Discrete Structures for Computer Science

George Voutsadakis[1]

[1]Mathematics and Computer Science
Lake Superior State University

LSSU CSci 341

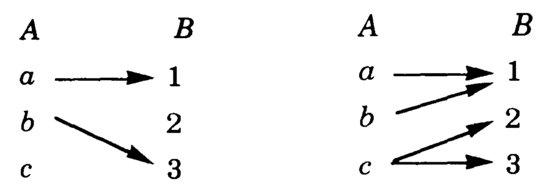
1. Functions
   - Definitions and Examples
     - Some Useful Functions
   - Composition of Functions
   - Properties of Functions
   - Infinite Sets

Subsection 1

Definitions and Examples

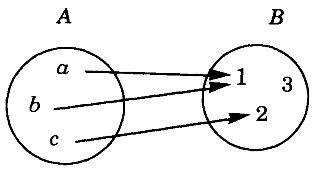
Functions

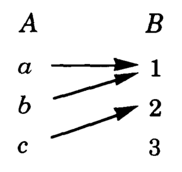
- Let A and B be sets.
- A **function from A to B** is an association to each element in A of exactly one element in B.
- Functions are normally denoted by letters like f, g and h.
- If f is a function from A to B, written $f : A \to B$ or $A \xrightarrow{f} B$, and f associates $x \in A$ with $y \in B$, then we write $y = f(x)$.
- When $f(x) = y$, we often say, "f **maps** x **to** y".
- Functions are also called **mappings**, **transformations** and **operators**.
- The following associations are not functions from A to B.

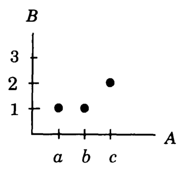


Description of Functions

- Functions can be described in many ways:
  - By a formula.
    The function $f : \mathbb{N} \to \mathbb{N}$ mapping every natural number x to its square can be described by
    $$f(x) = x^2, \text{ for all } x \in \mathbb{N}.$$
  - By a list.
    A function $g : A \to B$ from $A = \{a, b, c\}$ to $B = \{1, 2, 3\}$ may be defined by
    $$g(a) = 1, \quad g(b) = 1, \quad g(c) = 2.$$
  - By a graph (e.g., Venn diagram, digraph, Cartesian graph).







Terminology

- The set of all functions from A to B is denoted $A \to B$.
- If $f \in A \to B$, i.e., $f : A \to B$, then we say f has **type** $A \to B$.
  - The set A is called the **domain** of f.
  - The set B is the **codomain** of f.
- If $f(x) = y$, then:
  - x is an **argument** of f;
  - y is a **value** of f.
- If the domain of a function f is a product of n sets, $A_1 \times \cdots \times A_n$, then we say that f has **arity** n, or f **has** n **arguments**.
- If $(x_1, \ldots, x_n) \in A_1 \times \cdots \times A_n$, then instead of $f((x_1, \ldots, x_n))$ we usually write $f(x_1, \ldots, x_n)$.

Binary Functions and Infix Notation

- A function f with two arguments is called a **binary function**.
- Binary functions give us the option of writing $f(x, y)$ in the popular **infix form** xfy.
- Example: Consider addition of real numbers

$$+ : \mathbb{R} \times \mathbb{R} \to \mathbb{R}.$$

Instead of writing $+(4, 5)$, we usually prefer $4 + 5$.

Range, Images and Pre-Images

- The **range** of f, written $\text{range}(f)$, is the set of elements in B that are associated with some element of A:
$$\text{range}(f) = \{f(a) : a \in A\}.$$
- If $S \subseteq A$, then the **image** of S under f, written $f(S)$, is the set of values in B associated with elements of S:
$$f(S) = \{f(x) : x \in S\}.$$
- As a special case $f(A) = \text{range}(f)$.
- If $T \subseteq B$, then the **pre-image** or **inverse image** of T under f, written $f^{-1}(T)$, is the set of elements in A that associate with some elements of T:
$$f^{-1}(T) = \{a \in A : f(a) \in T\}.$$
- We have $f^{-1}(B) = A$.

Example

- Consider the function $f : \{a, b, c\} \to \{1, 2, 3\}$ defined by $f(a) = f(b) = 1$ and $f(c) = 2$.
  - f has type $\{a, b, c\} \to \{1, 2, 3\}$.
  - The domain of f is $\{a, b, c\}$.
  - The codomain of f is $\{1, 2, 3\}$.
  - The range of f is $\{1, 2\}$.
  - $f(\{a\}) = \{1\}$;
  - $f(\{a, b\}) = \{1\}$;
  - $f(A) = f(\{a, b, c\}) = \{1, 2\} = \text{range}(f)$;
  - $f^{-1}(\{1, 2\}) = \{a, b, c\}$;
  - $f^{-1}(\{1, 3\}) = \{a, b\}$;
  - $f^{-1}(\{3\}) = \emptyset$;
  - $f^{-1}(B) = f^{-1}(\{1, 2, 3\}) = \{a, b, c\} = A$.

Tuples as Functions

- Any sequence of objects can be thought of as a function.
- Example: The tuple $(22, 14, 55, 1, 700, 67)$ can be considered a listing of the values of a function

$$f : \{0, 1, 2, 3, 4, 5\} \to \mathbb{N}.$$

That is, we defined f by setting

$$f(0) = 22,\ f(1) = 14,\ f(2) = 55,\ f(3) = 1,\ f(4) = 700,\ f(5) = 67.$$

Then $(22, 14, 55, 1, 700, 67)$ is just a listing of the values of f.
- An infinite sequence can also be considered a function.
- Example: Suppose we have the following sequence of things from a set S:

$$(b_0, b_1, \ldots, b_n, \ldots).$$

The elements b_n can be considered values of the function $b : \mathbb{N} \to S$, defined by $b(n) = b_n$.

Functions and Binary Relations

- Functions are special kinds of binary relations.
- A function $f : A \to B$ is a binary relation from A to B such that

$$\text{for each } a \in A \text{ there is a unique } b \in B, \text{ such that } (a, b) \in f.$$

- We can describe this uniqueness condition in the following way:

$$\text{If } (a, b), (a, c) \in f, \text{ then } b = c.$$

- In case the relation $f \subseteq A \times B$ happens to be a function of type $A \to B$, the functional notation $f(a) = b$ is preferred over the relational notations $f(a, b)$ and $(a, b) \in f$.

Example

- Consider the sets $A = \{a, b, c, d, e\}$ and $B = \{0, 1, 2\}$.
- Let $R \subseteq A \times B$ be the following binary relation from A to B:

$$R = \{(a, 0), (b, 0), (c, 2), (d, 1), (e, 2)\}.$$

- Since R associates to each element of A a unique element of B, it is a function $R : A \to B$.
- In this case, instead of the relational $(c, 2) \in R$ or $R(c, 2)$, we may write the functional $R(c) = 2$.

Equality of Functions

- If f and g are both functions of type $A \to B$, then f and g are said to be **equal**, written $f = g$, if
$$f(x) = g(x), \text{ for all } x \in A.$$
- Example: Suppose f and g are functions of type $\mathbb{N} \to \mathbb{N}$ and they are defined by the formulas
$$f(x) = x + x$$
and
$$g(x) = 2x.$$
Then $f = g$.

Definition by Cases

- Functions can be defined **by cases**.
- Example: The absolute value function abs has type $\mathbb{R} \to \mathbb{R}$ and can be defined by the following rule:

$$\text{abs}(x) = \begin{cases} x, & \text{if } x \geq 0 \\ -x, & \text{if } x < 0 \end{cases}$$

- A definition by cases can also be written in terms of the **if-then-else** rule.
- Example: We can write the preceding definition in the form:

$$\text{abs}(x) = \text{ if } x \geq 0 \text{ then } x \text{ else } -x.$$

Partial Functions

- A **partial function** from A to B is like a function except that it might not be defined for some elements of A.
- We still have the requirement that if $x \in A$ is associated with $y \in B$, then x cannot be associated with any other element of B.
- Example: Since division by zero is not allowed, $\div$ is a partial function of type $\mathbb{R} \times \mathbb{R} \to \mathbb{R}$.
- When discussing partial functions, to avoid confusion we use the term **total function** to mean a function that is defined on all its domain.

From Partial Functions to Total Functions

- Any partial function can be transformed into a total function.
- One simple technique is to **shrink the domain** to the set of elements for which the partial function is defined.
- Example: $\div$ is a total function of type $\mathbb{R} \times (\mathbb{R} - \{0\}) \to \mathbb{R}$.
- A second technique keeps the domain the same but **increases the size of the codomain**.
- Example: Suppose $f : A \to B$ is a partial function.
  - Pick some symbol that is not in B, say $\# \notin B$;
  - Assign $f(x) = \#$ whenever $f(x)$ is not defined.

  Then we can think of f as the total function of type $A \to B \cup \{\#\}$.
- In programming, the analogy would be to pick an error message to indicate that an incorrect input string has been received.

The Floor and Ceiling Functions

- The **floor function** has type $\mathbb{R} \to \mathbb{Z}$ and is defined by

$$\text{floor}(x) = \text{the largest integer less than or equal to } x.$$

- Example: $\text{floor}(8) = 8,\ \text{floor}(8.9) = 8,\ \text{floor}(-3.5) = -4.$
- $\text{floor}(x)$ is also denoted by $\lfloor x \rfloor$.
- The **ceiling function** has type $\mathbb{R} \to \mathbb{Z}$ and is defined by

$$\text{ceiling}(x) = \text{the smallest integer greater than or equal to } x.$$

- Example: $\text{ceiling}(8) = 8,\ \text{ceiling}(8.9) = 9,\ \text{ceiling}(-3.5) = -3.$
- $\text{ceiling}(x)$ is also denoted by $\lceil x \rceil$.

A Simple Property of the Floor Function

- For all $x \in \mathbb{R}$ and all $n \in \mathbb{Z}$,

$$\lfloor x + n \rfloor = \lfloor x \rfloor + n.$$

Let $x \in \mathbb{R}$ and $n \in \mathbb{Z}$.

- If $x \in \mathbb{Z}$, then $x + n \in \mathbb{Z}$.
  So we have $\lfloor x + n \rfloor = x + n = \lfloor x \rfloor + n$.
- If $x \notin \mathbb{Z}$, then, there exists $m \in \mathbb{Z}$ and $0 < r < 1$, such that $x = m + r$.
  So we have:

$$\begin{aligned} \lfloor x + n \rfloor &= \lfloor m + r + n \rfloor = \lfloor (m + n) + r \rfloor \\ &= m + n = \lfloor m + r \rfloor + n \\ &= \lfloor x \rfloor + n. \end{aligned}$$

Floor and Ceiling: Divide and Conquer

- If $n \in \mathbb{Z}$, then
$$n = \lfloor n/2 \rfloor + \lceil n/2 \rceil.$$
Consider two cases:
  - If n is even, then $n = 2k$ for some $k \in \mathbb{Z}$.
    So we have
    $$\begin{array}{rcl} \lfloor n/2 \rfloor & = & \lfloor 2k/2 \rfloor = \lfloor k \rfloor = k; \\ \lceil n/2 \rceil & = & \lceil 2k/2 \rceil = \lceil k \rceil = k. \end{array}$$
    So $\lfloor n/2 \rfloor + \lceil n/2 \rceil = k + k = 2k = n$.
  - If n is odd, then $n = 2k + 1$ for some $k \in \mathbb{Z}$.
    In this case, we have
    $$\begin{array}{rcl} \lfloor n/2 \rfloor & = & \lfloor (2k+1)/2 \rfloor = \lfloor k + 1/2 \rfloor = k; \\ \lceil n/2 \rceil & = & \lceil (2k+1)/2 \rceil = \lceil k + 1/2 \rceil = k + 1. \end{array}$$
    So $\lfloor n/2 \rfloor + \lceil n/2 \rceil = k + k + 1 = 2k + 1 = n$.

Greatest Common Divisor

- The **greatest common divisor** of two integers a and b, not both zero, denoted $\gcd(a, b)$, is the largest number that divides them both.
- Example:

  The common divisors of 12 and 18 are $\pm 1, \pm 2, \pm 3, \pm 6$.
  So $\gcd(12, 18) = 6$.
- Example: $\gcd(-44, -12) = 4$, $\gcd(5, 0) = 5$.
- If $a \neq 0$, we have $\gcd(a, 0) = |a|$.
- If $\gcd(a, b) = 1$, we say a and b are **relatively prime**.
- Example: 9 and 4 are relatively prime.

Division Algorithm

- **Division Algorithm**:

  If a and b are integers and $b \neq 0$, then there are unique integers q and r such that $a = bq + r$, where $0 \leq r < |b|$.

- Example: If $a = 19$ and $b = 4$, then

$$19 = 4 \cdot 4 + 3.$$

- Example: If $a = -16$ and $b = 3$, then

$$-16 = 3 \cdot (-6) + 2.$$

Euclid's Algorithm

- We describe Euclid's Algorithm that calculates $\gcd(a, b)$ for a and b natural numbers that are not both zero.
- **Euclid's Algorithm**:

    Input two natural numbers a and b, not both zero.
    while $b > 0$
        Use the division algorithm to compute q and r such that
            $a = bq + r$, where $0 \leq r < b$;
        $a := b$;
        $b := r$;
    Output a.

- Apply Euclid's Algorithm to compute the gcd of 315 and 54.
    - Initialization: $a := 315$; $b := 54$;
    - While Loop:
        - Iteration 1: $315 = 54 \cdot 5 + 45$; $a = 54$; $b := 45$;
        - Iteration 2: $54 = 45 \cdot 1 + 9$; $a := 45$; $b := 9$;
        - Iteration 3: $45 = 9 \cdot 5 + 0$; $a := 9$; $b := 0$;
    - Output: $a = 9$.

Greatest Common Divisor as Linear Combination

- The following holds for all nonnegative integers a, b that are not both zero:

  If $g = \gcd(a, b)$, then there exist integers m, n, such that $g = m \cdot a + n \cdot b$.

- We can use Euclid's algorithm to find m and n.
- Keep track of the equations $a = bq + r$ from each execution of the loop:

$$\begin{aligned} 315 &= 54 \cdot 5 + 45; \\ 54 &= 45 \cdot 1 + 9; \\ 45 &= 9 \cdot 5 + 0. \end{aligned}$$

- Work backwards to solve for $\gcd(a, b)$ in terms of a and b.
  - Solve the second equation for 9:

$$9 = 54 - 45 \cdot 1.$$

  - Use the first equation to replace 45:

$$9 = 54 - (315 - 54 \cdot 5) \cdot 1 = 54 - 315 + 54 \cdot 5 = -315 + 54 \cdot 6.$$

The Mod Function

- If a and b are integers, where $b > 0$, then the division algorithm states that there are two unique integers q and r such that

$$a = bq + r, \text{ where } 0 \leq r < b.$$

- We say that q is the **quotient** and r is the **remainder** upon division of a by b.
- If a and b are integers with $b > 0$, then the remainder upon the division of a by b is denoted

$$a \mod b$$

- Example:

$$5 \mod 4 = 1; \; -5 \mod 4 = 3;$$

The Mod n Function

- Fix n as a positive integer constant.
- Define a function $f : \mathbb{Z} \to \mathbb{N}$ by
$$f(x) = x \mod n.$$
- Example: Fix $n = 3$. We have
  - $25 \mod 3 = 1$;
  - $12 \mod 3 = 0$;
  - $8 \mod 3 = 2$;
  - $-4 \mod 3 = 2$;
  - $-8 \mod 3 = 1$.
- The range of f is $\{0, 1, \ldots, n-1\}$, which is the set of possible remainders obtained upon division of x by n.
- We let $\mathbb{N}_n$ or $\mathbb{Z}_n$ denote the set
$$\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}.$$
- For example, $\mathbb{Z}_0 = \emptyset$, $\mathbb{Z}_1 = \{0\}$, and $\mathbb{Z}_2 = \{0, 1\}$.

Some Properties of the Mod n Function

(a) For all $x, y \in \mathbb{Z}$, $x \mod n = y \mod n$ iff n divides $x - y$ iff $(x - y) \mod n = 0$.

Suppose $x \mod n = y \mod n = r$.

Then we have $x = q_1 n + r$ and $y = q_2 n + r$.

Therefore, $x - y = q_1 n + r - (q_2 n + r) = (q_1 - q_2)n + 0$.

Thus, $(x - y) \mod n = 0$.

Conversely, suppose $x - y = 0 \mod n$. Then $x - y = qn$, for some $q \in \mathbb{Z}$. But then, we have $x \mod n = (y + qn) \mod n = y \mod n$.

(b) For all $a, x, y \in \mathbb{Z}$, if $ax \mod n = ay \mod n$ and $\gcd(a, n) = 1$, then $x \mod n = y \mod n$.

Suppose $ax \mod n = ay \mod n$.

Then, by Property (a), $(ax - ay) \mod n = 0$. Thus, $n \mid a(x - y)$.

But, if a positive integer divides a product and is relatively prime with one of its factors, then it must divide the other. It follows that $n \mid (x - y)$. By Property (a) again $x \mod n = y \mod n$.

From Decimal to Binary Notation

- We can use the floor and mod functions to implement division by 2:

$$x = 2 \cdot \left\lfloor \frac{x}{2} \right\rfloor + (x \mod 2).$$

- This enables writing an integer in binary notation by keeping track of remainders.
- Example: Write 53 in binary notation.

$$\begin{array}{rcl} 53 & = & 2 \cdot 26 + 1; \\ 26 & = & 2 \cdot 13 + 0; \\ 13 & = & 2 \cdot 6 + 1; \\ 6 & = & 2 \cdot 3 + 0; \\ 3 & = & 2 \cdot 1 + 1; \\ 1 & = & 2 \cdot 0 + 1. \end{array}$$

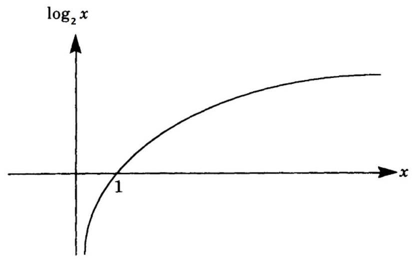
So the binary representation of 53 is 110101.

The Log Function: Definition

- Let $0 < b \neq 1$ be a fixed real number.
- The **log (logarithm) function base** b, $\log_b : \mathbb{R}_+ \to \mathbb{R}$ is defined by

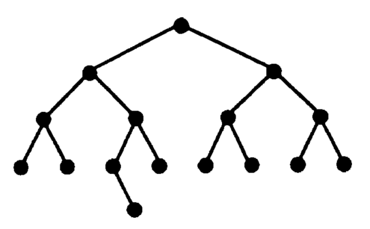
$$\log_b x = y, \text{ where } b^y = x.$$

- Example: We have:
  - $\log_2 16 = 4$;
  - $\log_3 27 = 3$;
  - $\log_7 \frac{1}{49} = -2$;
  - $\log_{32} 2 = \frac{1}{5}$;
  - $\log_8 \frac{1}{2} = -\frac{1}{3}$.



The Log Function: Application

- Consider a binary search tree with 16 nodes having the structure shown:



- Then the depth of the tree is 4.
- So a maximum of 4 comparisons are needed to find any element in the tree.
- Since $16 = 2^4$, the depth in terms of the number of nodes is:

$$4 = \log_2 16.$$

The Log Function: Properties

- The log base b satisfies the following properties:
  - $\log_b 1 = 0$ and $\log_b b = 1$;
  - $\log_b(b^x) = x$ and $b^{\log_b x} = x$;
  - $\log_b(xy) = \log_b x + \log_b y$;
  - $\log_b(\frac{x}{y}) = \log_b x - \log_b y$;
  - $\log_b(x^y) = y\log_b x$;
  - $\log_b x = \frac{\log_a x}{\log_a b}$.
- Example: Write $\log_2(2^7 3^4)$ in terms of $\log_2 3$.

  We have, using the properties above:

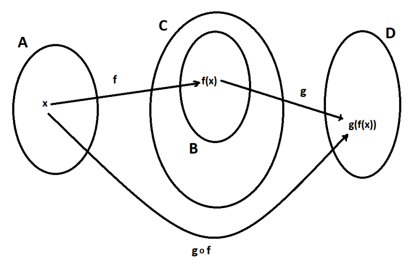
$$\log_2(2^7 3^4) = \log_2(2^7) + \log_2(3^4) = 7 + 4\log_2 3.$$

Subsection 2

Composition of Functions

Composition of Functions

- Consider two functions in which the domain of one contains the codomain of the other: $f : A \to B$, $B \subseteq C$ and $g : C \to D$.



- The **composition of f and g** is the function $g \circ f : A \to D$ defined by

$$(g \circ f)(x) = g(f(x)).$$

- This means that we first apply f to x and then apply g to the resulting value.

Examples of Composition

- Consider $f : \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x + 1$ and $g : \mathbb{R} \to \mathbb{R}$ defined by $g(x) = x^2$.

  Then we have:
  - $(g \circ f)(7) = g(f(7)) = g(8) = 64$;
  - $(f \circ g)(3) = f(g(3)) = f(9) = 10$;
  - $(g \circ f)(x) = g(f(x)) = g(x+1) = (x+1)^2$;
  - $(f \circ g)(x) = f(g(x)) = f(x^2) = x^2 + 1$;
  - $(f \circ f)(x) = f(x+1) = (x+1) + 1 = x + 2$;
- Consider $\log_2 : (0, \infty) \to \mathbb{R}$ and floor $: \mathbb{R} \to \mathbb{Z}$.

  Then we have:
  - $\text{floor}(\log_2 64) = \text{floor}(6) = 6$;
  - $\text{floor}(\log_2 5) = 2$, because $2 < \log_2 5 < 3$.

Associativity of Composition

- If f, g and h are functions of the right type such that $(f \circ g) \circ h$ and $f \circ (g \circ h)$ make sense, then
$$(f \circ g) \circ h = f \circ (g \circ h).$$
To prove this, calculate the expressions for both sides:
$$\begin{array}{rcl} ((f \circ g) \circ h)(x) & = & (f \circ g)(h(x)) = f(g(h(x))); \\ (f \circ (g \circ h))(x) & = & f((g \circ h)(x)) = f(g(h(x))). \end{array}$$
- This property allows writing the composition of three or more functions without the use of parentheses, since $f \circ g \circ h$ has exactly one meaning.

Non-Commutativity of Composition

- Composition is not commutative in general.

  This can be shown by counterexample.

  Consider $f(x) = x + 1$ and $g(x) = x^2$.

  We have

$$\begin{array}{rcl} (f \circ g)(2) & = & f(g(2)) = f(4) = 5; \\ (g \circ f)(2) & = & g(f(2)) = g(3) = 9. \end{array}$$

  So $(f \circ g)(2) \neq (g \circ f)(2)$.

  This shows that $f \circ g \neq g \circ f$.

Identity Function and Composition

- The **identity function** $\mathrm{id}_A : A \to A$ always returns its argument:

$$\mathrm{id}_A(a) = a, \text{ for all } a \in A.$$

- For every function $f : A \to B$, we have

$$f \circ \mathrm{id}_A = f = \mathrm{id}_B \circ f.$$

These equalities are easy to see: For every $a \in A$ we have:

- $(f \circ \mathrm{id}_A)(a) = f(\mathrm{id}_A(a)) = f(a)$.
- $(\mathrm{id}_B \circ f)(a) = \mathrm{id}_B(f(a)) = f(a)$.

Sequence, Distribute and Pairs Functions

- The **sequence function** $\text{seq} : \mathbb{N} \to \text{Lists}[\mathbb{N}]$ is defined by
$$\text{seq}(n) = \langle 0, 1, \ldots, n \rangle.$$
- Example: $\text{seq}(0) = \langle 0 \rangle$; $\text{seq}(4) = \langle 0, 1, 2, 3, 4 \rangle$.
- The **distribute function** $\text{dist} : A \times \text{Lists}[B] \to \text{Lists}[A \times B]$ takes an element x from A and a list y from $\text{Lists}[B]$ and returns the list of pairs made up by pairing x with each element of y.
- Example: $\text{dist}(x, \langle r, s, t \rangle) = \langle (x, r), (x, s), (x, t) \rangle$.
- The **pairs function** takes two lists of equal length and returns the list of pairs of corresponding elements.
- Example:
$$\text{pairs}(\langle a, b, c \rangle, \langle d, e, f \rangle) = \langle (a, d), (b, e), (c, f) \rangle.$$
- Since the domain of pairs is a proper subset of $\text{Lists}[A] \times \text{Lists}[B]$, it is only a partial function of type $\text{Lists}[A] \times \text{Lists}[B] \to \text{Lists}[A \times B]$.

Composition of Functions With Different Arities

- Suppose we are given the following three functions:

$$f : A \to B, \quad g : A \to C, \quad h : B \times C \to D.$$

- We can form the composition $h \circ (f, g) : A \to D$, defined, for all $x \in A$, by

$$(h \circ (f, g))(x) = h(f(x), g(x)).$$

- Example: Suppose $f : A \to \mathbb{R}$, $g : A \to \mathbb{R}$ and $+ : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Then we have that $+ \circ (f, g) : A \to \mathbb{R}$ is given, for all $x \in A$ by

$$(+ \circ (f, g))(x) = +(f(x), g(x)) = f(x) + g(x).$$

Example

- Use known functions and constructions to build the function $f : \mathbb{N} \to \text{Lists}[\mathbb{N} \times \mathbb{N}]$ defined by

$$f(n) = \langle (0,0), (1,1), \ldots, (n,n) \rangle.$$

We have

$$\begin{aligned} f(n) &= \langle (0,0), (1,1), \ldots, (n,n) \rangle \\ &= \text{pairs}(\langle 0, 1, \ldots, n \rangle, \langle 0, 1, \ldots, n \rangle) \\ &= \text{pairs}(\text{seq}(n), \text{seq}(n)). \end{aligned}$$

Example

- Use known functions and constructions to build the function $g : \mathbb{N} \to \text{Lists}[\mathbb{N} \times \mathbb{N}]$ defined by
$$g(k) = \langle (k,0),(k,1),\ldots,(k,k)\rangle, \text{ for all } k \in \mathbb{N}.$$
We have
$$\begin{array}{rcl} g(k) & = & \langle (k,0),(k,1),\ldots,(k,k)\rangle \\ & = & \text{dist}(k,\langle 0,1,\ldots,k\rangle) \\ & = & \text{dist}(k,\text{seq}(k)). \end{array}$$

The Map or Apply-To-All Function

- The **map** function $\text{map} : (A \to B) \to (\text{Lists}[A] \to \text{Lists}[B])$ takes one argument, a function $f : A \to B$, and returns as a result the function $\text{map}(f) : \text{Lists}[A] \to \text{Lists}[B]$, where $\text{map}(f)$ applies f to each element in its argument list:
$$\text{map}(f)(\langle a_1, \ldots, a_n \rangle) = \langle f(a_1), \ldots, f(a_n) \rangle.$$
- Example: Let $f : \{a, b, c\} \to \{1, 2, 3\}$ be defined by $f(a) = f(b) = 1$ and $f(c) = 2$. Then $\text{map}(f)$ applied to the list $\langle a, b, c, a \rangle$ can be calculated as follows:
$$\text{map}(f)(\langle a, b, c, a \rangle) = \langle f(a), f(b), f(c), f(a) \rangle = \langle 1, 1, 2, 1 \rangle.$$
- The map function is sometimes called the "applyToAll" function.
- Example: Consider $+$ and apply $\text{map}(+)$ to a list of pairs of integers:
$$\begin{array}{rcl} \text{map}(+)(\langle (1,2), (3,4), (5,6) \rangle) & = & \langle +(1,2), +(3,4), +(5,6) \rangle \\ & = & \langle 3, 7, 11 \rangle. \end{array}$$

Example

- Use known functions and constructions to build the function $\mathrm{squares} : \mathbb{N} \to \mathrm{Lists}[\mathbb{N}]$ defined by

$$\mathrm{squares}(n) = \langle 0, 1, 4, \ldots, n^2 \rangle.$$

We have:

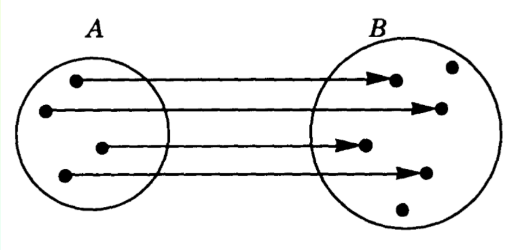
$$\begin{aligned}
\mathrm{squares}(n) &= \langle 0, 1, 4, \ldots, n^2 \rangle \\
&= \langle *(0,0), *(1,1), *(2,2), \ldots, *(n,n) \rangle \\
&= \mathrm{map}(*)(\langle (0,0), (1,1), (2,2), \ldots, (n,n) \rangle) \\
&= \mathrm{map}(*)(\mathrm{pairs}(\langle 0, 1, 2, \ldots, n \rangle, \langle 0, 1, 2, \ldots, n \rangle)) \\
&= \mathrm{map}(*)(\mathrm{pairs}(\mathrm{seq}(n), \mathrm{seq}(n))).
\end{aligned}$$

Subsection 3

Properties of Functions

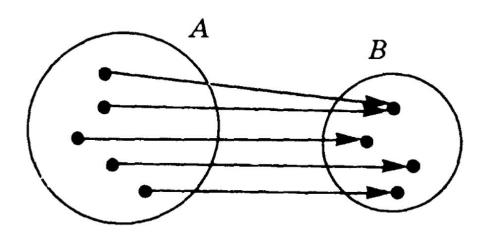
Injective Functions

- A function $f : A \to B$ is called **injective** (or **one-to-one** or an **embedding**) if no two elements in A map to the same element in B.
- Formally, f is injective if for all $x, y \in A$, whenever $x \neq y$, then $f(x) \neq f(y)$.
- By contraposition, f is injective if, for all $x, y \in A$, if $f(x) = f(y)$, then $x = y$.
- An injective function is called an **injection**.



Surjective Functions

- A function $f : A \to B$ is called **surjective** (or **onto**) if each element $b \in B$ can be written as $b = f(x)$ for some element x in A.
- Another way to say this is that f is surjective if $\text{range}(f) = B$.
- A surjective function is called a **surjection**.

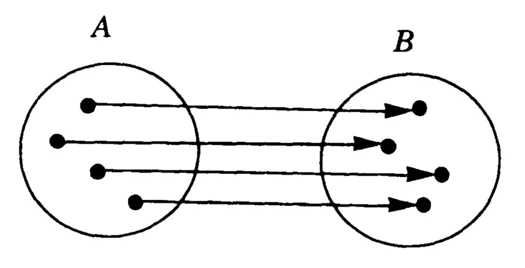


Example

- Function $f : \mathbb{R} \to \mathbb{Z}$; $f(x) = \lceil x + 1 \rceil$.
  - Injective? No! $f(0.1) = \lceil 0.1 + 1 \rceil = 2 = \lceil 0.2 + 1 \rceil = f(0.2)$.
  - Surjective? Yes! For $k \in \mathbb{Z}$, let $x \in \mathbb{R}$ be $x = k - 1$. Then $f(x) = \lceil (k - 1) + 1 \rceil = k$.
- Function $f : \mathbb{N}_8 \to \mathbb{N}_8$; $f(x) = 2x \mod 8$.
  - Injective? No! $f(0) = 0 \mod 8 = 8 \mod 8 = f(4)$.
  - Surjective? No! $\text{range}(f) = \{0, 2, 4, 6\}$.
- Function $f : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$; $f(x) = (x, x)$.
  - Injective? Yes! Suppose $x, y \in \mathbb{N}$, with $f(x) = f(y)$. Then $(x, x) = (y, y)$. This implies $x = y$.
  - Surjective? No! $(0, 1) \notin \text{range}(f)$.
- Function $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$; $f(x, y) = 2x + y$.
  - Injective? No! $f(0, 2) = 2 = f(1, 0)$.
  - Surjective? Yes! Suppose $n \in \mathbb{N}$. Let $x = (0, n) \in \mathbb{N} \times \mathbb{N}$. Then $f(0, n) = 2 \cdot 0 + n = n$.

Bijective Functions

- A function is called **bijective** (or **one-to-one and onto**) if it is both injective and surjective.
- A bijective function is called a **bijection** or a **one-to-one correspondence**.



Injectivity, Surjectivity and Composition

- Suppose $f : A \to B$ and $g : B \to C$.
  (a) If f and g are injective, then $g \circ f$ is injective.
  (b) If f and g are surjective, then $g \circ f$ is surjective.
  (c) If f and g are bijective, then $g \circ f$ is bijective.
  (d) There is an injection from A to B if and only if there is a surjection from B to A.

(a) Let $a, a' \in A$, such that $(g \circ f)(a) = (g \circ f)(a')$.
This means $g(f(a)) = g(f(a'))$.
By the injectivity of g, we get $f(a) = f(a')$.
By the injectivity of f, we get $a = a'$.
We conclude that $g \circ f$ is injective.

(b) To show that $g \circ f$ is surjective, let $c \in C$.
Since g is surjective, there exists $b \in B$, such that $g(b) = c$.
Since f is surjective, there exists $a \in A$, such that $f(a) = b$.
Thus, we get $g(f(a)) = g(b) = c$.
We conclude $g \circ f$ is surjective.

Bijections and Inverse Functions

- A function $g : B \to A$ is called an **inverse** of a function $f : A \to B$, denoted $g = f^{-1}$, if $g \circ f = \mathrm{id}_A$ and $f \circ g = \mathrm{id}_B$.
- A function $f : A \to B$ is bijective if and only if it has an inverse function $g : B \to A$.

($\Rightarrow$) Suppose that f is a bijection. To define $g : B \to A$, let $b \in B$. Since f is onto, there exists $a \in A$, such that $f(a) = b$. Since f is 1-1, there cannot exist $a' \neq a$ in A, such that $f(a') = b$. We define

$$g(b) = a, \text{ for the unique } a \in A \text{ such that } f(a) = b.$$

Then we have:

- $g(f(a)) = g(b) = a = \mathrm{id}_A(a)$.
- $f(g(b)) = f(a) = b = \mathrm{id}_B(b)$.

($\Leftarrow$) Suppose, conversely, that there exists $g : B \to A$, such that $g \circ f = \mathrm{id}_A$ and $f \circ g = \mathrm{id}_B$. Then, for all $a, a' \in A$ and $b \in B$:

- $b = \mathrm{id}_B(b) = f(g(b))$, so f is onto.
- $f(a) = f(a') \Rightarrow g(f(a)) = g(f(a')) \Rightarrow a = a'$, so f is 1-1.

Example

- Let Odd and Even be the sets of odd and even natural numbers.
  (a) Show that the function $f : \text{Odd} \to \text{Even}$ defined by $f(x) = x - 1$ is a bijection.
  (b) Find the inverse function f^{-1}.

(a) f is injective: Suppose $x_1, x_2 \in \text{Odd}$, such that $x_1 \neq x_2$. Then $x_1 - 1 \neq x_2 - 1$. So $f(x_1) \neq f(x_2)$.
f is surjective: Let $y \in \text{Even}$. Then $x = y + 1 \in \text{Odd}$ and $f(x) = (y + 1) - 1 = y$.

(b) Define $g : \text{Even} \to \text{Odd}$ by setting

$$g(y) = y + 1, \text{ for all } y \in \text{Even}.$$

It is easy to check that

$$g(f(x)) = x, \text{ for all } x \in \text{Odd},$$
$$f(g(y)) = y, \text{ for all } y \in \text{Even}.$$

So $g = f^{-1}$.

Example

- Consider the function $f : \mathbb{N}_5 \to \mathbb{N}_5$, defined by $f(x) = 2x \mod 5$.
  (a) Show that f is a bijection.
  (b) Find the inverse function f^{-1}.

(a) Since the domain of f is a small finite set, we create a table of values:

x	$f(x)$
0	0
1	2
2	4
3	1
4	3

x	$f^{-1}(x)$
0	0
1	3
2	1
3	4
4	2

- f is injective: No two elements share the same image.
- f is surjective: The range is $\mathbb{N}_5$.

(b) The table on the right specifies f^{-1}. Note that $f^{-1}(x) = 3x \mod 5$.

Subsection 4

Infinite Sets

Equipotence

- We say that two sets A and B have the **same size** or the **same cardinality** or are **equipotent**, denoted $|A| = |B|$, if there is a bijection between them.
- Formally, $|A| = |B|$ if there is a function $f : A \to B$ that is bijective.
- Example: Show that $A = \{x^2 : x \in \mathbb{N} \text{ and } 1 \leq x^2 \leq 90\}$ and $B = \{0, 1, \ldots, 8\}$ are equipotent sets.

  Note that $A = \{1, 4, 9, \ldots, 81\}$.

  Define a function $f : B \to A$, by setting

  $$f(x) = (x+1)^2, \text{ for all } x \in B.$$

  f is one-to-one and onto, so a bijection.

  We conclude that $|A| = |B|$.

Example

- Show that the set Even and the set Odd of even and odd natural numbers, respectively, have the same cardinality.

  Consider the function f : Even $\rightarrow$ Odd, defined by

  $$f(x) = x + 1, \text{ for all } x \in \text{Even}.$$

  - f is injective: If $x_1, x_2 \in$ Even, with $x_1 \neq x_2$, then $x_1 + 1 \neq x_2 + 1$. So $f(x_1) \neq f(x_2)$.
  - f is surjective: Let $y \in$ Odd. Then $x = y - 1 \in$ Even and

    $$f(x) = f(y - 1) = y - 1 + 1 = y.$$

    So f is surjective.

  We conclude that f is a bijection. So $|\text{Even}| = |\text{Odd}|$.

Example

- Show that the set Odd has the same cardinality with $\mathbb{N}$.
- Define $f : \mathbb{N} \to \text{Odd}$ by seting

$$f(x) = 2x + 1, \text{ for all } x \in \mathbb{N}.$$

  - f is injective: Suppose $x_1, x_2 \in \mathbb{N}$, with $f(x_1) = f(x_2)$. Then $2x_1 + 1 = 2x_2 + 1$. Subtracting 1 from both sides and then dividing by 2, we get $x_1 = x_2$.
  - f is surjective: Let $y \in \text{Odd}$. Then $x = \frac{y-1}{2} \in \mathbb{N}$ and

$$f(x) = f(\frac{y-1}{2}) = 2\frac{y-1}{2} + 1 = y.$$

We conclude that f is a bijection. So $|\mathbb{N}| = |\text{Odd}|$.

Inequalities Between Cardinalities

- For two sets A and B, we say **the cardinality of A is less than or equal to the cardinality of B**, written $|A| \leq |B|$,

$$\begin{aligned} &\text{iff there is an injection } f : A \to B \\ &\text{iff there is a surjection } g : B \to A. \end{aligned}$$

- If there is an injection $f : A \to B$ but no bijection between them, we write $|A| < |B|$ and say that **the cardinality of A is less than the cardinality of B**.
- Thus, the cardinality of A is less than the cardinality of B if:
  - $|A| \leq |B|$; and
  - $|A| \neq |B|$.

Countable and Uncountable Sets

- A set C is **countable** if it is finite or if $|C| = |\mathbb{N}|$.
- In the case $|C| = |\mathbb{N}|$ we sometimes say that C is **countably infinite**.
- If a set is not countable, it is called **uncountable**.
- Example: $\mathbb{N}$ is the fundamental example of a countably infinite set.
- **Important Properties**:
  - (a) Every subset of $\mathbb{N}$ is countable.
  - (b) A set S is countable if and only if $|S| \leq |\mathbb{N}|$.
  - (c) Every subset of a countable set is countable.
  - (d) Any image of a countable set is countable.

Cartesian Products of Countable Sets

- The set $\mathbb{N} \times \mathbb{N}$ is countable.

  We need to describe a bijection between $\mathbb{N} \times \mathbb{N}$ and $\mathbb{N}$.

  We arrange the elements of $\mathbb{N} \times \mathbb{N}$ so that they can be counted.

  In the following listing each row lists a sequence of tuples in $\mathbb{N} \times \mathbb{N}$ followed by a corresponding sequence of natural numbers:

$$\begin{array}{lcl} (0,0) & \leftrightarrow & 0 \\ (0,1),(1,0) & \leftrightarrow & 1,2 \\ (0,2),(1,1),(2,0) & \leftrightarrow & 3,4,5 \\ (0,3),(1,2),(2,1),(3,0) & \leftrightarrow & 6,7,8,9 \\ & \vdots & \end{array}$$

  This listing shows that we have a bijection between $\mathbb{N} \times \mathbb{N}$ and $\mathbb{N}$.

  Therefore, $\mathbb{N} \times \mathbb{N}$ is countable.

Countable Unions of Countable Sets

- If $S_0, S_1, \ldots, S_n, \ldots$ is a sequence of countable sets, then the union $S_0 \cup S_1 \cup \cdots \cup S_n \cup \cdots$ is a countable set.

  Since each set S_n is countable, its elements can be listed (possibly with repetitions) $x_{n0}, x_{n1}, x_{n2}, \ldots$.
  So the elements of the union can be arranged as on the right.

$$\begin{array}{cccc} x_{00}, & x_{01}, & x_{02}, & \cdots \\ x_{10}, & x_{11}, & x_{12}, & \cdots \\ \vdots & & & \\ x_{i0}, & x_{i1}, & x_{i2}, & \cdots \\ \vdots & & & \end{array}$$

  Thus, we have a function $f : \mathbb{N} \times \mathbb{N} \to S_1 \cup S_2 \cup \cdots$ defined by $f(m, n) = x_{mn}$.
  This mapping is surjective since the array includes all elements in the union.
  Therefore, $S_1 \cup S_2 \cup \cdots$ is the image of the countable set $\mathbb{N} \times \mathbb{N}$.
  So it is itself countable.

Countability of the Rationals

- We show that the set $\mathbb{Q}$ of rational numbers is countable.

  Let $\mathbb{Q}^+$ denote the set of positive rational numbers.

  We can represent $\mathbb{Q}^+$ as the following set of fractions (with repetitions)
  $$\mathbb{Q}^+ = \{m/n : m, n \in \mathbb{N} \text{ and } n \neq 0\}.$$
  The function $f : \mathbb{Q}^+ \to \mathbb{N} \times \mathbb{N}$, defined by $f(m/n) = (m, n)$ is an injection.

  Since $\mathbb{N} \times \mathbb{N}$ is countable, we conclude that $\mathbb{Q}^+$ is countable.

  Similarly, the set $\mathbb{Q}^-$ of negative rational numbers is countable.

  But $\mathbb{Q} = \mathbb{Q}^+ \cup \{0\} \cup \mathbb{Q}^-$ is now the union of a sequence of countable sets.

  So $\mathbb{Q}$ is countable.

Set of Strings over a Finite Alphabet

- If A is a finite alphabet, then the set A^* of all strings over A is countably infinite.

  For each $n \in \mathbb{N}$, let A_n be the set of strings over A having length n.

  It follows that A^* is the union of the sets $A_0, A_1, \ldots, A_n, \ldots$.

  Since each set A_n is finite, we conclude that A^* is countable.

Diagonalization: Uncountability of $(0, 1)$

- The set $(0, 1)$ is uncountable.

  Suppose $(0, 1)$ is countable.

  Then, its elements can be listed as $r_0, r_1, r_2, \ldots$.
  Represent each number in decimal $r_i = 0.d_{i0}d_{i1}d_{i2}\ldots$.
  In this way we get the list:

  $$\begin{array}{rcl} r_0 & = & 0.d_{00}d_{01}d_{02}\ldots \\ r_1 & = & 0.d_{10}d_{11}d_{12}\ldots \\ r_2 & = & 0.d_{20}d_{21}d_{22}\ldots \\ & \vdots & \end{array}$$

  Construct a new number $s = 0.s_0s_1s_2\ldots \in (0, 1)$ as follows:

  $$s_i = \text{if } d_{ii} = 4 \text{ then } 5 \text{ else } 4.$$

  $s \in (0, 1)$, but s does not occur in the listing above since it differs from r_i in the i-th decimal place, for all i

  So the listing above does not exhaust all numbers in $(0, 1)$.

  So $(0, 1)$ is uncountable.

Diagonalization: Uncountability of $\mathbb{N} \to \mathbb{N}$

- The set of functions $\mathbb{N} \to \mathbb{N}$ is uncountable.

  Assume, by way of contradiction, that the set is countable.

  Then we can list all the functions of type $\mathbb{N} \to \mathbb{N}$ as $f_0, f_1, f_2, \ldots$.

  Represent each function f_n as the sequence of its values $(f_n(0), f_n(1), f_n(2), \ldots)$.

  Define a function $g : \mathbb{N} \to \mathbb{N}$ by

$$g(n) = \begin{cases} 1, & \text{if } f_n(n) = 2 \\ 2, & \text{if } f_n(n) \neq 2. \end{cases}$$

  Then the sequence of values $(g(0), g(1), g(2), \ldots)$ is different from each of the sequences for the listed functions because $g(n) \neq f_n(n)$ for each n.

  It follows that $f_0, f_1, f_2, \ldots$ does not list all functions in $\mathbb{N} \to \mathbb{N}$, a contradiction.

Diagonalization: Cantor's Theorem

- Let A be a set. Then
$$|A| < |\mathcal{P}(A)|.$$
To show this we must show that:
(a) $|A| \leq |\mathcal{P}(A)|$, i.e., there is an injection from A to $\mathcal{P}(A)$;
(b) $|A| \neq |\mathcal{P}(A)|$, i.e., there is no bijection between A and $\mathcal{P}(A)$.

(a) Let $f : A \to \mathcal{P}(A)$ be defined by
$$f(a) = \{a\}, \text{ for all } a \in A.$$
f is an injection: $f(a) = f(a')$ implies $\{a\} = \{a'\}$ implies $a = a'$.

(b) Suppose $g : A \to \mathcal{P}(A)$ is a bijection.
Consider the set $D = \{a \in A : a \notin g(a)\} \in \mathcal{P}(A)$.
Since $g : A \to \mathcal{P}(A)$ is onto, there exists $d \in A$, such that $g(d) = D$.
- If $d \in D$, then $d \notin g(d)$, so $d \notin D$, contradiction.
- If $d \notin D$, then $d \in g(d)$, so $d \in D$, contradiction.

Number of Programs

- The set of all programs in a programming language is countably infinite.

  Consider each program as a finite string of symbols over a fixed finite alphabet A.

  For example, A might consist of all characters that can be typed from a keyboard.

  For each natural number n, let P_n denote the set of all programs that are strings of length n over A.

  For example, the program $\{\text{print}(4)\}$ is in P_{10} because it's a string of length 10.

  So the set of all programs is the union of the sets $P_0, P_1, \ldots, P_n, \ldots$.

  Since each P_n is finite, we get that the set of all programs is countable.

Not Everything is Computable

- There are functions of type $\mathbb{N} \to \mathbb{N}$ that cannot be computed by any computer program in a given programming language.
  Note that:
  - The set of all computer programs in a given language is countably infinite.
  - The set of all functions in $\mathbb{N} \to \mathbb{N}$ is uncountable.

  We conclude that there exist functions of type $\mathbb{N} \to \mathbb{N}$ that cannot be computed by any program in the given language.
- Over any finite alphabet, there are languages that cannot be decided by any computer program.
  Again note that:
  - The set of all computer programs in a given language is countably infinite.
  - The set of all languages in $\mathcal{P}(A^*)$ is uncountable by Cantor's Theorem.

  We conclude that there exist languages over A that cannot be decided by any program in the given language.