CAS 701 Fall 2005

03 Numbers, Sets, Functions, And Relations

Instructor: W. M. Farmer

Revised: 23 October 2005

Number Systems

- **N**, the natural numbers.
- **O**, the ordinal numbers.
- **Z**, the integers.
- $\mathbf{Z}_n$, the integers modulo n.
- **Q**, the rational numbers.
- **R**, the real numbers.
- **H**, the hyperreal numbers.
- **S**, the surreal numbers.
- **C**, the complex numbers.

Foundational Mathematical Objects

- The three most common kinds of foundational objects:

  1. Sets.
  2. Functions.
  3. Relations.

- Each kind of object can be used to represent the other two kinds of objects.

Sets

- A **set** is a collection of objects.

- 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.
  - A universal set.
  - No universal set (e.g., **ZF set theory**).
  - 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).
  - Power set.
- Special sets: the emptyset, universal sets, functions, relations, ordinals, cardinals.
- Functions and relations can be represented as special kinds of sets (e.g., as sets of **tuples**).

Functions

- There are two definitions of a function:
  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 : **Z** $\to$ **Q** where $x \mapsto 1/x$.
  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** $R \subseteq O$.

- A set or relation can be represented as a special kind of function (e.g., as a **predicate**, a **characteristic function**, or an **indicator**).

Function Concepts

- Basic properties:
  - arity (0, 1, $n \geq 2$, multiary).
  - 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 $f : A \rightarrow B$ 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. $\mathbf{N} \approx \mathbf{Q}$.
  2. **(Cantor)** $\mathbf{N} \not\approx \mathbf{R}$.

- **Theorem (Schröder-Bernstein)**. If $A \preceq B$ and $B \preceq A$, then $A \approx B$.

Relations

- 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 $f : A_1 \times \cdots \times A_n \to B$ 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 relation properties:
  - Reflexive, symmetric, transitive.
- Basic relation operations:
  - Domain, range.
  - Composition, inverse.
- Special relations: the empty relation, universal relations, equivalence relations.
- Ways of representing relations:
  - Using zero-one matrices.
  - Using directed graphs.

Closures of Relations

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

Equivalence Relations

- A 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.

Orderings

- A **pre-order** is a relation R on a set that is reflexive and transitive.

- A **weak partial order** is a relation $\leq$ on a set that is reflexive, antisymmetric, and transitive.

- A **strict partial order** is a relation $<$ on a set that is irreflexive, asymmetric, and transitive.

- A **weak total order** is a relation $\leq$ on a set that is antisymmetric, transitive, and total.

- A **strict total order** is a strict partial order $<$ that satisfies the trichotomy law.

- A **well order** is a total order such that every nonempty set of elements contains a least element with respect to the ordering.

Algebras

- A **mathematical structure** is a structured collection of sets, functions, and relations.

- An **algebra** is a mathematical structure consisting of:

  1. A set of elements called the **domain**.
  2. A set of distinguished elements, functions, and relations called the **signature** that impose a structure on the domain of elements.

- The signature usually determines a **language** for describing and making assertions about the elements of the domain.

- Algebras are often described by tuples of the form

$$(D, e_1, \ldots, e_k, f_1, \ldots, f_m, r_1, \ldots . r_n).$$

Examples of Algebras

- Number systems (e.g, natural, ordinal, integer, rational, real, hyperreal, surreal, complex).

- Algebraic structures (e.g, monoids, groups, rings, fields).

- Orders (e.g., pre, partial, total, well).

- Lattices and boolean algebras.

- Graphs and trees.

- Abstract data types (ADTs) used in Computer Science (e.g, ADTs for strings, lists, streams, arrays, records, stacks, queues).