bound property.*

*Remark* (Dependencies).

- Complete Ordered Field
- $\mathbb{R}$ Is an Ordered Field
- Least-Upper-Bound Property

## 8.7 Embedding $\mathbb{Q}$ into $\mathbb{R}$

**Definition**

**Definition 8.7.1** (Embedding of $\mathbb{Q}$ into $\mathbb{R}$). Define $\iota : \mathbb{Q} \to \mathbb{R}$ by

$$\iota(q) := q^*.$$

*Remark* (Dependencies).

- Rational Numbers
- The Real Numbers
- Rational Cut

**Theorem**

**Theorem 8.7.2** (Embedding Preserves Zero and One). *The embedding satisfies*

$$\iota(0_{\mathbb{Q}}) = 0_{\mathbb{R}} \qquad \textit{and} \qquad \iota(1_{\mathbb{Q}}) = 1_{\mathbb{R}}.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Zero in $\mathbb{Q}$
- One in $\mathbb{Q}$
- Zero in $\mathbb{R}$
- One in $\mathbb{R}$

**Theorem**

**Theorem 8.7.3** (Embedding $\mathbb{Q}$ into $\mathbb{R}$ Is Injective). *For all* $p, q \in \mathbb{Q}$,

$$\iota(p) = \iota(q) \Longrightarrow p = q.$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Rational Cut
- $\mathbb{Q}$ Is a Totally Ordered Set

**Theorem**

**Theorem 8.7.4** (Embedding Preserves Addition and Multiplication). *For all* $p, q \in \mathbb{Q}$,

$$\iota(p + q) = \iota(p) + \iota(q) \qquad \textit{and} \qquad \iota(p \cdot q) = \iota(p) \cdot \iota(q).$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Addition on $\mathbb{R}$
- Multiplication on $\mathbb{R}$
- Addition on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$

**Theorem**

**Theorem 8.7.5** (Embedding Preserves Order). *For all* $p, q \in \mathbb{Q}$,

$$p \leq q \iff \iota(p) \leq \iota(q).$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Order on $\mathbb{Q}$
- Order on $\mathbb{R}$
- Rational Cut

**Theorem**

**Theorem 8.7.6** ($\mathbb{Q}$ Embeds into $\mathbb{R}$ as an Ordered Field). *The map* $\iota$ *is an injective order-preserving field homomorphism.*

*Remark* (Dependencies).

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Embedding Preserves Zero and One
- Embedding $\mathbb{Q}$ into $\mathbb{R}$ Is Injective
- Embedding Preserves Addition and Multiplication
- Embedding Preserves Order

**Theorem**

**Theorem 8.7.7** (Density of $\mathbb{Q}$ in $\mathbb{R}$). *For all* $\alpha, \beta \in \mathbb{R}$,

$$\alpha < \beta \implies \exists q \in \mathbb{Q}\,(\alpha < \iota(q) < \beta).$$

*Remark* (Dependencies).

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Strict Order on $\mathbb{R}$
- Density of $\mathbb{Q}$ in Itself

- Dedekind Cut

**Theorem**

**Theorem 8.7.8** (Archimedean Property of $\mathbb{R}$). *For* $\alpha, \beta \in \mathbb{R}$,

$$0_{\mathbb{R}} < \alpha \implies \exists n \in \mathbb{N}\,(\beta < \iota(n) \cdot \alpha).$$

*Remark* (Dependencies).

- Least-Upper-Bound Property

## 8.8 The Gap Filled

**Definition**

**Definition 8.8.1** (The Square-Root-of-Two Cut). Define

$$\sqrt{2} := \{r \in \mathbb{Q} : r < 0_{\mathbb{Q}} \vee r \cdot r < 2_{\mathbb{Q}}\}.$$

*Remark* (Dependencies).

- Rational Numbers
- Order on $\mathbb{Q}$
- Multiplication on $\mathbb{Q}$
- Two in $\mathbb{Q}$

**Lemma**

**Lemma 8.8.2** (The Square-Root-of-Two Cut Is a Cut). *The set* $\sqrt{2}$ *is a Dedekind cut.*

*Remark* (Dependencies).

- Density of $\mathbb{Q}$

**Theorem**

**Theorem 8.8.3** (Its Square Is Two). *In* $\mathbb{R}$,

$$\sqrt{2} \cdot \sqrt{2} = 2_{\mathbb{R}},$$

*where* $2_{\mathbb{R}} := \iota(2_{\mathbb{Q}})$.

*Remark* (Dependencies).

- The Square-Root-of-Two Cut
- The Square-Root-of-Two Cut Is a Cut
- Multiplication on $\mathbb{R}$

- Embedding of $\mathbb{Q}$ into $\mathbb{R}$
- Two in $\mathbb{Q}$

**Corollary**

**Corollary 8.8.4** ($\mathbb{R}$ Contains the Supremum $\mathbb{Q}$ Was Missing). *Let*

$$S = \{q \in \mathbb{Q} : 0_{\mathbb{Q}} < q \wedge q \cdot q < 2_{\mathbb{Q}}\}.$$

*Then*

$$\sqrt{2} = \sup \iota(S).$$

*Thus the bounded-above rational set with no rational supremum has a supremum in $\mathbb{R}$.*

*Remark* (Dependencies).

- The Gap Set in $\mathbb{Q}$
- The Gap Set Has No Rational Supremum

## 8.9 Characterization of $\mathbb{R}$

**Definition**

**Definition 8.9.1** (Complete Ordered Field). A *complete ordered field* is an ordered field satisfying the least-upper-bound property.

*Remark* (Dependencies).

- Ordered Field
- Least-Upper-Bound Property on $\mathbb{R}$

**Theorem**

**Theorem 8.9.2** (Uniqueness of the Complete Ordered Field). *Any two complete ordered fields are isomorphic by a unique order-isomorphism.*

*Remark* (Dependencies).

- Complete Ordered Field
- $\mathbb{R}$ Is a Complete Ordered Field

**Corollary**

**Corollary 8.9.3** ($\mathbb{R}$ Is the Real Numbers). *The field $\mathbb{R}$ is determined up to unique order-isomorphism by being a complete ordered field.*

*Remark* (Dependencies).

- Complete Ordered Field
- Uniqueness of the Complete Ordered Field
- $\mathbb{R}$ Is a Complete Ordered Field

## 8.10 Transition and Forward

*Remark* 8.10.1 (What Completeness Buys). The least-upper-bound property promotes the arbitrary closeness available in $\mathbb{Q}$ to guaranteed limits in $\mathbb{R}$: bounded monotone sequences converge, Cauchy sequences converge, and suprema exist. The general theory of these phenomena belongs to analysis; this chapter supplies the complete ordered field in which they live.

*Remark* 8.10.2 (Parallel Constructions). Cauchy sequences, nested intervals, and dyadic expansions give alternative constructions of complete ordered fields. By Uniqueness of the Complete Ordered Field, each yields this same $\mathbb{R}$ up to unique isomorphism.

# Chapter 9

# Summary: The Number Systems

**numbers-summary**

reals $\to$ **numbers-summary** $\to$ Embedding the Number Systems

## 9.1 Axiom Systems for $\mathbb{N}$, $\mathbb{W}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$

**Derived Properties Quick Reference**

| Name | Statement | Proof sketch |
|---|---|---|
| Zero multiplication | $a \cdot 0 = 0$ | $a \cdot 0 = a \cdot (0 + 0) = a \cdot 0 + a \cdot 0$; cancel $a \cdot 0$ via A4. |
| Negation rule | $(-a) \cdot b = -(ab)$ | $ab + (-a)b = (a + (-a))b = 0 \cdot b = 0$. |
| $(-1)$ rule | $(-1) \cdot a = -a$ | Special case: $b = a$, $-a = (-1)a$. |
| Double negation | $(-a)(-b) = ab$ | $(-a)(-b) = -(a(-b)) = -(-(ab)) = ab$. |
| No zero divisors | $ab = 0 \Rightarrow a = 0$ or $b = 0$ | $\mathbb{Z}$ is an integral domain; proved via well-ordering + sign analysis. |

### The Natural Numbers $\mathbb{N}$

**One-Based Peano Axioms for $\mathbb{N}$**

Let $\mathbb{N}$ be a set with distinguished element 1 and successor operation $n \mapsto n{+}{+}$.

**P1** $1 \in \mathbb{N}$.

**P2** $n \in \mathbb{N} \Rightarrow n{+}{+} \in \mathbb{N}$. *(successor closure)*

**P3** $n{+}{+} \neq 1$ for all $n \in \mathbb{N}$. *(the first element has no predecessor)*

**P4** $n{+}{+} = m{+}{+} \Rightarrow n = m$. *(successor is injective)*

**P5** If $P(0)$ holds and $P(n) \Rightarrow P(n{+}{+})$, then $P(n)$ holds for all $n \in \mathbb{N}$. *(induction)*

*Remark.* P1–P4 rule out wrap-around, ceiling, and rogue elements. P5 eliminates anything unreachable from 1 by finite iteration of the successor. Addition,

multiplication, and order are not axioms - they are *defined* recursively from the successor and then shown to satisfy familiar laws by induction.

*Remark* (Well-ordering and induction are equivalent over $\mathbb{N}$). The following are logically equivalent:

- Principle of Mathematical Induction (P5).
- Well-Ordering Principle: every nonempty $A \subseteq \mathbb{N}$ has a least element.
- Strong Induction: $(\forall k < n,\, P(k)) \Rightarrow P(n)$ for all $n$ implies $P$ holds everywhere

One must assume at least one; the others follow. Well-ordering is the engine behind the Division Algorithm, Bézout's Identity, and unique prime factorization

## The Whole Numbers $\mathbb{W}$

**$\mathbb{W}$ as $\mathbb{N}$ with an Adjoined Zero**

The whole numbers are

$$\mathbb{W} = \mathbb{N} \cup \{0\}, \qquad 0 \notin \mathbb{N}.$$

The successor, addition, multiplication, and order on $\mathbb{W}$ extend the corresponding structures on $\mathbb{N}$:

- $S_{\mathbb{W}}(0) = 1$, and $S_{\mathbb{W}}$ agrees with $S_{\mathbb{N}}$ on $\mathbb{N}$;
- 0 is the additive identity;
- 0 is absorbing for multiplication;
- 0 is the least element of $\mathbb{W}$.

*Remark* (What $\mathbb{W}$ gains over $\mathbb{N}$, and what it lacks). **Gain:** an additive identity This makes $(\mathbb{W}, +, 0)$ a commutative monoid and prepares the ordered-pair construction of the integers.

**Lack:** additive inverses for nonzero elements. Subtraction is still not an internal operation in general, so $\mathbb{W}$ is not a ring.

## The Integers $\mathbb{Z}$

**Axioms for $\mathbb{Z}$ Ordered Commutative Ring**

**Addition**

| | | |
|---|---|---|
| **A1** | $a + b = b + a$ | *(commutativity)* |
| **A2** | $(a + b) + c = a + (b + c)$ | *(associativity)* |
| **A3** | $\exists\, 0 \in \mathbb{Z}$ such that $a + 0 = a$ | *(additive identity)* |
| **A4** | $\forall\, a,\ \exists\, -a \in \mathbb{Z}$ with $a + (-a) = 0$ | *(additive inverses)* |

**Multiplication**

| | | |
|---|---|---|
| **M1** | $a \cdot b = b \cdot a$ | *(commutativity)* |
| **M2** | $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ | *(associativity)* |
| **M3** | $\exists\, 1 \in \mathbb{Z},\ 1 \neq 0$, such that $a \cdot 1 = a$ | *(multiplicative identity)* |

**Distributivity**

| | | |
|---|---|---|
| **D** | $a \cdot (b + c) = a \cdot b + a \cdot c$ | *(distributivity)* |

**Order**

| | | |
|---|---|---|
| **O1** | $a \leq b$ or $b \leq a$ | *(totality)* |
| **O2** | $a \leq b \wedge b \leq a \Rightarrow a = b$ | *(antisymmetry)* |
| **O3** | $a \leq b \wedge b \leq c \Rightarrow a \leq c$ | *(transitivity)* |
| **O4** | $b \leq c \Rightarrow a + b \leq a + c$ | *(order + addition)* |
| **O5** | $a \geq 0 \wedge b \geq 0 \Rightarrow ab \geq 0$ | *(order + multiplication)* |

*Remark* (What $\mathbb{Z}$ gains over $\mathbb{W}$, and what it lacks). **Gain:** Axiom A4 - additive inverses. Subtraction is always defined in $\mathbb{Z}$; it is not in $\mathbb{W}$.

**Lack:** Multiplicative inverses. There is no $x \in \mathbb{Z}$ with $2x = 1$, so M4 (every nonzero element is invertible) fails. In the language of abstract algebra, $\mathbb{Z}$ is a *commutative ordered ring* but not a field. The passage to $\mathbb{Q}$ restores multiplicative inverses by formally adjoining fractions.

## Derived Properties of $\mathbb{Z}$ (from the Ring Axioms)

*Remark* (Zero multiplication the step most often looked up). The identity $a \cdot 0 = 0$ is not an axiom; it is derived. The proof uses only A3, A4, and D:

$$a \cdot 0 = a \cdot (0 + 0) \overset{\text{D}}{=} a \cdot 0 + a \cdot 0.$$

Adding $-(a \cdot 0)$ to both sides (A4) gives $0 = a \cdot 0$, i.e. $a \cdot 0 = 0$. No order axioms, no well-ordering, no properties of $\mathbb{N}$ are needed this holds in any ring.

## Structural Comparison: $\mathbb{N}$, $\mathbb{W}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$

**Axiom Comparison Table**

| **Group** | **Axiom** | $\mathbb{N}$ | $\mathbb{W}$ | $\mathbb{Z}$ | $\mathbb{Q}$ |
|---|---|---|---|---|---|
| Addition | A1 commutativity | ✓ | ✓ | ✓ | ✓ |
| | A2 associativity | ✓ | ✓ | ✓ | ✓ |
| | A3 identity (0) | × | ✓ | ✓ | ✓ |
| | A4 inverses ($-a$) | × | × | ✓ | ✓ |
| Multiplication | M1 commutativity | ✓ | ✓ | ✓ | ✓ |
| | M2 associativity | ✓ | ✓ | ✓ | ✓ |
| | M3 identity (1) | ✓ | ✓ | ✓ | ✓ |
| | M4 inverses ($1/a$) | × | × | × | ✓ |
| | D distributivity | ✓ | ✓ | ✓ | ✓ |
| Order | O1–O3 total order | ✓ | ✓ | ✓ | ✓ |
| | O4–O5 order + arithmetic | ✓ | ✓ | ✓ | ✓ |
| | Well-ordering | ✓ | ✓ | × | × |
| Completeness | Least upper bound | × | × | × | × |
| **Structure** | | *positive arithmetic* | *comm. semiring* | *comm. ring* | *ordered* |

✓ = satisfied × = fails

*Remark* (The extension hierarchy). Each number system repairs exactly one deficiency of its predecessor:

$$\mathbb{N} \xrightarrow{+\,0} \mathbb{W} \xrightarrow{+\,\mathrm{A4}} \mathbb{Z} \xrightarrow{+\,\mathrm{M4}} \mathbb{Q} \xrightarrow{+\,\mathrm{LUB}} \mathbb{R}.$$

$\mathbb{N} \to \mathbb{W}$: adjoin 0 so addition has an identity. $\mathbb{W} \to \mathbb{Z}$: adjoin additive inverses so subtraction is always defined. $\mathbb{Z} \to \mathbb{Q}$: adjoin multiplicative inverses so division by any nonzero element is always defined. $\mathbb{Q} \to \mathbb{R}$: adjoin least upper bounds so every bounded increasing sequence converges.

Note that $\mathbb{N}$ and $\mathbb{W}$ are the well-ordered systems in the chain. $\mathbb{Z}$ loses well-ordering the moment negative integers are introduced (the set of all negative integers has no least element).

### From Extension to Embedding

The hierarchy above describes what each new system adds. The next chapter looks at a separate question: how the earlier systems are identified inside the later ones. That identification is not merely a matter of writing subset symbols. It is carried by structure-preserving embeddings that justify treating a number from an earlier construction as the corresponding number in a larger system.

# Chapter 10

# Embedding the Number Systems

**embedding-number-systems**

numbers-summary → **Embedding the Number Systems** →
Number Lines and Intervals

## 10.1 Embeddings as Structure Preserving Maps

### From Construction to Identification

The preceding chapters built $\mathbb{N}$, $\mathbb{W}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ one at a time. In that construction-first development the symbol 2 denotes a different object in each system: a Peano element, a whole number, an integer difference object, a rational fraction object, and a real number are literally distinct sets. Nothing in the constructions has made them identical.

An *embedding* is the device that closes this gap without pretending the objects were the same to begin with. It is an injective map that preserves the relevant structure present in the source system — the distinguished constants, the operations, and the order being considered. Once such a map is fixed, the source system is identified with its image inside the target, and only then is the familiar chain $\mathbb{N} \subseteq \mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R}$ written down. The inclusion symbol records an identification along an embedding, not a literal subset relation.

**Definition (Structure-Preserving Map)**

**Definition 10.1.1** (Structure-Preserving Map). Let $A$ and $B$ be ordered arithmetic systems with a specified common source structure: selected constants, selected operations, and a strict order. A map $\varphi : A \to B$ is *structure-preserving* exactly when it preserves each selected constant and, for every selected binary operation $\star$ and all $a, a' \in A$,

$$\varphi(a \star_A a') = \varphi(a) \star_B \varphi(a'),$$

and $a <_A a' \Longrightarrow \varphi(a) <_B \varphi(a')$.

*Remark* (Standard quantified statement).

$$\begin{aligned}
&\varphi : A \to B \text{ is structure-preserving} \\
\iff &\Big(\text{every selected constant is preserved} \\
&\quad \wedge \text{ every selected operation is preserved} \\
&\quad \wedge \forall a, a' \in A\ (a <_A a' \Rightarrow \varphi(a) <_B \varphi(a'))\Big).
\end{aligned}$$

*Remark* (Interpretation). A structure-preserving map carries the chosen arithmetic and order from $A$ into $B$ without distortion. The choice of structure matters: the positive part of $\mathbb{W}$ carries the Peano successor from $\mathbb{N}$, while the later ring and field embeddings focus on $0$, $1$, addition, multiplication, and order. A structure-preserving map need not be injective; collapsing distinct elements is still allowed at this stage. Injectivity is added separately to obtain an embedding.

*Remark* (Dependencies).

- Ordered Field
- $\mathbb{N}$ Embeds into $\mathbb{W}$ as the Positive Part

**Definition (Order Embedding)**

**Definition** (Order Embedding). Let $A$ and $B$ be ordered arithmetic systems and $\varphi : A \to B$ a map. Then $\varphi$ is an *order embedding* exactly when, for all $a, a' \in A$,

$$a <_A a' \iff \varphi(a) <_B \varphi(a').$$

**Definition (Embedding of Number Systems)**

**Definition 10.1.2** (Embedding of Number Systems). Let $A$ and $B$ be ordered arithmetic systems. A map $\varphi : A \to B$ is an *embedding* exactly when $\varphi$ is injective and structure-preserving and is an order embedding.

*Remark* (Standard quantified statement).

$$\begin{aligned}
&\varphi \text{ is an embedding} \\
\iff &\Big(\varphi \text{ is injective} \wedge \varphi \text{ is structure-preserving} \\
&\quad \wedge \forall a, a' \in A\ (a <_A a' \Leftrightarrow \varphi(a) <_B \varphi(a'))\Big).
\end{aligned}$$

*Remark* (Negated quantified statement).

$$\begin{aligned}
&\varphi \text{ is not an embedding} \\
\iff &\Big(\exists a, a' \in A\ (a \neq a' \wedge \varphi(a) = \varphi(a')) \\
&\quad \vee \ \varphi \text{ fails to preserve a constant, sum, or product} \\
&\quad \vee \ \exists a, a' \in A\ \big((a <_A a' \wedge \neg(\varphi(a) <_B \varphi(a'))) \vee (\varphi(a) <_B \varphi(a') \wedge \neg(a <_A a'))\big)\Big).
\end{aligned}$$

*Remark* (Failure modes). An embedding can fail in three structurally distinct ways: it may collapse two distinct elements to the same image (injectivity fails), fail to transport one of the selected constants or operations (algebraic preservation fails), or fail to preserve or reflect a strict-order comparison (order embedding fails).

*Remark* (Interpretation). An embedding is the precise meaning of "$A$ sits inside $B$." It is injective, so distinct elements stay distinct; it preserves the arithmetic and reflects the order, so the copy of $A$ inside $B$ behaves exactly as $A$ did. This is what licenses treating $2 \in \mathbb{N}$ and $2 \in \mathbb{R}$ as "the same number."

*Remark* (Interpretation). Most embeddings after $\mathbb{W}$ transport the *ordered ring/field* structure: constants, the two operations, and order. The first bridge $\mathbb{N} \hookrightarrow \mathbb{W}$ is slightly different because $\mathbb{N}$ is one-based and carries Peano successor. That local theorem is proved in the whole-number chapter. In every case, induction is not transported to the larger target: the phrase "the naturals sit inside the integers" must not be read as carrying induction across all of $\mathbb{Z}$.

*Remark* (Dependencies).

- Structure-Preserving Map
- Order Embedding

**Definition (Image of an Embedding)**

**Definition 10.1.3** (Image of an Embedding). Let $\varphi : A \to B$ be an embedding. The *image* of $\varphi$ is the set $\varphi(A) = \{\varphi(a) : a \in A\} \subseteq B$, equipped with the selected constants, operations, and order of $B$ restricted to $\varphi(A)$.

*Remark* (Standard quantified statement).

$$\varphi(A) = \{\, b \in B : \exists a \in A\ (b = \varphi(a)) \,\}.$$

*Remark* (Interpretation). The image is the literal subset of $B$ that plays the role of $A$. The identification theorem below states that $\varphi$, viewed as a map onto its image, is an isomorphism, so $A$ and $\varphi(A)$ are the same ordered arithmetic system under two names.

*Remark* (Dependencies).

- Embedding of Number Systems

**Theorem (Embeddings Identify a System with Its Image)**

**Theorem 10.1.4** (Embeddings Identify a System with Its Image). *Let $\varphi : A \to B$ be an embedding of ordered arithmetic systems relative to a selected source structure. Then the corestriction $\varphi : A \to \varphi(A)$ is an isomorphism onto its image: it is a bijection that preserves and reflects the selected constants, operations, and order. Consequently $A$ may be identified with the corresponding substructure $\varphi(A) \subseteq B$. Go to proof.*

*Remark* (Standard quantified statement).

$\varphi : A \to B$ an embedding
$\implies \varphi : A \to \varphi(A)$ is a bijective isomorphism for the selected structure.

*Remark* (Interpretation). This is the theorem that makes the inclusion notation legitimate. After identification, $A \subseteq B$ abbreviates "$A$ embeds via $\varphi$ as the substructure $\varphi(A)$." Every later use of $\mathbb{N} \subseteq \mathbb{R}$ is shorthand for this identificatio

*Remark* (Dependencies).

- Embedding of Number Systems
- Image of an Embedding

## 10.2 Embedding N Into W

**Embedding $\mathbb{N}$ into $\mathbb{W}$**

This section records, from the global embedding viewpoint, the local bridge proved in the whole-number chapter: $\mathbb{N}$ sits inside $\mathbb{W}$ as the positive part. The map is the bridge that lets elements of $\mathbb{N}$ be regarded as nonzero elements of $\mathbb{W}$ after identification.

**Definition (Embedding $\mathbb{N}$ into $\mathbb{W}$)**

**Definition 10.2.1** (Embedding $\mathbb{N}$ into $\mathbb{W}$). The *canonical embedding* $\iota : \mathbb{N} \to \mathbb{W}$ is defined by

$$\iota(n) = n \quad \text{(the copy of each natural number inside the whole numbers).}$$

*Remark* (Standard quantified statement).

$\iota : \mathbb{N} \to \mathbb{W}, \qquad \iota(n) = n$ (the copy of each natural number inside the whole numbers).

*Remark* (Interpretation). The map sends each element of $\mathbb{N}$ to its natural representativ in $\mathbb{W}$. The next result certifies that this representative choice preserves all arithmetic and order, so $\mathbb{N}$ may be identified with its image in $\mathbb{W}$.

*Remark* (Dependencies).

- Construction of $\mathbb{W}$ from $\mathbb{N}$ by adjoining 0
- $\mathbb{N}$ Embeds into $\mathbb{W}$ as the Positive Part

**Theorem ($\mathbb{N}$ Embeds in $\mathbb{W}$)**

**Theorem 10.2.2** ($\mathbb{N}$ Embeds in $\mathbb{W}$). *The canonical map $\iota : \mathbb{N} \to \mathbb{W}$ identifies $\mathbb{N}$ with the positive part $\mathbb{W} \setminus \{0\}$ and preserves successor, 1, addition, multiplication, and order. Go to proof.*

*Remark* (Standard quantified statement).

$\iota : \mathbb{N} \to \mathbb{W} \setminus \{0\}$ is bijective and preserves successor, arithmetic, and order.

*Remark* (Interpretation). Once this theorem is in place, $\mathbb{N}$ may be treated as the positive substructure $\iota(\mathbb{N}) = \mathbb{W} \setminus \{0\}$ inside $\mathbb{W}$. The new element of $\mathbb{W}$ is exactly the adjoined zero.

*Remark* (Dependencies).

- Construction of $\mathbb{W}$ from $\mathbb{N}$ by adjoining 0
- Embedding $\mathbb{N}$ into $\mathbb{W}$
- $\mathbb{N}$ Embeds into $\mathbb{W}$ as the Positive Part

## 10.3 Embedding W Into Z

### Embedding $\mathbb{W}$ into $\mathbb{Z}$

This section fixes the canonical embedding of $\mathbb{W}$ into $\mathbb{Z}$ and records that it is injective, structure-preserving, and order-reflecting. The map is the bridge that lets elements of $\mathbb{W}$ be regarded as elements of $\mathbb{Z}$ after identification.

**Definition (Embedding $\mathbb{W}$ into $\mathbb{Z}$)**

**Definition 10.3.1** (Embedding $\mathbb{W}$ into $\mathbb{Z}$). The *canonical embedding* $\iota : \mathbb{W} \to \mathbb{Z}$ is defined by

$$\iota(w) = [(w, 0)] \quad \text{(the integer represented by } w \text{ and } 0\text{)}.$$

*Remark* (Standard quantified statement).

$$\iota : \mathbb{W} \to \mathbb{Z}, \qquad \iota(w) = [(w, 0)] \quad \text{(the integer represented by } w \text{ and } 0\text{)}.$$

*Remark* (Interpretation). The map sends each element of $\mathbb{W}$ to its natural representati in $\mathbb{Z}$. The next result certifies that this representative choice preserves all arithmetic and order, so $\mathbb{W}$ may be identified with its image in $\mathbb{Z}$.

*Remark* (Dependencies).

- Formal-Difference Equivalence Is an Equivalence Relation
- Embedding of Number Systems

**Theorem ($\mathbb{W}$ Embeds in $\mathbb{Z}$)**

**Theorem 10.3.2** ($\mathbb{W}$ Embeds in $\mathbb{Z}$). *The canonical map $\iota : \mathbb{W} \to \mathbb{Z}$ is an embedding of ordered arithmetic systems: it is injective, preserves 0, 1, addition, and multiplication, and reflects the order. Go to proof.*

*Remark* (Standard quantified statement).

$\iota : \mathbb{W} \to \mathbb{Z}$ is injective, structure-preserving, and order-reflecting.

*Remark* (Interpretation). Once this theorem is proved, the identification theorem of §10.1 applies, and $\mathbb{W}$ may be treated as the substructure $\iota(\mathbb{W}) \subseteq \mathbb{Z}$.

*Remark* (Dependencies).

- Formal-Difference Equivalence Is an Equivalence Relation
- Embedding $\mathbb{W}$ into $\mathbb{Z}$
- Embeddings Identify a System with Its Image

## 10.4 Embedding Z Into Q

### Embedding $\mathbb{Z}$ into $\mathbb{Q}$

This section fixes the canonical embedding of $\mathbb{Z}$ into $\mathbb{Q}$ and records that it is injective, structure-preserving, and order-reflecting. The map is the bridge that lets elements of $\mathbb{Z}$ be regarded as elements of $\mathbb{Q}$ after identification.

**Definition (Embedding $\mathbb{Z}$ into $\mathbb{Q}$)**

**Definition 10.4.1** (Embedding $\mathbb{Z}$ into $\mathbb{Q}$). The *canonical embedding* $\iota : \mathbb{Z} \to \mathbb{Q}$ is defined by

$$\iota(z) = [(z, 1)] \quad \text{(the rational represented by } z/1).$$

*Remark* (Standard quantified statement).

$$\iota : \mathbb{Z} \to \mathbb{Q}, \qquad \iota(z) = [(z, 1)] \quad \text{(the rational represented by } z/1).$$

*Remark* (Interpretation). The map sends each element of $\mathbb{Z}$ to its natural representativ in $\mathbb{Q}$. The next result certifies that this representative choice preserves all arithmetic and order, so $\mathbb{Z}$ may be identified with its image in $\mathbb{Q}$.

*Remark* (Dependencies).

- Construction of $\mathbb{Q}$ as fraction classes of integers
- Embedding of Number Systems

**Theorem ($\mathbb{Z}$ Embeds in $\mathbb{Q}$)**

**Theorem 10.4.2** ($\mathbb{Z}$ Embeds in $\mathbb{Q}$). *The canonical map $\iota : \mathbb{Z} \to \mathbb{Q}$ is an embedding of ordered arithmetic systems: it is injective, preserves $0$, $1$, addition, and multiplication, and reflects the order. Go to proof.*

*Remark* (Standard quantified statement).

$$\iota : \mathbb{Z} \to \mathbb{Q} \text{ is injective, structure-preserving, and order-reflecting.}$$

*Remark* (Interpretation). Once this theorem is proved, the identification theorem of §10.1 applies, and $\mathbb{Z}$ may be treated as the substructure $\iota(\mathbb{Z}) \subseteq \mathbb{Q}$.

*Remark* (Dependencies).

- Construction of $\mathbb{Q}$ as fraction classes of integers
- Embedding $\mathbb{Z}$ into $\mathbb{Q}$
- Embeddings Identify a System with Its Image

## 10.5 Embedding Q Into R

**Embedding $\mathbb{Q}$ into $\mathbb{R}$**

This section fixes the canonical embedding of $\mathbb{Q}$ into $\mathbb{R}$ and records that it is injective, structure-preserving, and order-reflecting. The map is the bridge that lets elements of $\mathbb{Q}$ be regarded as elements of $\mathbb{R}$ after identification.

**Definition (Embedding $\mathbb{Q}$ into $\mathbb{R}$)**

**Definition 10.5.1** (Embedding $\mathbb{Q}$ into $\mathbb{R}$). The *canonical embedding* $\iota : \mathbb{Q} \to \mathbb{R}$ is defined by

$$\iota(q) = q^* \quad \text{(the real number determined by } q\text{)}.$$

*Remark* (Standard quantified statement).

$$\iota : \mathbb{Q} \to \mathbb{R}, \qquad \iota(q) = q^* \quad \text{(the real number determined by } q\text{)}.$$

*Remark* (Interpretation). The map sends each element of $\mathbb{Q}$ to its natural representativ in $\mathbb{R}$. The next result certifies that this representative choice preserves all arithmetic and order, so $\mathbb{Q}$ may be identified with its image in $\mathbb{R}$.

*Remark* (Dependencies).

- Construction of $\mathbb{R}$ by Dedekind cuts
- Embedding of Number Systems

**Theorem ($\mathbb{Q}$ Embeds in $\mathbb{R}$)**

**Theorem 10.5.2** ($\mathbb{Q}$ Embeds in $\mathbb{R}$). *The canonical map $\iota : \mathbb{Q} \to \mathbb{R}$ is an embedding of ordered arithmetic systems: it is injective, preserves $0$, $1$, addition, and multiplication, and reflects the order. Go to proof.*

*Remark* (Standard quantified statement).

$$\iota : \mathbb{Q} \to \mathbb{R} \text{ is injective, structure-preserving, and order-reflecting.}$$

*Remark* (Interpretation). Once this theorem is proved, the identification theorem of §10.1 applies, and $\mathbb{Q}$ may be treated as the substructure $\iota(\mathbb{Q}) \subseteq \mathbb{R}$.

*Remark* (Dependencies).

- Construction of $\mathbb{R}$ by Dedekind cuts
- Embedding $\mathbb{Q}$ into $\mathbb{R}$
- Embeddings Identify a System with Its Image

## 10.6 Compatibility of Arithmetic and Order

### One Arithmetic Through the Chain

Each embedding preserves arithmetic and order individually. What remains is to confirm that the embeddings are mutually compatible: composing $\mathbb{N} \hookrightarrow \mathbb{W} \hookrightarrow \mathbb{Z} \hookrightarrow \mathbb{Q} \hookrightarrow \mathbb{R}$ agrees with the direct embedding of $\mathbb{N}$ into $\mathbb{R}$, so that "2" has one unambiguous meaning after all identifications.

**Theorem (Compatibility of the Embedding Chain)**

**Theorem 10.6.1** (Compatibility of the Embedding Chain). *Let $\iota_1 : \mathbb{N} \to \mathbb{W}$, $\iota_2 : \mathbb{W} \to \mathbb{Z}$, $\iota_3 : \mathbb{Z} \to \mathbb{Q}$, and $\iota_4 : \mathbb{Q} \to \mathbb{R}$ be the canonical embeddings. For every $n \in \mathbb{N}$ the composite $(\iota_4 \circ \iota_3 \circ \iota_2 \circ \iota_1)(n)$ is the real number identified with $n$, and the composite preserves the arithmetic and order carried by the source system. Hence arithmetic computed in any system agrees with arithmetic on its image in $\mathbb{R}$. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall n, m \in \mathbb{N}\ \big(\Phi(n{+}m) = \Phi(n){+}\Phi(m)\ \wedge\ \Phi(n{\cdot}m) = \Phi(n){\cdot}\Phi(m)\ \wedge\ (n < m \Leftrightarrow \Phi(n) < \Phi(m))\big), \quad \Phi = \iota_4 \circ$$

*Remark* (Interpretation). Compatibility is what makes the single symbol 2 well-defined across the whole chain: the answer to $2{+}2$ is the same whether computed in $\mathbb{N}$ or after embedding into $\mathbb{R}$. Without this coherence the inclusion notation would silently carry two different meanings for the same numeral.

*Remark* (Dependencies).

- $\mathbb{N}$ Embeds into $\mathbb{W}$ as the Positive Part
- $\mathbb{W}$ Embeds in $\mathbb{Z}$
- $\mathbb{Z}$ Embeds in $\mathbb{Q}$
- $\mathbb{Q}$ Embeds in $\mathbb{R}$

## 10.7 One Chain Many Structures

### One Chain, Many Structures

The embeddings assemble into a single chain $\mathbb{N} \hookrightarrow \mathbb{W} \hookrightarrow \mathbb{Z} \hookrightarrow \mathbb{Q} \hookrightarrow \mathbb{R}$. Each arrow is injective, preserves the relevant arithmetic, and reflects order; each step also *adds* structure that the previous system lacked.

From $\mathbb{N}$ to $\mathbb{W}$ an additive identity 0 appears. From $\mathbb{W}$ to $\mathbb{Z}$ additive inverses appear, completing a commutative ring. From $\mathbb{Z}$ to $\mathbb{Q}$ multiplicative inverses of nonzero elements appear, completing an ordered field. From $\mathbb{Q}$ to $\mathbb{R}$ order-completeness appears: the gaps in the rational line are filled. The numerals carry through unchanged by compatibility, but the ambient structure grows at every step. This chain is the bridge from number construction to the analysis of the real line.

*Remark* (Exposition). Read as a ladder of structures: counting ($\mathbb{N}$) $\to$ counting with zero ($\mathbb{W}$) $\to$ signed counting / ring ($\mathbb{Z}$) $\to$ ordered field ($\mathbb{Q}$) $\to$ complete ordered field ($\mathbb{R}$). Only the last rung is order-complete, and that completeness is the property the structure-of-the-real-line chapter turns into metric topology.

# Chapter 11

# Number Lines and Intervals

**number-lines**

Embedding the Number Systems → **Number Lines and Intervals** → Structure of the Real Line

## 11.1 Number Lines

### The Geometric Model of an Ordered System

A number line is the geometric realization of a linearly ordered system: points on a line arranged so that "left of" matches the order. The construction so far is algebraic; the number line is the picture that accompanies it, and it becomes exact for $\mathbb{R}$.

**Definition (Number Line)**

**Definition 11.1.1** (Number Line). A *number line* for a linearly ordered system $(A, <_A)$ is a linearly ordered set of points $L$ together with an order isomorphism $\lambda : A \to L$, so that for all $a, a' \in A$,

$$a <_A a' \iff \lambda(a) \text{ lies to the left of } \lambda(a').$$

*Remark* (Standard quantified statement).

$$\lambda : A \to L \text{ is a bijection with } \forall a, a' \in A\ (a <_A a' \Leftrightarrow \lambda(a) \prec \lambda(a')).$$

*Remark* (Interpretation). The number line records only order, not arithmetic: it is the visual home of betweenness and length. For $\mathbb{R}$ the order isomorphism is onto a full continuum, which is why the real line has no gaps; for $\mathbb{Q}$ the same picture has holes at every irrational location.

*Remark* (Dependencies).

- Order Embedding

## 11.2 Intervals on a Number Line

### The Basic Pieces of the Line

Intervals are the natural named pieces of a number line. This section gives each basic interval shape its notation and spells out which endpoints are included.

**Definition (Open Interval)**

**Definition 11.2.1** (Open Interval). Let $a, b \in \mathbb{R}$ with $a \leq b$. The *open interval* with endpoints $a$ and $b$ is

$$(a, b) = \{\, x \in \mathbb{R} : a < x < b \,\}.$$

*Remark* (Standard quantified statement).

$$(a, b) = \{\, x \in \mathbb{R} : a < x < b \,\}.$$

*Remark* (Interpretation). The open interval excludes both endpoints. It is one of the four bounded interval shapes.

*Remark* (Dependencies).

- Ordered Field

**Definition (Closed Interval)**

**Definition 11.2.2** (Closed Interval). Let $a, b \in \mathbb{R}$ with $a \leq b$. The *closed interval* with endpoints $a$ and $b$ is

$$[a, b] = \{\, x \in \mathbb{R} : a \leq x \leq b \,\}.$$

*Remark* (Standard quantified statement).

$$[a, b] = \{\, x \in \mathbb{R} : a \leq x \leq b \,\}.$$

*Remark* (Interpretation). The closed interval includes both endpoints. It is one of the four bounded interval shapes.

*Remark* (Dependencies).

- Ordered Field

**Definition (Left-Half-Open Interval)**

**Definition 11.2.3** (Left-Half-Open Interval). Let $a, b \in \mathbb{R}$ with $a \leq b$. The *left-half-open interval* with endpoints $a$ and $b$ is

$$(a, b] = \{\, x \in \mathbb{R} : a < x \leq b \,\}.$$

*Remark* (Standard quantified statement).

$$(a, b] = \{\, x \in \mathbb{R} : a < x \leq b \,\}.$$

*Remark* (Interpretation). The left-half-open interval excludes the left endpoint, includes the right. It is one of the four bounded interval shapes.

*Remark* (Dependencies).

- Ordered Field

**Definition (Right-Half-Open Interval)**

**Definition 11.2.4** (Right-Half-Open Interval). Let $a, b \in \mathbb{R}$ with $a \leq b$. The *right-half-open interval* with endpoints $a$ and $b$ is

$$[a, b) = \{\, x \in \mathbb{R} : a \leq x < b \,\}.$$

*Remark* (Standard quantified statement).

$$[a, b) = \{\, x \in \mathbb{R} : a \leq x < b \,\}.$$

*Remark* (Interpretation). The right-half-open interval includes the left endpoint, excludes the right. It is one of the four bounded interval shapes.

*Remark* (Dependencies).

- Ordered Field

**Definition (Ray)**

**Definition 11.2.5** (Ray). Let $a \in \mathbb{R}$. The four *rays* determined by $a$ are

$$(a, \infty) = \{x : x > a\}, \quad [a, \infty) = \{x : x \geq a\}, \quad (-\infty, a) = \{x : x < a\}, \quad (-\infty, a] = \{x : x \leq a\}.$$

*Remark* (Standard quantified statement).

$$\begin{aligned} (a, \infty) &= \{\, x \in \mathbb{R} : a < x \,\}, & [a, \infty) &= \{\, x \in \mathbb{R} : a \leq x \,\}, \\ (-\infty, a) &= \{\, x \in \mathbb{R} : x < a \,\}, & (-\infty, a] &= \{\, x \in \mathbb{R} : x \leq a \,\}. \end{aligned}$$

*Remark* (Interpretation). Rays are the unbounded one-sided intervals. Together with the bounded intervals and the whole line $(-\infty, \infty) = \mathbb{R}$, they exhaust the interval shapes.

*Remark* (Dependencies).

- Ordered Field

# Chapter 12

# Structure of the Real Line

**structure-of-real-line**

Number Lines and Intervals → **Structure of the Real Line** → Formalizing the Number Systems

## 12.1 Real Line One Dimensional

### The Line as Order Plus Distance

The real numbers carry two compatible structures at once: a linear order $<$ and a notion of distance $d(x, y) = |x - y|$. Together these make $\mathbb{R}$ a one-dimensional space — a line. This chapter develops only the metric topology of $\mathbb{R}$, not topology in general spaces. Every open set is built from intervals, and compactness is introduced only on $\mathbb{R}$, with Heine–Borel as the capstone.

The intuition is geometric: points are positions on a line, order is "left of," and distance is the length of the segment between two points. The caution to keep throughout is that $\mathbb{R}$ is special. Order and metric agree here in a way that fails on a general space, so definitions are stated in the form that survives to metric spaces and manifolds, with $\mathbb{R}$ as the first instance rather than the only one.

## 12.2 Order Distance Length

### Distance on the Line

The absolute value of a difference turns $\mathbb{R}$ into a metric space. Length of an interval is the distance between its endpoints. These are the quantitative tools underneath every later definition of nearness.

**Definition (Distance on the Real Line)**

**Definition 12.2.1** (Distance on the Real Line). The *distance* on $\mathbb{R}$ is the map $d : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ given by

$$d(x, y) = |x - y|.$$

*Remark* (Standard quantified statement).

$$\forall x, y \in \mathbb{R}\ (d(x,y) = |x-y|).$$

*Remark* (Interpretation). The distance between two points is the length of the segment joining them. It is the metric that makes "close" precise and is the basis for balls, neighborhoods, and open sets.

*Remark* (Dependencies).

- Triangle Inequality

**Definition (Length of a Bounded Interval)**

**Definition 12.2.2** (Length of a Bounded Interval). Let $a, b \in \mathbb{R}$ with $a \leq b$. The *length* of any bounded interval with endpoints $a$ and $b$ is

$$\ell = b - a.$$

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{R}\ \big(a \leq b \Longrightarrow \ell([a,b]) = b - a = d(a,b)\big).$$

*Remark* (Interpretation). Length depends only on the endpoints, not on whether they are included. It is the distance $d(a,b)$ and measures the size of the basic pieces of the line.

*Remark* (Dependencies).

- Distance on the Real Line

**Theorem (The Distance Function Is a Metric)**

**Theorem 12.2.3** (The Distance Function Is a Metric). *The map $d(x,y) = |x-y|$ satisfies, for all $x, y, z \in \mathbb{R}$: (i) $d(x,y) \geq 0$ with $d(x,y) = 0$ iff $x = y$; (ii) $d(x,y) = d(y,x)$; and (iii) $d(x,z) \leq d(x,y) + d(y,z)$. Hence $(\mathbb{R}, d)$ is a metric space. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall x, y, z \in \mathbb{R}\ \big(d(x,y) \geq 0\ \wedge\ (d(x,y) = 0 \Leftrightarrow x = y)\ \wedge\ d(x,y) = d(y,x)\ \wedge\ d(x,z) \leq d(x,y) + d(y,z)\big).$$

*Remark* (Interpretation). The three metric axioms are exactly the properties of $|\cdot|$: nonnegativity with the zero-distance characterization, symmetry, and the triangle inequality. This is the precise sense in which $\mathbb{R}$ is a one-dimensional metric space.

*Remark* (Dependencies).

- Distance on the Real Line
- Triangle Inequality

## 12.3 Absolute Value as Distance

### Absolute Value Is Distance to Zero

The metric $d$ specializes to absolute value when one point is the origin. This is the geometric reading of $|a|$: how far $a$ sits from 0 on the line.

**Proposition (Absolute Value Is Distance to the Origin)**

**Proposition 12.3.1** (Absolute Value Is Distance to the Origin). *For every* $a \in \mathbb{R}$, $|a| = d(a, 0)$. *Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a \in \mathbb{R}\ (|a| = d(a, 0)).$$

*Remark* (Interpretation). Absolute value is not a separate idea from distance; it is the distance to the origin. The two-sided bound $|x| \leq r \Leftrightarrow -r \leq x \leq r$ is then exactly the statement that $x$ lies within distance $r$ of $0$, i.e. in the interval $[-r, r]$.

*Remark* (Dependencies).

- Distance on the Real Line

## 12.4 Intervals as Subsets

### Intervals as the Basic Pieces

The interval shapes were introduced in §11.2. Here they are recast as subsets of the metric line and supplemented with boundedness, which is the size condition compactness will need.

**Definition (Bounded Subset of the Real Line)**

**Definition 12.4.1** (Bounded Subset of the Real Line). A set $E \subseteq \mathbb{R}$ is *bounded* exactly when there exists $M \geq 0$ such that $|x| \leq M$ for all $x \in E$; equivalently, $E$ is contained in some closed interval $[-M, M]$.

*Remark* (Standard quantified statement).

$$E \text{ is bounded} \iff \exists M \geq 0\ \forall x \in E\ (|x| \leq M).$$

*Remark* (Negated quantified statement).

$$E \text{ is unbounded} \iff \forall M \geq 0\ \exists x \in E\ (|x| > M).$$

*Remark* (Interpretation). Boundedness says $E$ fits inside a single closed interval -- it does not run off to infinity. Together with closedness it is one half of the Heine-Borel characterization of compact subsets of $\mathbb{R}$.

*Remark* (Dependencies).

- Closed Interval
- Triangle Inequality

## Set Operations on Intervals

### Intervals as Sets

Once intervals are regarded as subsets of $\mathbb{R}$, the usual set operations apply to them. This is the first place where intervals behave like the building blocks of topology rather than merely as pieces of notation on the number line. Intersections often remain intervals, while unions, complements, and differences may split into several interval pieces.

**Definition (Union of Intervals)**

**Definition 12.4.2** (Union of Intervals). Let $I, J \subseteq \mathbb{R}$ be intervals. Their *union* is the set

$$I \cup J = \{x \in \mathbb{R} : x \in I \text{ or } x \in J\}.$$

*Remark* (Dependencies).

- Open Interval
- Closed Interval
- Union

**Definition (Intersection of Intervals)**

**Definition 12.4.3** (Intersection of Intervals). Let $I, J \subseteq \mathbb{R}$ be intervals. Their *intersection* is the set

$$I \cap J = \{x \in \mathbb{R} : x \in I \text{ and } x \in J\}.$$

*Remark* (Dependencies).

- Open Interval
- Closed Interval
- Intersection

**Definition (Complement of an Interval)**

**Definition 12.4.4** (Complement of an Interval). Let $I \subseteq \mathbb{R}$ be an interval. Its *complement* in $\mathbb{R}$ is the set

$$I^c = \mathbb{R} \setminus I = \{x \in \mathbb{R} : x \notin I\}.$$

*Remark* (Dependencies).

- Open Interval
- Closed Interval
- Complement

**Definition (Difference of Intervals)**

**Definition 12.4.5** (Difference of Intervals). Let $I, J \subseteq \mathbb{R}$ be intervals. The *difference* of $I$ by $J$ is the set

$$I \setminus J = \{x \in \mathbb{R} : x \in I \text{ and } x \notin J\} = I \cap J^c.$$

*Remark* (Predicate readings). For any $x \in \mathbb{R}$,

$$x \in I \cup J \iff (x \in I) \vee (x \in J),$$

$$x \in I \cap J \iff (x \in I) \wedge (x \in J),$$

$$x \in I^c \iff x \notin I, \qquad x \in I \setminus J \iff (x \in I) \wedge (x \notin J).$$

These are set-theoretic operations; no arithmetic operation is being performed on the endpoints.

*Remark* (Dependencies).

- Open Interval
- Closed Interval
- Left-Half-Open Interval
- Right-Half-Open Interval
- Union of Intervals
- Intersection of Intervals
- Complement of an Interval

*Remark* (Intersections of intervals). The intersection of two intervals is either empty or another interval. For example, if $a \leq b$ and $c \leq d$, then

$$[a, b] \cap [c, d] = \begin{cases} [\max\{a, c\}, \min\{b, d\}], & \max\{a, c\} \leq \min\{b, d\}, \\ \varnothing, & \max\{a, c\} > \min\{b, d\}. \end{cases}$$

Endpoint inclusion follows the endpoint rules of the original intervals.

*Remark* (Unions of intervals). The union of two intervals need not be an interval:

$$(0, 1) \cup (2, 3)$$

has a gap. If two intervals overlap or touch without a missing endpoint, their union is again an interval, for instance

$$(0, 2] \cup [2, 5) = (0, 5).$$

The missing or included endpoints determine whether the joined set is truly one interval.

*Remark* (Complements of intervals). The complement of a bounded interval usually splits into two rays:

$$(a,b)^c = (-\infty, a] \cup [b, \infty), \qquad [a,b]^c = (-\infty, a) \cup (b, \infty).$$

Complements reverse endpoint inclusion: an endpoint excluded from the interval is included in the complement, and an endpoint included in the interval is excluded from the complement.

*Remark* (Differences of intervals). A difference is an intersection with a complement, so it can also split:

$$(0,5) \setminus [2,3] = (0,2) \cup (3,5).$$

If the removed interval only cuts off one side, the result may remain a single interval:

$$(0,5) \setminus (3,\infty) = (0,3].$$

The bracket at 3 appears because $3 \notin (3,\infty)$, so it is not removed.

*Remark* (Topological motivation). Open sets in $\mathbb{R}$ are built from unions of open intervals. Closed sets are understood through complements of open sets. Boundaries arise when a point is not cleanly inside a set or its complement. Thus interval set operations are the grammar behind the metric topology of the real line.

## 12.5 Neighborhoods and Balls

### Balls Are Open Intervals

On the line, the ball of radius $r$ about a point is an open interval centered at that point. Neighborhoods are sets that contain such a ball. These are the local objects from which open sets are assembled.

**Definition (Open Ball on the Real Line)**

**Definition 12.5.1** (Open Ball on the Real Line). Let $x \in \mathbb{R}$ and $r > 0$. The *open ball* of radius $r$ about $x$ is

$$B_r(x) = \{\, y \in \mathbb{R} : |y - x| < r \,\} = (x - r,\, x + r).$$

*Remark* (Standard quantified statement).

$$B_r(x) = \{\, y \in \mathbb{R} : d(y,x) < r \,\} = (x - r, x + r).$$

*Remark* (Interpretation). A ball on the line is just a symmetric open interval: all points strictly within distance $r$ of $x$. Writing balls as intervals keeps the chapter inside the metric topology of $\mathbb{R}$.

*Remark* (Dependencies).

- Distance on the Real Line
- Open Interval

**Definition (Neighborhood of a Point)**

**Definition 12.5.2** (Neighborhood of a Point). A set $N \subseteq \mathbb{R}$ is a *neighborhood* of $x \in \mathbb{R}$ exactly when there exists $r > 0$ with $B_r(x) \subseteq N$.

*Remark* (Standard quantified statement).

$$N \text{ is a neighborhood of } x \iff \exists r > 0\, (B_r(x) \subseteq N).$$

*Remark* (Interpretation). A neighborhood of $x$ is any set with room to spare around $x$ -- it contains a whole ball, not just $x$ itself. Neighborhoods are the language of "near $x$" used to define interior points and limits.

*Remark* (Dependencies).

- Open Ball on the Real Line

## 12.6 Open Sets

### Sets Built from Balls

An open set is one that contains a ball around each of its points. On the line these are exactly the unions of open intervals. The closure properties recorded here are the metric-topology core.

**Definition (Open Set in $\mathbb{R}$)**

**Definition 12.6.1** (Open Set in $\mathbb{R}$). A set $U \subseteq \mathbb{R}$ is *open* exactly when

$$\forall x \in U\ \exists r > 0\, (B_r(x) \subseteq U).$$

*Remark* (Standard quantified statement).

$$U \text{ is open} \iff \forall x \in U\ \exists r > 0\, (B_r(x) \subseteq U).$$

*Remark* (Negated quantified statement).

$$U \text{ is not open} \iff \exists x \in U\ \forall r > 0\, (B_r(x) \not\subseteq U).$$

*Remark* (Failure modes). Openness fails exactly when some point of $U$ has no ball contained in $U$ -- a boundary point trapped inside $U$, every ball around which pokes outside.

*Remark* (Interpretation). An open set has interior room around each of its points: a small step in either direction remains inside the set. The empty set and $\mathbb{R}$ are open, and open intervals are the model examples.

*Remark* (Dependencies).

- Open Ball on the Real Line

**Theorem (Open Intervals Are Open)**

**Theorem 12.6.2** (Open Intervals Are Open). *Every open interval $(a, b)$, every open ray $(a, \infty)$ and $(-\infty, b)$, and $\mathbb{R}$ itself are open sets. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{R}\ (a < b \Longrightarrow (a, b) \text{ is open}),$$
$$\forall a \in \mathbb{R}\ \big((a, \infty) \text{ is open} \wedge (-\infty, a) \text{ is open}\big), \qquad \mathbb{R} \text{ is open.}$$

*Remark* (Interpretation). The name "open interval" is justified: each such interval satisfies the open-set condition, with the radius at $x$ taken as the distance to the nearer endpoint.

*Remark* (Dependencies).

- Open Set in $\mathbb{R}$
- Open Interval

**Theorem (Unions and Finite Intersections of Open Sets)**

**Theorem 12.6.3** (Unions and Finite Intersections of Open Sets). *An arbitrary union of open subsets of $\mathbb{R}$ is open, and a finite intersection of open subsets of $\mathbb{R}$ is open. Go to proof.*

*Remark* (Standard quantified statement).

$$\Big(\forall i \in I\ U_i \text{ open}\Big) \Longrightarrow \Big(\bigcup_{i \in I} U_i\Big) \text{ open;} \qquad \Big(\bigwedge_{k=1}^{n} U_k \text{ open}\Big) \Longrightarrow \Big(\bigcap_{k=1}^{n} U_k\Big) \text{ open.}$$

*Remark* (Interpretation). Openness is preserved by arbitrary unions but only finite intersections. The infinite intersection $\bigcap_n (-1/n, 1/n) = \{0\}$ shows why "finite" cannot be dropped: a shrinking family of open intervals can close down onto a single point.

*Remark* (Dependencies).

- Open Set in $\mathbb{R}$

## 12.7 Closed Sets

### Complements of Open Sets

Closed sets are the complements of open sets. They can equivalently be described as the sets that contain all of their limit points, the description used when reasoning about convergence.

**Definition (Closed Set in $\mathbb{R}$)**

**Definition 12.7.1** (Closed Set in $\mathbb{R}$). A set $F \subseteq \mathbb{R}$ is *closed* exactly when its complement $\mathbb{R} \setminus F$ is open.

*Remark* (Standard quantified statement).

$$F \text{ is closed} \iff (\mathbb{R} \setminus F) \text{ is open.}$$

*Remark* (Interpretation). Closed and open are complementary, not opposite: a set can be both (e.g. $\emptyset$, $\mathbb{R}$) or neither (e.g. $[0,1)$). Closed intervals $[a,b]$ are closed, as are finite sets and $\mathbb{R}$ itself.

*Remark* (Dependencies).

- Open Set in $\mathbb{R}$

**Theorem (Closed Sets Contain Their Limit Points)**

**Theorem 12.7.2** (Closed Sets Contain Their Limit Points). *A set $F \subseteq \mathbb{R}$ is closed if and only if it contains every one of its limit points. Go to proof.*

*Remark* (Standard quantified statement).

$$F \text{ closed} \iff \forall x \in \mathbb{R} \left(x \text{ is a limit point of } F \Rightarrow x \in F\right).$$

*Remark* (Interpretation). This is the convergence-flavored description of closedness: a closed set cannot be approached from outside without already containing the approached point. It is the form used when showing limits of sequences in $F$ stay in $F$.

*Remark* (Dependencies).

- Closed Set in $\mathbb{R}$
- Limit Point

## 12.8 Interior Exterior Boundary

### Inside, Outside, and Edge

Relative to a set $E$, each point of the line is interior to $E$, exterior to $E$, or on its boundary. These three regions partition $\mathbb{R}$ and refine the open/closed distinction.

**Definition (Interior Point)**

**Definition 12.8.1** (Interior Point). Let $E \subseteq \mathbb{R}$ and $x \in \mathbb{R}$. Then $x$ is an interior point of $E$ exactly when

$$\exists r > 0 \, (B_r(x) \subseteq E).$$

*Remark* (Standard quantified statement).

$$x \text{ is an interior point of } E \iff \exists r > 0 \, (B_r(x) \subseteq E).$$

*Remark* (Interpretation). An interior point of $E$ sits inside $E$ with room to spare: a whole ball around it lies in $E$.

*Remark* (Dependencies).

- Open Ball on the Real Line

**Definition (Exterior Point)**

**Definition 12.8.2** (Exterior Point). Let $E \subseteq \mathbb{R}$ and $x \in \mathbb{R}$. Then $x$ is an exterior point of $E$ exactly when

$$\exists r > 0\ (B_r(x) \subseteq \mathbb{R} \setminus E).$$

***Remark*** (Standard quantified statement).

$$x \text{ is an exterior point of } E \iff \exists r > 0\ (B_r(x) \subseteq \mathbb{R} \setminus E).$$

***Remark*** (Interpretation). An exterior point of $E$ is interior to the complement; a whole ball around it misses $E$.

*Remark* (Dependencies).

- Open Ball on the Real Line

**Definition (Boundary Point)**

**Definition 12.8.3** (Boundary Point). Let $E \subseteq \mathbb{R}$ and $x \in \mathbb{R}$. Then $x$ is a boundary point of $E$ exactly when

$$\forall r > 0\ (B_r(x) \cap E \neq \emptyset \ \wedge\ B_r(x) \cap (\mathbb{R} \setminus E) \neq \emptyset).$$

***Remark*** (Standard quantified statement).

$$x \text{ is a boundary point of } E \iff \forall r > 0\ (B_r(x) \cap E \neq \emptyset \ \wedge\ B_r(x) \cap (\mathbb{R} \setminus E) \neq \emptyset).$$

***Remark*** (Interpretation). A boundary point of $E$ is approached from both sides; every ball around it meets both $E$ and its complement.

*Remark* (Dependencies).

- Open Ball on the Real Line

**Definition (Interior of a Set)**

**Definition 12.8.4** (Interior of a Set). The *interior* of $E \subseteq \mathbb{R}$, written $\operatorname{int}(E)$, is the set of all interior points of $E$.

***Remark*** (Standard quantified statement).

$$\operatorname{int}(E) = \{\, x \in \mathbb{R} : \exists r > 0\ (B_r(x) \subseteq E) \,\}.$$

***Remark*** (Interpretation). The interior is the largest open set inside $E$; $E$ is open exactly when $E = \operatorname{int}(E)$. The boundary is what $E$ and its complement share, and a set is closed exactly when it contains its boundary.

*Remark* (Dependencies).

- Interior Point

## 12.9 Limit and Isolated Points

### Approachable and Detached Points

A limit point of $E$ is one that $E$ crowds arbitrarily closely, even with the point itself removed. An isolated point of $E$ belongs to $E$ but has a ball meeting $E$ only at itself. These notions drive the convergence description of closed sets.

**Definition (Limit Point)**

**Definition 12.9.1** (Limit Point). Let $E \subseteq \mathbb{R}$ and $x \in \mathbb{R}$. Then $x$ is a *limit point* of $E$ exactly when

$$\forall r > 0\ \exists y \in E\ \big(0 < |y - x| < r\big).$$

*Remark* (Standard quantified statement).

$$x \text{ is a limit point of } E \iff \forall r > 0\ \exists y \in E\ (0 < |y - x| < r).$$

*Remark* (Negated quantified statement).

$$x \text{ is not a limit point of } E \iff \exists r > 0\ \forall y \in E\ (y = x \vee |y - x| \geq r).$$

*Remark* (Interpretation). Every punctured ball around a limit point meets $E$; the set accumulates at $x$. The strict inequality $0 < |y-x|$ excludes $x$ itself; so $x$ need not belong to $E$ to be a limit point. This is the exact condition under which a sequence in $E$ can converge to $x$.

*Remark* (Dependencies).

- Open Ball on the Real Line

**Definition (Isolated Point)**

**Definition 12.9.2** (Isolated Point). Let $E \subseteq \mathbb{R}$ and $x \in E$. Then $x$ is an *isolated point* of $E$ exactly when

$$\exists r > 0\ \forall y \in E\ \big(|y - x| < r \Rightarrow y = x\big).$$

*Remark* (Standard quantified statement).

$$x \text{ is an isolated point of } E \iff x \in E \wedge \exists r > 0\ \forall y \in E\ \big(|y - x| < r \Rightarrow y = x\big).$$

*Remark* (Interpretation). An isolated point sits in $E$ but stands apart: some ball around it contains no other point of $E$. A point of $E$ is either isolated or a limit point of $E$, never both.

*Remark* (Dependencies).

- Open Ball on the Real Line

## 12.10 Unions and Intersections

### Closure Laws for Open and Closed Sets

The open closure laws of §12.6 dualize to closed sets by complementation. Collecting them gives the bookkeeping rules used throughout the rest of the development.

**Theorem (Closure Laws for Closed Sets)**

**Theorem 12.10.1** (Closure Laws for Closed Sets). *An arbitrary intersection of closed subsets of $\mathbb{R}$ is closed, and a finite union of closed subsets of $\mathbb{R}$ is closed. Go to proof.*

*Remark* (Standard quantified statement).

$$\left(\forall i \in I\ F_i \text{ closed}\right) \Longrightarrow \left(\bigcap_{i\in I} F_i\right) \text{ closed}; \qquad \left(\bigwedge_{k=1}^{n} F_k \text{ closed}\right) \Longrightarrow \left(\bigcup_{k=1}^{n} F_k\right) \text{ closed}.$$

*Remark* (Interpretation). The roles of union and intersection swap relative to open sets: closed sets are stable under arbitrary intersection but only finite union. The asymmetry mirrors the open case under De Morgan's laws.

*Remark* (Dependencies).

- Closed Set in $\mathbb{R}$
- Unions and Finite Intersections of Open Sets

## 12.11 Covers and Subcovers

### Covering a Set with Open Pieces

A cover of $E$ is a family of sets whose union contains $E$; an open cover uses open sets. A subcover keeps enough of the family to still cover, and a finite subcover keeps only finitely many. These are the exact ingredients of compactness, defined so as to survive beyond the line.

**Definition (Open Cover)**

**Definition 12.11.1** (Open Cover). Let $E \subseteq \mathbb{R}$. An *open cover* of $E$ is a family $\{U_i\}_{i\in I}$ of open subsets of $\mathbb{R}$ with

$$E \subseteq \bigcup_{i\in I} U_i.$$

*Remark* (Standard quantified statement).

$$\{U_i\}_{i\in I} \text{ is an open cover of } E \Longleftrightarrow \left(\forall i \in I\ U_i \text{ open}\right) \wedge E \subseteq \textstyle\bigcup_{i\in I} U_i.$$

*Remark* (Interpretation). An open cover is a way of surrounding every point of $E$ with an open piece of the family. The index set $I$ may be infinite; the question compactness asks is whether finitely many pieces already suffice.

*Remark* (Dependencies).

- Open Set in $\mathbb{R}$

**Definition (Subcover and Finite Subcover)**

**Definition 12.11.2** (Subcover and Finite Subcover). Let $\{U_i\}_{i\in I}$ be an open cover of $E$. A *subcover* is a subfamily $\{U_i\}_{i\in J}$ with $J \subseteq I$ that still covers $E$. It is a *finite subcover* when $J$ is finite.

*Remark* (Standard quantified statement).

$$\{U_i\}_{i\in J} \text{ is a finite subcover of } E \iff \left(J \subseteq I \wedge J \text{ is finite} \wedge E \subseteq \bigcup_{i\in J} U_i\right).$$

*Remark* (Interpretation). A subcover discards some pieces while still covering $E$; a finite subcover discards all but finitely many. The existence of a finite subcover for *every* open cover is the defining property of compactness.

*Remark* (Dependencies).

- Open Cover

## 12.12 Compactness on R

### The Open-Cover Definition

Compactness is defined by the open-cover property, deliberately and not as "closed and bounded." The open-cover formulation is the one that survives to general metric spaces and to manifolds such as the torus, where "bounded" has no intrinsic meaning. On $\mathbb{R}$ the two descriptions coincide, and that coincidence is the Heine–Borel theorem of the next section, proved rather than assumed.

**Definition (Compact Set in $\mathbb{R}$)**

**Definition 12.12.1** (Compact Set in $\mathbb{R}$). A set $K \subseteq \mathbb{R}$ is *compact* exactly when every open cover of $K$ admits a finite subcover:

$$\forall \{U_i\}_{i\in I} \text{ open cover of } K\ \exists \text{finite } J \subseteq I \left(K \subseteq \bigcup_{i\in J} U_i\right).$$

*Remark* (Standard quantified statement).

$$K \text{ compact} \iff \forall \{U_i\}_{i\in I}\left(\left(\forall i \in I\ U_i \text{ open}\right) \wedge K \subseteq \bigcup_{i\in I} U_i \Rightarrow \exists \text{finite } J \subseteq I\ K \subseteq \bigcup_{i\in J} U_i\right).$$

*Remark* (Negated quantified statement).

$$K \text{ not compact} \iff \exists \{U_i\}_{i\in I}\left(\left(\forall i \in I\ U_i \text{ open}\right) \wedge K \subseteq \bigcup_{i\in I} U_i \wedge \forall \text{finite } J \subseteq I\ K \not\subseteq \bigcup_{i\in J} U_i\right).$$

*Remark* (Failure modes). Compactness fails exactly when some open cover has no finite subcover. Two standard witnesses: $(0,1]$ fails via the cover $\{(1/n,2)\}_n$ (not closed), and $[0,\infty)$ fails via $\{(-1,n)\}_n$ (not bounded).

*Remark* (Interpretation). Compactness is a finiteness property phrased entirely in open sets: no matter how a set is covered by open pieces, finitely many already do the job. This is what later guarantees, on a compact domain, that continuous functions attain extrema and are uniformly continuous -- the facts Analysis I relies on. Crucially, this definition does not mention order or boundedness, so it transfers unchanged to the torus.

*Remark* (Dependencies).

- Open Cover
- Subcover and Finite Subcover

**Theorem (Compact Subsets of $\mathbb{R}$ Are Closed and Bounded)**

**Theorem 12.12.2** (Compact Subsets of $\mathbb{R}$ Are Closed and Bounded). *If $K \subseteq \mathbb{R}$ is compact, then $K$ is closed and bounded. Go to proof.*

*Remark* (Standard quantified statement).

$$K \text{ compact} \Longrightarrow (K \text{ closed} \wedge K \text{ bounded}).$$

*Remark* (Interpretation). This is the easy half of Heine-Borel and holds the same way in every metric space. Boundedness comes from covering $K$ by growing balls about a fixed point; closedness comes from separating an outside point from $K$ by disjoint balls. The converse -- closed and bounded implies compact -- is special to $\mathbb{R}^n$ and is the next theorem.

*Remark* (Dependencies).

- Compact Set in $\mathbb{R}$
- Closed Set in $\mathbb{R}$
- Bounded Subset of the Real Line

## 12.13 Heine Borel

### The Capstone of the Real Line

Heine–Borel closes the chapter by proving the converse direction: on $\mathbb{R}$, closed and bounded is enough for compactness. The keystone is that every closed bounded interval $[a, b]$ is compact; the general statement follows because a closed subset of a compact set is compact.

**Theorem (Closed Bounded Intervals Are Compact)**

**Theorem 12.13.1** (Closed Bounded Intervals Are Compact). *Every closed bounded interval $[a, b] \subseteq \mathbb{R}$ is compact. Go to proof.*

*Remark* (Standard quantified statement).

$$\forall a, b \in \mathbb{R} \left(a \leq b \Longrightarrow [a, b] \text{ is compact}\right).$$

*Remark* (Interpretation). This is the hard half. The standard proof uses completeness directly: let $S$ be the set of $x \in [a, b]$ such that $[a, x]$ has a finite subcover, and show $\sup S = b$ is attained. Bisection (the nested-interval method) gives the same result. Either route shows completeness is what makes the line compact in the large.

*Remark* (Dependencies).

- Compact Set in $\mathbb{R}$
- Closed Interval
- Least-Upper-Bound Property

**Theorem (Heine–Borel Theorem for $\mathbb{R}$)**

**Theorem 12.13.2** (Heine-Borel Theorem for $\mathbb{R}$). *A set $K \subseteq \mathbb{R}$ is compact if and only if $K$ is closed and bounded. Go to proof.*

*Remark* (Standard quantified statement).

$$K \text{ compact} \iff (K \text{ closed} \wedge K \text{ bounded}).$$

*Remark* (Interpretation). Heine-Borel is the bridge into Analysis I: it certifies that the closed bounded sets -- the ones on which the Extreme Value and uniform-continuity theorems live -- are exactly the compact ones. It is stated here as the equivalence *on* $\mathbb{R}$; the open-cover side is the definition that later carries to the torus, while "closed and bounded" does not.

*Remark* (Exposition). With Heine-Borel in hand the chain is complete: number construction $\to$ embedded number systems $\to$ the real line as an ordered metric space $\to$ Analysis I. Completeness, introduced as the last rung of the embedding ladder, is exactly the property that makes the line compact in the large, and compactness is the hypothesis Analysis I leans on.

*Remark* (Dependencies).

- Compact Subsets of $\mathbb{R}$ Are Closed and Bounded
- Closed Bounded Intervals Are Compact

# Chapter 13

# Formalizing the Number Systems

**formalizing-number-systems**

Structure of the Real Line → **Formalizing the Number Systems** → Lean 4 Formalization

### Why Formalization Is Needed

Ordinary arithmetic is one of the most familiar parts of mathematics. We learn to count, add, subtract, multiply, and divide long before we ask what sort of objects numbers are or why the usual rules are justified. For foundational work, however, intuition is not enough. The basic number systems must be specified by precise axioms and constructions, so that their operations and order relations are not merely assumed but accounted for.

### The Natural-Number Structure

The first place where this becomes visible is the natural numbers. A foundational treatment cannot simply point to the counting numbers and begin calculating. It has to identify a first element, a successor operation, and the induction principle that makes recursive definitions and proofs possible. Peano's role was to isolate this structure: a first element, a successor map, conditions saying that successor is injective and never returns to the first element, and an induction principle strong enough to characterize the system. This is the role of Peano's 1889 presentation of arithmetic [Peano, 1889].

### Structural Characterization

Dedekind's work emphasized the same structural point of view. What matters is not the particular names attached to the elements, but the pattern determined by a distinguished first element, successor, recursion, and induction. This perspective underlies the treatment of Peano systems earlier in the volume, and Dedekind's name also sits behind the cut-based construction of the real numbers developed later in the usual foundational tradition [Dedekind, 1901].

### Later Number Systems

Once the natural numbers have been fixed, the later systems require their own explicit constructions. The integers are built so that subtraction becomes an internal operation. The rational numbers are built so that division by nonzero quantities is represented inside the system. The real numbers are then introduced to address the completeness failures of the rationals. In each case, the construction supplies the operations, the order, and the structural properties needed for analysis, rather than treating them as background facts.

# Chapter 14

# Lean 4 Formalization

**lean**

Formalizing the Number Systems → **Lean 4 Formalization** → real-analysis

## 14.1 A First Lean 4 Reading

### Subsets as Predicates

Lean represents many mathematical objects by their defining behavior. For a type $P$, a subset of $P$ can be read as a predicate $P \to Prop$: it takes an element of $P$ as input and returns the proposition that the element belongs to the subset. This is the key syntax bridge for the Peano-system arguments that use inductive subsets.

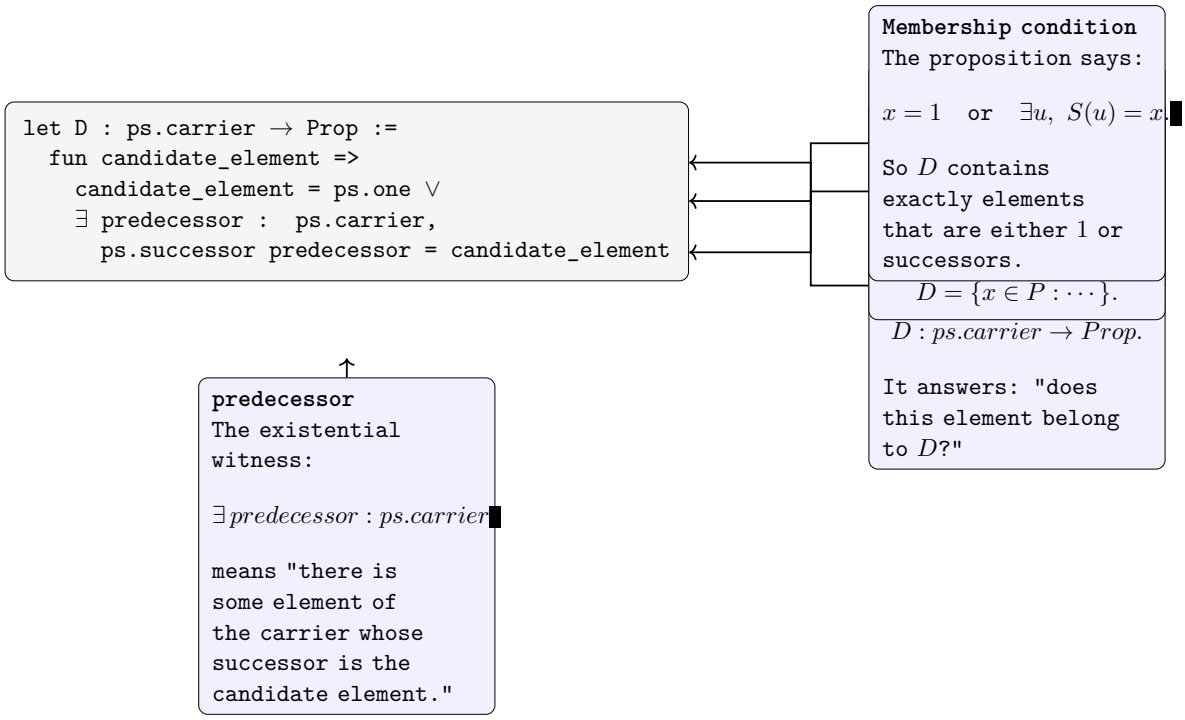



Figure 14.1: Reading a Lean subset predicate as a mathematical subset condition.

*Remark* (Interpretation). The Lean line ‘D : ps.carrier → Prop‘ says that membership in $D$ is a proposition depending on an input from the carrier of the Peano system. In ordinary mathematical notation, this is the same role played by a subset $D \subseteq P$ or by a defining condition $D(x)$.

### Reading an Initial Proof

Lean proofs are built from local goals and the terms that solve them. A first reading strategy is to identify the current hypotheses, the target proposition, and the exact line that transforms the local context into the required result.

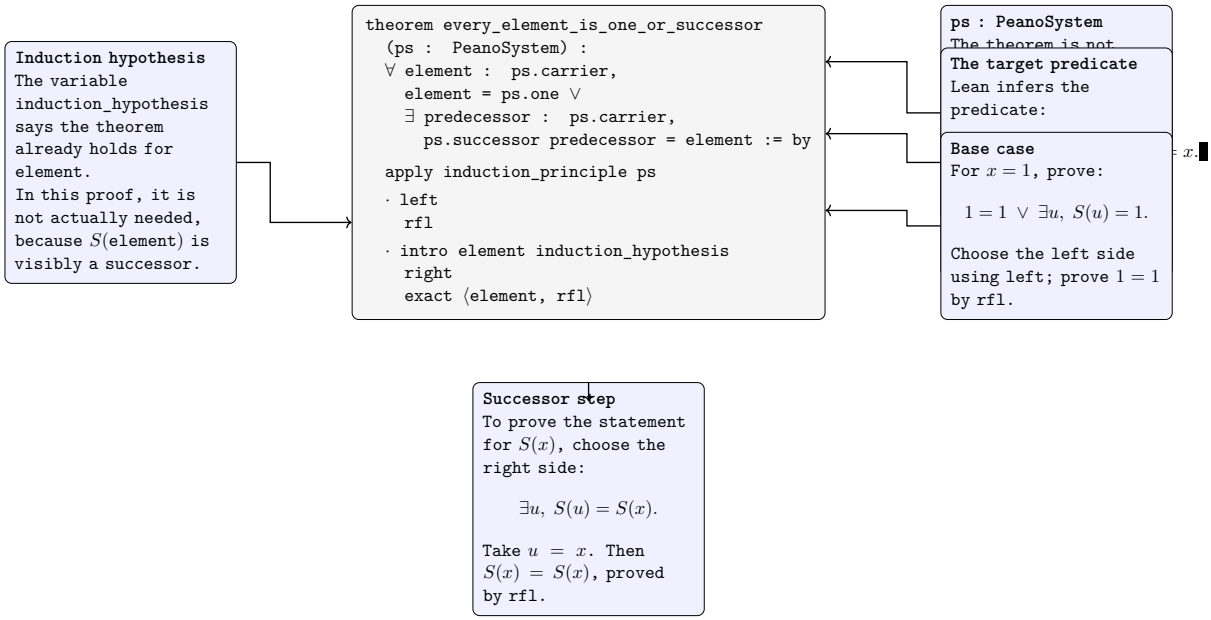



Figure 14.2: Reading the shape of an initial Lean proof.

*Remark* (Interpretation). The figure isolates the proof-state reading habit used throughout Lean: separate the available assumptions from the current goal, then read each proof term as an instruction that closes or transforms that goal.

# Bibliography


Richard Dedekind. *Essays on the Theory of Numbers*. Open Court, Chicago, 1901. URL https://www.gutenberg.org/ebooks/21016. Project Gutenberg eBook #21016; contains *Continuity and Irrational Numbers* and *The Nature and Meaning of Numbers*.

Solomon Feferman. *The Number Systems: Foundations of Algebra and Analysis*. Addison-Wesley Publishing Company, Reading, MA, 1964. URL https://archive.org/details/numbersystemsfou0000fefe. Internet Archive scan.

Ronald C. Freiwald. Peano systems and the whole number system. Washington University in St. Louis course notes, 2009. URL https://www.math.wustl.edu/~freiwald/310peanosys.pdf.

Karel Hrbacek and Thomas Jech. *Introduction to Set Theory*. Marcel Dekker, New York, 3 edition, 1999. ISBN 0824779150.

F. William Lawvere. Adjointness in foundations. *Dialectica*, 23(3–4):281–296, 1969. doi: 10.1111/j.1746-8361.1969.tb01194.x.

Elliott Mendelson. *Number Systems and the Foundations of Analysis*. Robert E. Krieger Publishing Company, Malabar, FL, 1982. Reprint of the 1973 Academic Press edition.

Giuseppe Peano. *Arithmetices principia, nova methodo exposita*. Fratres Bocca, Turin, 1889. Latin title commonly translated as *The Principles of Arithmetic, Presented by a New Method*.

Stephen G. Simpson. *Subsystems of Second Order Arithmetic*. Perspectives in Logic. Cambridge University Press, Cambridge, 2 edition, 2009.

H. A. Thurston. *The Number System*. Blackie and Son, London and Glasgow, 1956.