CAS 701

Logic and Discrete Mathematics

Fall 2017

1 Mathematical Values and Structures

William M. Farmer

Department of Computing and Software
McMaster University

September 11, 2017

Outline

1. What is mathematics?
2. Mathematical values.
    a. Numbers.
    b. Sets.
    c. Sequences.
    d. Relations.
    e. Functions.
3. Mathematical structures.
    a. Orders.
    b. Algebraic structures.
    c. Lattices.
    d. Graphs.
    e. Trees.

What is Mathematics?

- Mathematics is a process for understanding the mathematical aspects of the world involving such things as time, space, measure, pattern, and logical consequence.
- The process consists of the following intertwined activities:

  1. The creation of mathematical models representing mathematical aspects of the world.
  2. The exploration of the models by;

     a. Stating and proving conjectures.
     b. Performing computations.
     c. Creating and studying visual representations.
     d. Studying examples of the models.

  3. The organization and interconnection of the models.
  4. The presentation of the models in a narrative form.

Mathematical Models

- There are many kinds of mathematical models.
- Mathematical values and mathematical structures are the building blocks of mathematical models!

Mathematical Proof

- In mathematics, a proof is a deductive argument that shows a conclusion follows from a set of premises.
- Proof is a central component of the mathematics process which is unique to mathematics.
- Proofs are used for several purposes:
  1. Communicating mathematical ideas.
  2. Certifying that mathematical results are correct.
  3. Discovering new mathematical facts.
  4. Learning mathematics.
  5. Establishing connections between mathematical ideas.
  6. Showing the universality of mathematical results.
  7. Creating mathematical beauty.

Formal Proofs

- A formal proof is a derivation in a proof system for a formal logic.
- A formal proof can be presented in two ways:
  - As a description of the actual derivation.
  - As a prescription for creating the derivation.
- Software systems can be used to interactively develop and mechanically check formal proofs.
- The writer is highly constrained by the logic, the proof system, and the fact that every detail must be verified.
  - So the meaning of the theorem and the key ideas of proof may not be readily apparent.
- But there is a very high assurance that the theorem is correct.

What is a Number System?

- A number system consists of:
  1. A set of mathematical values called numbers.
  2. A set of functions on the numbers called arithmetic operations.
- The arithmetic operations include addition ($+$) and multiplication ($*$) and may also include other operations such as subtraction, division, and exponentiation.
- A number system is not a numeral system like the Hindu-Arabic or Roman numeral systems.

Common Number Systems

$\mathbb{N} = \{0, 1, 2, \ldots\}$, the natural numbers, for counting and ordering.

$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$, the integers, for counting forwards and backwards.

$\mathbb{Q}$, the rational numbers, for measuring.

$\mathbb{R}$, the real numbers, for solving geometric problems.

$\mathbb{C}$, the complex numbers, for solving algebraic problems.

$\mathbb{Z}_n$, the modular integers, for integer arithmetic modulo n (clock arithmetic).

$\mathbb{N} \subseteq \mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R} \subseteq \mathbb{C}$.

$\mathbb{Z}_n \subseteq \mathbb{N}$.

Other Notable Number Systems

$\mathbb{O}$, the ordinals, for ordering infinite sets.

${}^*\mathbb{R}$, the hyperreal numbers, for calculating limits in calculus.

$\mathbb{H}$, the quaternions, for calculating in three-dimensional space.

$\mathbb{S}$, the surreal numbers, for "total happiness".

$\mathbb{N} \subseteq \mathbb{O} \subseteq \mathbb{S}$.

$\mathbb{R} \subseteq {}^*\mathbb{R} \subseteq \mathbb{S}$.

$\mathbb{C} \subseteq \mathbb{H}$.

Table of Number Systems

Number System	Algebraic Classification
$\mathbb{N}$	semiring
$\mathbb{Z}$	ring
$\mathbb{Q}$	ordered field
$\mathbb{R}$	complete ordered field
$\mathbb{C}$	algebraically closed field
$\mathbb{Z}_n$	ring
$\mathbb{Z}_p$, p prime	field
$\mathbb{O}$	near-semiring
${}^*\mathbb{R}$	ordered field
$\mathbb{H}$	division ring
$\mathbb{S}$	ordered field

Computer Number Systems

- Fixed precision integers.
  - Finite set of integers represented using 2^n bits.
  - Representation scheme is usually two's complement.
  - Usually implemented as $\mathbb{Z}_{2^n}$.
- Floating point numbers.
  - Finite set of rational numbers represented in scientific notation $\pm a * 2^b$ using 2^n bits.
- Arbitrary precision integers.
  - Every integer is representable.
  - Know as "bignums".
- Arbitrary precision rational numbers.
  - Every rational number is represented by two bignums.

Sets

- A set is a collection of objects.
- A set can be a member of a set.
- Some very large collections of objects cannot be sets.
  - For example, consider the the Russell set, the set of all sets that do not contain themselves.
  - A collection that is too large to be a set is called a proper class.
- Styles of set theories:
  - Naive set theory.
  - Having a universal set.
  - Having no universal set (e.g., ZF set theory).
  - Having a universal class (e.g., NBG set theory).

Set Concepts

- Basic properties: membership, subset, cardinality.
- Basic operations:
  - Union, intersection, complement, difference, symmetric difference.
  - Cartesian product (product), disjoint union (sum).
  - Sum set, power set.
- Special sets: the emptyset, universal sets, ordered pairs, sequences, relations, functions, ordinals, cardinals.
- Relations and functions can be represented as special kinds of sets (e.g., as sets of tuples).

Sequences

- A sequence is an ordered set.
  - Alternate definition: A sequence is a mapping of an initial segment of the natural numbers to a set.
  - Sequences can be either finite or infinite.
  - A finite sequence $a_1, \ldots, a_n$ is usually written as $(a_1, \ldots, a_n)$, $\langle a_1, \ldots, a_n \rangle$, or $[a_1, \ldots, a_n]$.
- Finite sequences are used in many ways in mathematics and computing:
  - A tuple is a finite sequence, usually written as $(a_1, \ldots, a_n)$, whose members may be of different kinds.
  - A list is a finite sequence, usually written as $[a_1, \ldots, a_n]$, whose members are of the same kind.
  - A string is a finite sequence of characters, usually written as "$a_1 \cdots a_n$".

Relations

- For $n \geq 1$, an *n-ary relation* is a set $R \subseteq A_1 \times \cdots \times A_n$ $(n \geq 1)$.
  - Any set can be considered as a unary relation.
  - Any nonunary relation can be considered as a binary relation.
- Functions are considered as special relations.
  - An n-ary function $f : A_1, \ldots, A_n \rightarrow B$ is identified with the corresponding $(n+1)$-ary relation $R_f \subseteq A_1 \times \cdots \times A_n \times B$ called the graph of the function.
- An n-ary relation can be represented by an n-ary predicate.

Relation Concepts

- Basic binary relation properties:
  - Reflexive, symmetric, transitive.
- Basic binary relation operations:
  - Domain, range.
  - Composition, inverse.
- Special relations: the empty relation, universal relations, equivalence relations.
- Ways of representing relations:
  - Using zero-one matrices (for binary relations).
  - Using directed graphs (for homogeneous binary relations).

Closures of Relations

- Reflexive closure.
- Symmetric closure.
- Transitive closure.
  - Equals the connectivity relation.

Equivalence Relations

- A binary relation on a set is an equivalence relation if it is reflexive, symmetric, and transitive.
- Given an equivalence relation R on a set S, the equivalence class of $a \in S$ is the set

$$\{b \in S \mid a \; R \; b\}.$$

- Theorem.
    1. The equivalence classes of an equivalence relation on a set S form a partition of S.
    2. Given a partition of a set S, there is an equivalence relation on S whose equivalence classes are the members of the partition.

(Unary) Functions

- Definition 1: A function is a rule $f : I \to O$ that associates members of I (inputs) with members of O (outputs).
  - Every input is associated with at most one output.
  - Some inputs may not be associated with an output.
    Example: $f : \mathbb{Z} \to \mathbb{Q}$ where $x \mapsto 1/x$.
- Definition 2: A function is a set $f \subseteq I \times O$ such that if $(x, y), (x, y') \in f$, then $y = y'$.
- Each function f has a domain $D \subseteq I$ and a range $R \subseteq O$.
  - f is total if $D = I$ and partial if $D \subset I$.
- A set or relation can be represented as a special kind of function (e.g., as a predicate, a characteristic function, or an indicator).

Lambda Notation

- Lambda notation is a precise, convenient way to specify functions.
- If B is an expression of type β,

$$\lambda\, x : \alpha \,.\, B$$

  denotes a function $f : \alpha \rightarrow \beta$ such that $f(a) = B[x \mapsto a]$.
- Example: Let $f = \lambda\, x : \mathbb{R} \,.\, x * x$.
  - $f(2) = (\lambda\, x : \mathbb{R} \,.\, x * x)(2) = 2 * 2$.
  - f denotes the squaring function.
- Lambda notation is used in many languages to express ideas about functions.
- Examples:
  - Lambda Calculus (a model of computability).
  - Simple Type Theory (a higher-order predicate logic).
  - Lisp (a functional programming language).

n-Ary Functions

- Definition 1: For $n \geq 0$, an n-ary function is a rule $f : I_1, \ldots, I_n \rightarrow O$ that associates members of $I_1, \ldots, I_n$ (inputs) with members of O (outputs).
  - Every list of inputs is associated with at most one output.
  - Some lists of inputs may not be associated with an output.
- Definition 2: For $n \geq 0$, an n-ary function is a set $f \subseteq I_1 \times \cdots \times I_n \times O$ such that if $(x_1, \ldots, x_n, y), (x_1, \ldots, x_n, y') \in f$, then $y = y'$.
- Each function f has a domain $D \subseteq I_1 \times \cdots \times I_n$ and a range $R \subseteq O$.

Representing n-Ary Functions as Unary Functions

There are two ways of representing a n-ary function as a unary function:

1. As a function of tuples: $f : I_1, \ldots, I_n \rightarrow O$ is represented as

   $$f' : I_1 \times \cdots \times I_n \rightarrow O$$

   where

   $$f(x_1, \ldots, x_n) = f'((x_1, \ldots, x_n)).$$

2. As a curryed function: $f : I_1, \ldots, I_n \rightarrow O$ is represented as

   $$f'' : I_1 \rightarrow (I_2 \rightarrow (\cdots (I_n \rightarrow O) \cdots))$$

   where

   $$f(x_1, \ldots, x_n) = f''(x_1) \cdots (x_n).$$

Example

- Let $f = \lambda\, x, y : \mathbb{R} \,.\, x^2 + y^2$.
- $f' = \lambda\, p : \mathbb{R} \times \mathbb{R} \,.\, [\mathsf{fst}(p)]^2 + [\mathsf{snd}(p)]^2$.

  $$\begin{aligned} f'((a,b)) &= (\lambda\, p : \mathbb{R} \times \mathbb{R} \,.\, [\mathsf{fst}(p)]^2 + [\mathsf{snd}(p)]^2)((a,b)) \\ &= [\mathsf{fst}((a,b))]^2 + [\mathsf{snd}((a,b))]^2 \\ &= a^2 + b^2. \end{aligned}$$
- $f'' = \lambda\, x : \mathbb{R} \,.\, \lambda\, y : \mathbb{R} \,.\, x^2 + y^2$.

  $$\begin{aligned} f''(a)(b) &= (\lambda\, x : \mathbb{R} \,.\, \lambda\, y : \mathbb{R} \,.\, x^2 + y^2)(a)(b) \\ &= (\lambda\, y : \mathbb{R} \,.\, a^2 + y^2)(b) \\ &= a^2 + b^2. \end{aligned}$$

Function Concepts

- Basic properties:
  - Arity (0-ary, unary, n-ary with $n \geq 2$, flexary).
  - Total, injective, surjective, bijective.
  - Image, inverse image.
- Basic operations: composition, restriction, inverse.
- Special functions: the empty function, identity functions, choice functions.

Cardinality

- Two sets A and B are equipollent, written $A \approx B$, if there is a bijection $f : A \to B$ between them.
- $A \preceq B$ means $A \approx B'$ for some $B' \subseteq B$.
- A set is infinite if it is equipollent with a proper subset of itself.
- The cardinality of a set A is the cardinal number c such that A and c are equipollent.
- Theorem.
  1. $\mathbb{N} \approx \mathbb{Q}$.
  2. (Cantor) $\mathbb{N} \not\approx \mathbb{R}$.
- Theorem (Schröder-Bernstein). If $A \preceq B$ and $B \preceq A$, then $A \approx B$.

Mathematical Structures

- Loosely speaking, a mathematical structure is a set of mathematical values that are structured in some manner.
- A typical mathematical structure consists of:
  1. A finite set of nonempty domains (sets) of values: $D_1, D_2, \ldots, D_n$.
  2. A set of distinguished values in the domains: $a_1, a_2, \ldots$
  3. A set of functions whose inputs and outputs are in the domains: $f_1, f_2, \ldots$
  4. A set of relations over the domains: $R_1, R_2, \ldots$
- Such a mathematical structure may be written as a tuple:

  $$(D_1, D_2, \ldots, D_n; a_1, a_2, \ldots; f_1, f_2, \ldots; R_1, R_2, \ldots).$$

  The semicolons may be dropped for the meaning is clear.
- Example: The real numbers as a complete ordered field:

  $$(\mathbb{R}; 0, 1; +, -, *, {}^{-1}; =, <).$$

Examples of Mathematical Structures

- Number systems.
- Orders.
- Algebraic structures.
- Lattices.
- Graphs.
- Trees.
- Abstract data types (ADTs) used in computing (e.g, ADTs for strings, lists, streams, arrays, records, stacks, queues).

Orders

- An order is a mathematical structure of the form

  $$(D; ; ; R)$$

  where R is binary relation that orders D is some manner.
- The order is weak [strict] if

  $$a \; R \; a \quad [\neg(a \; R \; a)]$$

  holds for all $a \in D$.

Pre-Orders

- A pre-order is a mathematical structure $(S, \leq)$ where $\leq$ is a binary relation on S that is:
  - Reflexive: $\forall x \mathrel{.} x \leq x$.
  - Transitive: $\forall x, y, z \mathrel{.} x \leq y \wedge y \leq z \Rightarrow x \leq z$.
- Example: $(F, \Rightarrow)$ is a pre-order where F is a set of formulas and $\Rightarrow$ is implication.
- A pre-order can have cycles.
- Every binary relation R on a set S can be extended to a pre-order on S by taking the reflexive and transitive closure of R.

Partial Orders

- A weak partial order is a mathematical structure $(S, \leq)$ where $\leq$ is a binary relation on S that is:
  - Reflexive: $\forall x \,.\, x \leq x$.
  - Antisymmetric: $\forall x, y \,.\, x \leq y \wedge y \leq x \Rightarrow x = y$.
  - Transitive: $\forall x, y, z \,.\, x \leq y \wedge y \leq z \Rightarrow x \leq z$.
- A strict partial order is a mathematical structure $(S, <)$ where $<$ is a binary relation on S that is:
  - Irreflexive: $\forall x \,.\, \neg(x < x)$.
  - Asymmetric: $\forall x, y \,.\, x < y \Rightarrow \neg(y < x)$.
  - Transitive: $\forall x, y, z \,.\, x < y \wedge y < z \Rightarrow x < z$.
- Examples: $(\mathcal{P}(S), \subseteq)$ and $(\mathcal{P}(S), \subset)$ are weak and strict partial orders.
- A partial order does not have cycles.
- Every pre-order can be interpreted as a partial order.

Total Orders

- A weak total order is a mathematical structure $(S, \leq)$ where $\leq$ is a binary relation on S that is:
  - Antisymmetric: $\forall x, y \,.\, x \leq y \wedge y \leq x \Rightarrow x = y$.
  - Transitive: $\forall x, y, z \,.\, x \leq y \wedge y \leq z \Rightarrow x \leq z$.
  - Total: $\forall x, y \,.\, x \leq y \vee y \leq x$.
- A strict total order is a mathematical structure $(S, <)$ where $<$ is a binary relation on S that is:
  - Irreflexive: $\forall x \,.\, \neg(x < x)$.
  - Asymmetric: $\forall x, y \,.\, x < y \Rightarrow \neg(y < x)$.
  - Transitive: $\forall x, y, z \,.\, x < y \wedge y < z \Rightarrow x < z$.
  - Trichotomous: $\forall x, y \,.\, x < y \vee y < x \vee x = y$.

  Examples: $(\mathbb{N}, \leq)$, $(\mathbb{Z}, \leq)$, $(\mathbb{Q}, \leq)$, and $(\mathbb{R}, \leq)$ are weak total orders.

Some Basic Order Definitions

- Let $(P, \leq)$ be a partial order and $S \subseteq P$.
- A maximal element [minimal element] of S is a $M \in S$ [$m \in S$] such that $\neg(M < x)$ [$\neg(x < m)$] for all $x \in S$.
- The maximum element or greatest element [minimum element or least element] of S, if it exists, is a $M \in S$ [$m \in S$] such that $x \leq M$ [$m \leq x$] for all $x \in S$.
- An upper bound [lower bound] of S is a $u \in P$ [$l \in P$] such that $x \leq u$ [$l \leq x$] for all $x \in S$.
- The least upper bound or supremum [greatest lower bound or infimum of S, if it exists, is a $U \in P$ [$L \in P$] such that U is an upper bound of S and, if u is an upper bound of S, then $U \leq u$ [L is a lower bound of S and, if l is a lower bound of L, then $l \leq L$].
- A function $f : P \to P$ is monotone with respect to $\leq$ if, for all $a, b \in P$, $a \leq b$ implies $f(a) \leq f(b)$.

Well-Orders

- A well-order is a mathematical structure $(S, \leq)$ that is a weak total order such that every nonempty subset of S has a minimum element with respect to $\leq$.
- Examples: $(\mathbb{N}, \leq)$ and $(\mathbb{O}, \leq)$ are well-orders.
- A well-order has no infinite strictly decreasing sequences.
- The proof technique of induction and definition technique of recursion can be applied with respect to a well-ordered set.

Algebraic Structures

- An algebraic structure is a mathematical structure of the form

$$(D_1, D_2, \ldots, D_n; a_1, a_2, \ldots; f_1, f_2, \ldots;)$$

  (that contains no relations).
- There is a huge number of different kinds of algebraic structures.

Examples of Algebraic Structures

- Having a single binary operation:
  - Magma: $(D; ; \mathrm{mul};)$.
  - Semigroup: $(D; ; \mathrm{mul};)$.
  - Monoid: $(D; e; \mathrm{mul};)$.
  - Group: $(D; e; \mathrm{mul}, \mathrm{inv};)$.
- Having $+$ and $*$ operations:
  - Semiring: $(R; 0, 1; +, *;)$.
  - Ring: $(R; 0, 1; +, -, *;)$.
  - Division ring: $(R; 0, 1; +, -, *, ^{-1};)$.
  - Field: $(R; 0, 1; +, -, *, ^{-1};)$.
- Having $+$ and scalar multiplication:
  - Module over a ring R: $(M, R; \cdots)$.
  - Vector space over a field F: $(V, F; \cdots)$.

Boolean Algebras

- A boolean algebra is an algebraic structure

  $$(B; 0, 1; +, *, ^{-};)$$

  that satisfies the axioms (i.e., properties) given below. $+$ and $*$ are binary functions and $^{-}$ is a unary function.
- Boolean algebra was first presented by the logician George Boole (1815-1864) in 1847.
- There are infinitely many nonisomorphic boolean algebras.
- Boolean algebra is is used to model electronic circuits.
- Examples:

  1. $(\{\mathrm{T}, \mathrm{F}\}, \mathrm{F}, \mathrm{T}, \vee, \wedge, \neg)$.
  2. $(\mathcal{P}(U), \emptyset, U, \cup, \cap, ^{-})$ where U is any set.

The Axioms of a Boolean Algebra

Associativity Laws

$$\forall x, y, z \,.\, (x + y) + z = x + (y + z)$$
$$\forall x, y, z \,.\, (x * y) * z = x * (y * z)$$

Commutativity Laws

$$\forall x, y \,.\, x + y = y + x \qquad \forall x, y \,.\, x * y = y * x$$

Distributive Laws

$$\forall x, y, z \,.\, x + (y * z) = (x + y) * (x + z)$$
$$\forall x, y, z \,.\, x * (y + z) = (x * y) + (x * z)$$

Identity Laws

$$\forall x \,.\, x + 0 = x \qquad \forall x \,.\, x * 1 = x$$

Complement Laws

$$\forall x \,.\, x + \overline{x} = 1 \qquad \forall x \,.\, x * \overline{x} = 0$$

Some Theorems of a Boolean Algebra

Idempotent Laws

$$\forall x \,.\, x + x = x \qquad \forall x \,.\, x * x = x$$

Absorption Laws

$$\forall x, y \,.\, x + (x * y) = x \qquad \forall x, y \,.\, x * (x + y) = x$$

De Morgan Laws

$$\forall x, y \,.\, \overline{x + y} = \overline{x} * \overline{y}$$
$$\forall x, y \,.\, \overline{x * y} = \overline{x} + \overline{y}$$

Laws of Zero and One

$$\forall x \,.\, x + 1 = 1 \qquad \forall x, y \,.\, x * 0 = 0$$
$$\overline{0} = 1 \qquad \overline{1} = 0$$

Law of Double Complement

$$\forall x \,.\, \overline{\overline{x}} = x$$

Lattices

- A lattice is partial order $(L, \leq)$ such that:
  1. Every pair a, b of elements of L has a least upper bound in L called the join of a and b (joins exist).
  2. Every pair a, b of elements of L has a greatest lower bound in L called the meet of a and b (meets exist).
- The minimum and maximum of a lattice, if they exist, are called the bottom denoted by 0 or $\perp$ and the top denoted by 1 or $\top$, respectively.
- Examples:
  1. $(\mathcal{P}(S), \subseteq)$ is a lattice with a bottom and top.
  2. $(\mathbb{N}, \leq)$ is a lattice with a bottom but no top.
  3. $(\mathbb{N}, |)$, where $a \mid b$ means a divides b, is a lattice with a bottom and top.
  4. If $(B, 0, 1, +, *, ^{-})$ is a boolean algebra, then $(B, \leq)$ is a complemented distributive lattice with a top and bottom where $a \leq b$ means $a = a * b \wedge a + b = b$.

Semilattices

- A semilattice is partial order $(S, \leq)$ such that either joins exist or meets exist.
  - It is a join-semilattice if joins exist.
  - It is a meet-semilattice if meets exist.
- Examples:
  - Any lattice is a semilattice.
  - Any tree can be viewed as a semilattice.

Lattices as Algebraic Structures

- A lattice can also be defined as an algebraic structure

  $$(L; ; \vee, \wedge;)$$

  that satisfies the axioms given below.
- $\vee$ and $\wedge$ denotes the join and meet of the lattice, resp.
- If $(L, \vee, \wedge)$ are lattices, the $(L, \vee)$ and $(L, \wedge)$ are semilattices.
- A bounded lattice is an algebraic structure

  $$(L; 0, 1; \vee, \wedge;)$$

  such that $(L, \vee, \wedge)$ is a lattice, 0 is the identity element for $\vee$, and 1 is the identity element for $\wedge$.

The Axioms of a Lattice

Associativity Laws

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

Commutativity Laws

$$\forall x, y \,.\, x \vee y = y \vee x$$
$$\forall x, y \,.\, x \wedge y = y \wedge x$$

Absorption Laws

$$\forall x, y \,.\, x \vee (x \wedge y) = x$$
$$\forall x, y \,.\, x \wedge (x \vee y) = x$$

Notice that the absorption laws imply:

Idempotent Laws

$$\forall x \,.\, x \vee x = x$$
$$\forall x \,.\, x \wedge x = x$$

Complete Lattices

- A complete lattice is a partial order $(L, \leq)$ such that, for each $S \subseteq L$, S has a least upper bound and greatest lower bound in L.
- Examples:
  - $(\mathcal{P}(S), \subseteq)$ is a complete lattice.
  - $(\mathbb{R}(0,1), \leq)$ is not a complete lattice.
  - $(\mathbb{R}[0,1], \leq)$ is a complete lattice.
  - $(\mathbb{Q}[0,1], \leq)$ is not a complete lattice.
  - $(\mathbb{N}, \leq)$ is not a complete lattice.
  - $(\mathbb{N}, |)$ is a complete lattice.

Knaster-Tarski Fixed Point Theorem

- Theorem. Let $(L, \leq)$ be a complete lattice and $f : L \to L$ be monotone with respect to $\leq$. Then there exists a fixed point of f, i.e., there exists an $a \in L$ such that $f(a) = a$. Moreover, $(F, \leq)$, where F is the set of fixed points of f, is a complete lattice.
- There are several other fixed point theorems related to the Knaster-Tarski theorem.
- Fixed point theorems can be used to define objects by recursion.

Graphs

- A graph is a mathematical structure of the form

  $$(V, E; ; ; R)$$

  where $R \subseteq E \times V \times V$ and, for all $e \in E$, there are $v_1, v_2 \in V$ such that $R(e, v_1, v_2)$.
- The members of V and E are called the vertices and edges, respectively, of the graph.
- An edge $e \in E$ is undirected [directed] if $R(e, v_1, v_2)$ implies $R(e, v_2, v_1)$ [$\neg R(e, v_2, v_1)$].
- A graph (V, E, R) is undirected [directed] if every $e \in E$ is directed [undirected].
- A graph (V, E, R) is simple if $R(e_1, v_1, v_2)$ and $R(e_2, v_1, v_2)$ imply $e_1 = e_2$.

Connectedness

- Let $G = (V, E, R)$ be a graph and $v, v' \in V$.
- A path from v to v' in G is a finite sequence $\langle u_0, \ldots, u_n \rangle$ of vertices in V with $n \geq 1$ such that $v = u_0$, $v' = u_n$, and, for all i with $0 \leq i < n$, there is some $e_i \in E$, such that $R(e_i, u_i, u_{i+1})$.
- v is connected to v' if there is a path from v to v' in G.
- G is connected if each pair of vertices in V is connected.
- A cycle in G is a path from a vertex to itself in G.
- G is cyclic if there is a cycle in G.

Trees

- A tree is a connected, acyclic, undirected, simple graph.
- A directed tree is an acyclic directed graph (V, E, R) with a vertex $v \in V$ (called the root) such that there is a unique path from v to each other vertex in V.