Summary last week

- correctness and complexity statements and proofs of Floyd's algorithm
- proof method: proof by contradiction
- relations as digraphs; whether elements relate, not how
- properties of relations: reflexivity, symmetry, transitivity, . . .
- closing a relation with respect to a property
- Warshall's transitive closure algorithm
- functions as relations; every element related to some unique element
- defining functions by specifications

Summary last week

Theorem

The following overwrites the adjacency matrix A of R, with that of R^+

For r from 0 to $n-1$ repeat:
  Set $N = A$.
  For i from 0 to $n-1$ repeat:
    For j from 0 to $n-1$ repeat:
      Set $N_{ij} = max(A_{ij}, A_{ir} \cdot A_{rj})$
  Set $A = N$.

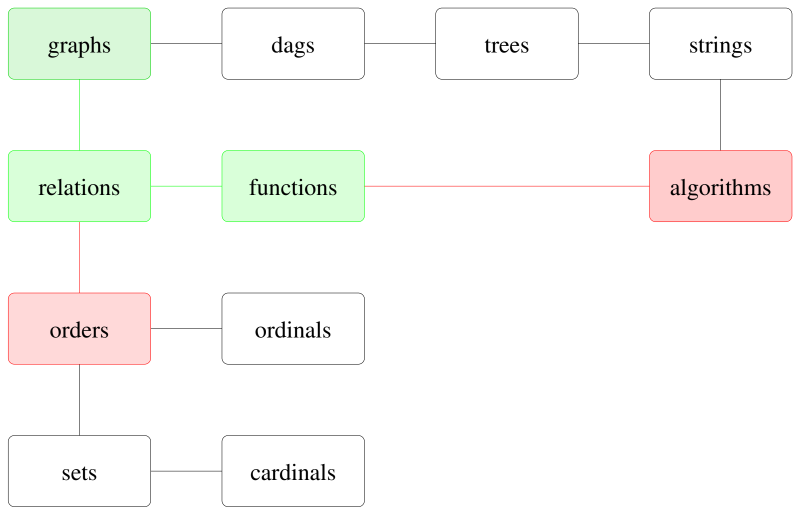
Proof.

as for Floyd ∎

Course themes

- directed and undirected graphs
- relations and functions
- orders and induction
- trees and dags
- finite and infinite counting
- elementary number theory
- Turing machines, algorithms, and complexity
- decidable and undecidable problem

Discrete structures



Functions as relations

Definition

A function on M is a relation R on M such that

1. for all $x \in M$, there exists y such that $x \; R \; y$ (totality)
2. for all $x, y, y' \in M$ if $x \; R \; y$ and $x \; R \; y'$ then $y = y'$, i.e. R relates uniquely.

we then write $R(x)$ to denote y.

Functions as relations

Definition

A function on M is a relation R on M such that

1. for all $x \in M$, there exists y such that $x \; R \; y$ (totality)
2. for all $x, y, y' \in M$ if $x \; R \; y$ and $x \; R \; y'$ then $y = y'$, i.e. R relates uniquely.

we then write $R(x)$ to denote y.

Example

- The squaring function on natural numbers is the relation $\{(0,0),(1,1),(2,4),(3,9),(4,16),\dots\}$.
- Taking the square root is not a function on natural numbers, since, e.g., the square root of 2 is not a natural number (existence fails)
- Taking the square root is not a function on the real numbers either, since, e.g., both -2 and 2 are square roots of 4 (uniqueness (also) fails)

Functions as relations

Definition

A function on M is a relation R on M such that

1. for all $x \in M$, there exists y such that $x \; R \; y$ (totality)
2. for all $x, y, y' \in M$ if $x \; R \; y$ and $x \; R \; y'$ then $y = y'$, i.e. R relates uniquely.

we then write $R(x)$ to denote y.

Specification of functions

A function is said to be defined by some specification this expresses that there exists a unique relation satisfying the specification and the relation is a function.

Example

The function f on natural numbers defined by

- $f(n) = n$? $\checkmark$or $f(n) = -1$? $\times$or $f(n) = f(n)$? $\times$
- $f(0) = 10$ and $f(1) = 2$? $\times$or $f(0) = 0$ and $f(n+1) = f(n)$? $\checkmark\dots$

How to represent functions?

How to represent functions?

Problem

Domains are typically infinite. Then representation by

- list of tuples? infinitely many tuples!
- adjacency matrix? infinitely many rows,columns!
- ...?

How to represent functions?

Problem

Domains are typically infinite. Then representation by

- list of tuples? infinitely many tuples!
- adjacency matrix? infinitely many rows,columns!
- ...?

Solution

Represent functions by algorithms. Many ways of defining algorithms, e.g. as Turing machines, Java programs, Haskell programs, ... but all equivalent

Limitations of algorithms

- There are more functions $f : \mathbb{N} \to \mathbb{N}$ than there are algorithms (programs, TMs); so some functions cannot be represented by algorithms;

Limitations of algorithms

- There are more functions $f : \mathbb{N} \to \mathbb{N}$ than there are algorithms (programs, TMs); so some functions cannot be represented by algorithms;
- No algorithms for checking interesting properties of programs (TMs) themselves; termination (halting problem), reachability (unreachable code), ... No interesting property of programs can be programmed.

Limitations of algorithms

- There are more functions $f : \mathbb{N} \to \mathbb{N}$ than there are algorithms (programs, TMs); so some functions cannot be represented by algorithms;
- No algorithms for checking interesting properties of programs (TMs) themselves; termination (halting problem), reachability (unreachable code), ... No interesting property of programs can be programmed.
- No algorithm for checking whether a formula in first-order logic is universally valid (Entscheidungsproblem).

Limitations of algorithms

- There are more functions $f : \mathbb{N} \to \mathbb{N}$ than there are algorithms (programs, TMs); so some functions cannot be represented by algorithms;
- No algorithms for checking interesting properties of programs (TMs) themselves; termination (halting problem), reachability (unreachable code), ... No interesting property of programs can be programmed.
- No algorithm for checking whether a formula in first-order logic is universally valid (Entscheidungsproblem).
- No algorithm for checking whether Diophantine equations have a solution (Hilbert's 10th problem).

Limitations of algorithms

- There are more functions $f : \mathbb{N} \to \mathbb{N}$ than there are algorithms (programs, TMs); so some functions cannot be represented by algorithms;
- No algorithms for checking interesting properties of programs (TMs) themselves; termination (halting problem), reachability (unreachable code), ... No interesting property of programs can be programmed.
- No algorithm for checking whether a formula in first-order logic is universally valid (Entscheidungsproblem).
- No algorithm for checking whether Diophantine equations have a solution (Hilbert's 10th problem).
- ...

Limitations of algorithms

- There are more functions $f : \mathbb{N} \to \mathbb{N}$ than there are algorithms (programs, TMs); so some functions cannot be represented by algorithms;
- No algorithms for checking interesting properties of programs (TMs) themselves; termination (halting problem), reachability (unreachable code), ... No interesting property of programs can be programmed.
- No algorithm for checking whether a formula in first-order logic is universally valid (Entscheidungsproblem).
- No algorithm for checking whether Diophantine equations have a solution (Hilbert's 10th problem).
- ...

Remark

These limitations will be addressed in the last few weeks of course

Functions as imperative programs

Definition

A deterministic 1-tape Turingmachine (TM) M is a 9-tuple

$$M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$$

1. Q is a finite set of states
2. Σ is a set of input symbols
3. Γ is a finite set of tape symbols with $\Sigma \subseteq \Gamma$
4. $\vdash \in \Gamma \setminus \Sigma$ is the left-end marker symbol
5. $\sqcup \in \Gamma$ ($\sqcup \neq \vdash$), is the blank symbol
6. $\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{\mathrm{L}, \mathrm{R}\}$ is the transition function
7. $s \in Q$, the initial or start state
8. $t \in Q$, the accepting state
9. $r \in Q$, the rejecting state ($t \neq r$)

Transition function

$\delta(p, a) = (q, b, d)$ means that if TM M is in state p and reads symbol a, then

1. M replaces a with b on the tape
2. the read/write-head moves one step in direction d
3. M goes into state q

Transition function

$\delta(p, a) = (q, b, d)$ means that if TM M is in state p and reads symbol a, then

1. M replaces a with b on the tape
2. the read/write-head moves one step in direction d
3. M goes into state q

Additional constraints

- The left-end marker cannot be overwritten

$$\forall\, p \in Q, \exists\, q \in Q \quad \delta(p, \vdash) = (q, \vdash, \mathrm{R})$$

Transition function

$\delta(p, a) = (q, b, d)$ means that if TM M is in state p and reads symbol a, then

1. M replaces a with b on the tape
2. the read/write-head moves one step in direction d
3. M goes into state q

Additional constraints

- The left-end marker cannot be overwritten

$$\forall\, p \in Q, \exists\, q \in Q \quad \delta(p, \vdash) = (q, \vdash, \mathrm{R})$$

- If TM is in the accepting/rejecting state, it will stay in that state

$$\forall b \in \Gamma \;\; \pi_1(\delta(t, b)) = t \text{ and } \pi_1(\delta(r, b)) = r$$

Here π_1 denotes projection on the first component

Example

Let $M = (\{s, p, t, r\}, \{0, 1\}, \{\vdash, \sqcup, 0, 1\}, \vdash, \sqcup, \delta, s, t, r)$ be a TM. Then δ can be specified by a transition table

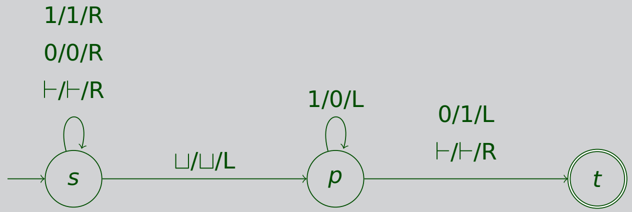
	$\vdash$	0	1	$\sqcup$
s	$(s, \vdash, \mathrm{R})$	$(s, 0, \mathrm{R})$	$(s, 1, \mathrm{R})$	$(p, \sqcup, \mathrm{L})$
p	$(t, \vdash, \mathrm{R})$	$(t, 1, \mathrm{L})$	$(p, 0, \mathrm{L})$	$\cdot$

Example

Let $M = (\{s, p, t, r\}, \{0, 1\}, \{\vdash, \sqcup, 0, 1\}, \vdash, \sqcup, \delta, s, t, r)$ be a TM. Then δ can be specified by a transition table

	$\vdash$	0	1	$\sqcup$
s	$(s, \vdash, \mathrm{R})$	$(s, 0, \mathrm{R})$	$(s, 1, \mathrm{R})$	$(p, \sqcup, \mathrm{L})$
p	$(t, \vdash, \mathrm{R})$	$(t, 1, \mathrm{L})$	$(p, 0, \mathrm{L})$	$\cdot$

or by a transition graph



Words

Definition (Alphabet)

Set Σ is an alphabet $\quad a \in \Sigma$ is a symbol

Words

Definition (Alphabet)

Set Σ is an alphabet $\quad a \in \Sigma$ is a symbol

Words

Definition (Alphabet)

Set Σ is an alphabet $a \in \Sigma$ is a symbol

Example

- $\mathbb{B} = \{0, 1\}$ is the binary alphabet
- $\{a, b, \ldots, z\}$ is the alphabet of letters
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is the alphabet of digits

Words

Definition (Alphabet)

Set Σ is an alphabet $a \in \Sigma$ is a symbol

Example

- $\mathbb{B} = \{0, 1\}$ is the binary alphabet
- $\{a, b, \ldots, z\}$ is the alphabet of letters
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is the alphabet of digits

Words

Definition (Alphabet)

Set Σ is an alphabet $a \in \Sigma$ is a symbol

Example

- $\mathbb{B} = \{0, 1\}$ is the binary alphabet
- $\{a, b, \ldots, z\}$ is the alphabet of letters
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is the alphabet of digits

Words

Definition (Alphabet)

Set Σ is an alphabet $a \in \Sigma$ is a symbol

Example

- $\mathbb{B} = \{0, 1\}$ is the binary alphabet
- $\{a, b, \ldots, z\}$ is the alphabet of letters
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is the alphabet of digits

Definition (Word)

$w = (w_0, \ldots, w_{n-1}) \in \Sigma^n$ is a word or string of length $\ell(w) = n$ over Σ

Words

Definition (Alphabet)

Set Σ is an alphabet $a \in \Sigma$ is a symbol

Example

- $\mathbb{B} = \{0, 1\}$ is the binary alphabet
- $\{a, b, \ldots, z\}$ is the alphabet of letters
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is the alphabet of digits

Definition (Word)

$w = (w_0, \ldots, w_{n-1}) \in \Sigma^n$ is a word or string of length $\ell(w) = n$ over Σ
Σ^* is the set of all words over Σ

Words

Definition (Alphabet)

Set Σ is an alphabet $a \in \Sigma$ is a symbol

Example

- $\mathbb{B} = \{0, 1\}$ is the binary alphabet
- $\{a, b, \ldots, z\}$ is the alphabet of letters
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ is the alphabet of digits

Definition (Word)

$w = (w_0, \ldots, w_{n-1}) \in \Sigma^n$ is a word or string of length $\ell(w) = n$ over Σ
Σ^* is the set of all words over Σ

Remark

As in ETI, we omit parentheses in words; words of length 1 are denoted by symbols

Definition

A configuration of a TM M is a triple (p, x, n) comprising

1. $p \in Q$ its state
2. $x = y\sqcup^\infty$ its tape content, $y \in \Gamma^*$
3. $n \in \mathbb{N}$ the position of the read/write-head

Definition

A configuration of a TM M is a triple (p, x, n) comprising

1. $p \in Q$ its state
2. $x = y\sqcup^\infty$ its tape content, $y \in \Gamma^*$
3. $n \in \mathbb{N}$ the position of the read/write-head

Definition

The initial or start configuration for input $x \in \Sigma^*$ is:

$$(s, \vdash x\sqcup^\infty, 0)$$

Definition

A configuration of a TM M is a triple (p, x, n) comprising

1. $p \in Q$ its state
2. $x = y\sqcup^\infty$ its tape content, $y \in \Gamma^*$
3. $n \in \mathbb{N}$ the position of the read/write-head

Definition

The initial or start configuration for input $x \in \Sigma^*$ is:

$$(s, \vdash x\sqcup^\infty, 0)$$

Example

For the TM M of the previous example

$$(s, \vdash 0010\sqcup^\infty, 0) \xrightarrow[M]{*} (t, \vdash 0011\sqcup^\infty, 3)$$

The step function of a TM

Definition

Let $n \in \mathbb{N}$ and $y = y_0 \cdots y_{m-1} \in \Gamma^*$ with $m > n$; the relation $\xrightarrow[M]{1}$ is defined by:

$$(p, y\sqcup^\infty, n) \xrightarrow[M]{1} \begin{cases} (q, y_0 \dots b \dots y_{m-1}\sqcup^\infty, n-1) & \delta(p, y_n) = (q, b, \mathrm{L}) \\ (q, y_0 \dots b \dots y_{m-1}\sqcup^\infty, n+1) & \delta(p, y_n) = (q, b, \mathrm{R}) \end{cases}$$

The step function of a TM

Definition

Let $n \in \mathbb{N}$ and $y = y_0 \cdots y_{m-1} \in \Gamma^*$ with $m > n$; the relation $\xrightarrow[M]{1}$ is defined by:

$$(p, y\sqcup^\infty, n) \xrightarrow[M]{1} \begin{cases} (q, y_0 \dots b \dots y_{m-1}\sqcup^\infty, n-1) & \delta(p, y_n) = (q, b, \mathrm{L}) \\ (q, y_0 \dots b \dots y_{m-1}\sqcup^\infty, n+1) & \delta(p, y_n) = (q, b, \mathrm{R}) \end{cases}$$

Definition

$\xrightarrow[M]{*}$ is the reflexive–transitive closure of $\xrightarrow[M]{1}$. That is, it corresponds to a finite sequence of consecutive steps.

Definition

a TM M

- accepts $x \in \Sigma^*$, if $\exists\, y, n$:
$$(s, \vdash x\sqcup^\infty, 0) \xrightarrow[M]{*} (t, y, n)$$
- rejects $x \in \Sigma^*$, if $\exists\, y, n$:
$$(s, \vdash x\sqcup^\infty, 0) \xrightarrow[M]{*} (r, y, n)$$
- halt on input x, if x is accepted or rejected
- does not halt in input x, if x is neither accepted nor rejected
- is total, if M halts on all inputs

Definition

a TM M

- accepts $x \in \Sigma^*$, if $\exists\, y, n$:
$$(s, \vdash x \sqcup^\infty, 0) \xrightarrow[M]{*} (t, y, n)$$
- rejects $x \in \Sigma^*$, if $\exists\, y, n$:
$$(s, \vdash x \sqcup^\infty, 0) \xrightarrow[M]{*} (r, y, n)$$
- halt on input x, if x is accepted or rejected
- does not halt in input x, if x is neither accepted nor rejected
- is total, if M halts on all inputs

Definition

A function $f : A \to B$ is defined by a TM M for every $x \in A$, M accepts input x with $f(y)$ on the tape (and does not halt or rejects on inputs $x \notin A$).

Functions as functional programs

Example

Squaring $sq : \mathbb{N} \to \mathbb{N}$

$$sq\, n = n \cdot n$$

ok, but multiplication $\cdot$? infinite table?

Functions as functional programs

Example

Squaring $sq : \mathbb{N} \to \mathbb{N}$ and multiplication $\cdot : \mathbb{N} \times \mathbb{N} \to N$

$$\begin{aligned} sq\, n &= n \cdot n \\ 0 \cdot k &= 0 \\ (1+n) \cdot k &= k + (n \cdot k) \end{aligned}$$

ok, but addition $+$? infinite table?

Functions as functional programs

Example

Squaring $sq : \mathbb{N} \to \mathbb{N}$, multiplication $\cdot : \mathbb{N} \times \mathbb{N} \to N$ and addition $+ : \mathbb{N} \times \mathbb{N} \to N$

$$\begin{aligned} sq\, n &= n \cdot n \\ 0 \cdot k &= 0 \\ (1+n) \cdot k &= k + (n \cdot k) \\ 0 + k &= k \\ (1+n) + k &= 1 + (n + k) \end{aligned}$$

ok, but successor $1+$? infinite table? (Sheet 3)

Algorithms, TMs, programs vs. functions

- finite representations of functions
- finiteness entails not all functions can be represented (later)
- a function that can be represented is called computable
- different programs may represent the same function; e.g. mergesort vs. bubblesort
- programs can be distinguished by their usage of resources; time, space, steps, energy, . . . ; functions cannot

Orders

Definition

A relation is a

- partial order if it is reflexive, anti-symmetric and transitive;
- total order if moreover every pair of elements is related either way; and
- strict order if it is irreflexive and transitive

Orders

Definition

A relation is a

- partial order if it is reflexive, anti-symmetric and transitive;
- total order if moreover every pair of elements is related either way; and
- strict order if it is irreflexive and transitive

Lemma

Strict orders are anti-symmetric

Orders

Definition

A relation is a

- partial order if it is reflexive, anti-symmetric and transitive;
- total order if moreover every pair of elements is related either way; and
- strict order if it is irreflexive and transitive

Lemma

Strict orders are anti-symmetric

Proof.

We show anti-symmetry vacuously holds, by showing that its assumption is never fulfilled. Suppose both $x\ R\ y$ and $y\ R\ x$ for some strict order R. By transitivity, then $x\ R\ x$, contradicting irreflexivity. ■

Example

The natural order $\leq$ on $\mathbb{Z}$, defined by

$$x \leq y \quad \text{if} \quad y - x \in \mathbb{N}$$

is a partial order and a total order, but not a strict order (consider $(5, 5)$).

Example

The natural order $\leq$ on $\mathbb{Z}$, defined by

$$x \leq y \quad \text{if} \quad y - x \in \mathbb{N}$$

is a partial order and a total order, but not a strict order (consider $(5, 5)$).

Example

$m \in \mathbb{N}$ divides $n \in \mathbb{N}$, if there is some $p \in \mathbb{N}$ such that

$$n = m \cdot p$$

Divisibility is a partial order, but neither total nor strict (consider $(2, 3)$ resp. $(2, 2)$)

Example

The natural order $\leq$ on $\mathbb{Z}$, defined by

$$x \leq y \quad \text{if} \quad y - x \in \mathbb{N}$$

is a partial order and a total order, but not a strict order (consider $(5, 5)$).

Example

$m \in \mathbb{N}$ divides $n \in \mathbb{N}$, if there is some $p \in \mathbb{N}$ such that

$$n = m \cdot p$$

Divisibility is a partial order, but neither total nor strict (consider $(2, 3)$ resp. $(2, 2)$)

Example

The natural order $\leq$ on $\mathbb{Z}$, defined by

$$x \leq y \quad \text{if} \quad y - x \in \mathbb{N}$$

is a partial order and a total order, but not a strict order (consider $(5, 5)$).

Example

$m \in \mathbb{N}$ divides $n \in \mathbb{N}$, if there is some $p \in \mathbb{N}$ such that

$$n = m \cdot p$$

Divisibility is a partial order, but neither total nor strict (consider $(2, 3)$ resp. $(2, 2)$)

Example

Let, for tuples $x = (x_1, x_2, \ldots, x_k)$ und $y = (y_1, y_2, \ldots, y_k)$ in M^k

$$x \; R_{\text{comp}} \; y \quad \text{if} \quad x_i \; R \; y_i \quad \text{for all } i = 1, \ldots, k$$

The componentwise extension R_{comp} of R is a partial resp. strict order, if R is on M.
Typically not total, even if R is; $(2, 1) \not\leq (1, 2)$ on $\mathbb{N}^2$.

Example

The natural order $\leq$ on $\mathbb{Z}$, defined by

$$x \leq y \ \text{ if } \ y - x \in \mathbb{N}$$

is a partial order and a total order, but not a strict order (consider $(5, 5)$).

Example

$m \in \mathbb{N}$ divides $n \in \mathbb{N}$, if there is some $p \in \mathbb{N}$ such that

$$n = m \cdot p$$

Divisibility is a partial order, but neither total nor strict (consider $(2, 3)$ resp. $(2, 2)$)

Example

Let, for tuples $x = (x_1, x_2, \dots, x_k)$ und $y = (y_1, y_2, \dots, y_k)$ in M^k

$$x\, R \qquad y \quad \text{if} \quad x_i\, R\, y_i \quad \text{for all } i = 1, \dots, k$$

The componentwise extension R_{comp} of R is a partial resp. strict order, if R is on M.
Typically not total, even if R is; $(2, 1) \not\leq (1, 2)$ on $\mathbb{N}^2$.

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Example

$\mathcal{P}(\{a, b\}) = \{\varnothing, \{a\}, \{b\}, \{a, b\}\}$

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Example

$\mathcal{P}(\{a,b\}) = \{\varnothing, \{a\}, \{b\}, \{a,b\}\}$ $\mathcal{P}_1(\{a,b\}) = \{\{a\},\{b\}\}$

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Example

$\mathcal{P}(\{a,b\}) = \{\varnothing, \{a\}, \{b\}, \{a,b\}\}$ $\mathcal{P}_1(\{a,b\}) = \{\{a\},\{b\}\}$

Theorem

The subset relation $S \subseteq T$ is a partial order on $\mathcal{P}(M)$

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Example

$\mathcal{P}(\{a,b\}) = \{\varnothing, \{a\}, \{b\}, \{a,b\}\}$ $\mathcal{P}_1(\{a,b\}) = \{\{a\},\{b\}\}$

Theorem

The subset relation $S \subseteq T$ is a partial order on $\mathcal{P}(M)$

Definition (Refining and coarsening of partitions)

Let P, Q be partitions of M. That is, $\bigcup P = M$ and $\forall p \neq p' \in P, p \cap p' = \emptyset$.

$$P \leq Q :\Leftrightarrow \text{every block of } P \text{ is a subset of a block of } Q$$

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Example

$\mathcal{P}(\{a,b\}) = \{\varnothing, \{a\}, \{b\}, \{a,b\}\}$ $\mathcal{P}_1(\{a,b\}) = \{\{a\},\{b\}\}$

Theorem

The subset relation $S \subseteq T$ is a partial order on $\mathcal{P}(M)$

Definition (Refining and coarsening of partitions)

Let P, Q be partitions of M.

$$P \leq Q :\Leftrightarrow \text{every block of } P \text{ is a subset of a block of } Q$$

If $P \leq Q$, then P is finer then Q (Q coarser than P)

Definition

$\mathcal{P}(M) := \{T \mid T \subseteq M\}$ the power set of M
$\mathcal{P}_k(M) := \{T \mid T \subseteq M \text{ and } \#(T) = k\}$ subsets of size k

Example

$\mathcal{P}(\{a,b\}) = \{\emptyset, \{a\}, \{b\}, \{a,b\}\}$ $\mathcal{P}_1(\{a,b\}) = \{\{a\}, \{b\}\}$

Theorem

The subset relation $S \subseteq T$ is a partial order on $\mathcal{P}(M)$

Definition (Refining and coarsening of partitions)

Let P, Q be partitions of M.

$$P \leq Q :\Leftrightarrow \text{every block of } P \text{ is a subset of a block of } Q$$

If $P \leq Q$, then P is finer then Q (Q coarser than P)

Example

The partition $\{\{a\}, \{b\}, \{c\}\}$ is (strictly) finer than each of

$$\{\{a\}, \{b,c\}\} \quad \{\{b\}, \{a,c\}\} \quad \{\{c\}, \{a,b\}\} \quad \{\{a,b,c\}\}$$

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

(2) *If $<$ is a strict order, then its reflexive closure*

$$x \leq y \;:\Leftrightarrow\; x < y \text{ or } x = y$$

defines a partial order

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

(2) *If $<$ is a strict order, then its reflexive closure*

$$x \leq y \;:\Leftrightarrow\; x < y \text{ or } x = y$$

defines a partial order

(3) *The functions $\leq \mapsto <$ in (1) and $< \mapsto \leq$ in (2) are inverse to each other*

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

(2) *If $<$ is a strict order, then its reflexive closure*

$$x \leq y \;:\Leftrightarrow\; x < y \text{ or } x = y$$

defines a partial order

(3) *The functions $\leq \mapsto <$ in (1) and $< \mapsto \leq$ in (2) are inverse to each other*

Remark

Partial orders defined by strict orders, and vice versa

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

(2) *If $<$ is a strict order, then its reflexive closure*

$$x \leq y \;:\Leftrightarrow\; x < y \text{ or } x = y$$

defines a partial order

(3) *The functions $\leq \mapsto <$ in (1) and $< \mapsto \leq$ in (2) are inverse to each other*

Remark

Partial orders defined by strict orders, and vice versa

Example

$< = \{(0,1),(1,2),(0,2)\}$ defines the partial order

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

(2) *If $<$ is a strict order, then its reflexive closure*

$$x \leq y \;:\Leftrightarrow\; x < y \text{ or } x = y$$

defines a partial order

(3) *The functions $\leq \mapsto <$ in (1) and $< \mapsto \leq$ in (2) are inverse to each other*

Remark

Partial orders defined by strict orders, and vice versa

Example

$< = \{(0,1),(1,2),(0,2)\}$ defines the partial order

$\leq = \{ \quad (0,1), \quad (1,2),(0,2) \quad \}$

Theorem

(1) *If $\leq$ partial order, then its predecessor relation*

$$x < y \;:\Leftrightarrow\; x \leq y \text{ and } x \neq y$$

is a strict order

(2) *If $<$ is a strict order, then its reflexive closure*

$$x \leq y \;:\Leftrightarrow\; x < y \text{ or } x = y$$

defines a partial order

(3) *The functions $\leq \mapsto <$ in (1) and $< \mapsto \leq$ in (2) are inverse to each other*

Remark

Partial orders defined by strict orders, and vice versa

Example

$< = \{(0,1),(1,2),(0,2)\}$ defines the partial order

$\leq = \{(0,0),(0,1),(1,1),(1,2),(0,2),(2,2)\}$

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.

(2) By definition, $\leq$ is reflexive.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.

(2) By definition, $\leq$ is reflexive. We show it also is transitive. Let $x, y, z \in M$ with $x \leq y$ and $y \leq z$.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.

(2) By definition, $\leq$ is reflexive. We show it also is transitive. Let $x, y, z \in M$ with $x \leq y$ and $y \leq z$. If $x = y$ and $y = z$, then $x = z$. In the other cases $x < z$, using in case $x < y$ and $y < z$ that $<$ is transitive.

■

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.
(2) By definition, $\leq$ is reflexive. We show it also is transitive. Let $x, y, z \in M$ with $x \leq y$ and $y \leq z$. If $x = y$ and $y = z$, then $x = z$. In the other cases $x < z$, using in case $x < y$ and $y < z$ that $<$ is transitive. To show $\leq$ is anti-symmetric $\leq$, it suffices to observe that $x \leq y$ and $y \leq x$ can only both hold if $x = y$ (if $y = x$)

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.
(2) By definition, $\leq$ is reflexive. We show it also is transitive. Let $x, y, z \in M$ with $x \leq y$ and $y \leq z$. If $x = y$ and $y = z$, then $x = z$. In the other cases $x < z$, using in case $x < y$ and $y < z$ that $<$ is transitive. To show $\leq$ is anti-symmetric $\leq$, it suffices to observe that $x \leq y$ and $y \leq x$ can only both hold if $x = y$ (if $y = x$) ; the other cases contradict irreflexivity of $<$

Proof.

(1) By definition, $x < y$ holds iff $x \leq y$ and $x \neq y$. Therefore, $<$ is irreflexive. To show $<$ is transitive, assume $x < y$ and $y < z$. First, $x \leq z$ by transitivity of $\leq$. Next, $x \neq z$ since otherwise $x \leq y$ and $y \leq x$ would yield $x = y$ by anti-symmetry. Hence $x < z$.
(2) By definition, $\leq$ is reflexive. We show it also is transitive. Let $x, y, z \in M$ with $x \leq y$ and $y \leq z$. If $x = y$ and $y = z$, then $x = z$. In the other cases $x < z$, using in case $x < y$ and $y < z$ that $<$ is transitive. To show $\leq$ is anti-symmetric $\leq$, it suffices to observe that $x \leq y$ and $y \leq x$ can only both hold if $x = y$ (if $y = x$) ; the other cases contradict irreflexivity of $<$
(3) Starting from a partial order $\leq$, the relation defined by $(x \leq y \,\wedge\, x \neq y) \,\vee\, x = y$ is $\leq$ again, as the construction amount to first removing all loops, and then adding them, noting $\leq$ has all loops. The other direction is similar, using that a strict order $<$ has no loops.

Definition

Let $\leq$ be a partial order on M. Then $x \in M$ is

- least in M, if for all $y \in M$, $x \leq y$
- greatest in M, if for all $y \in M$, $y \leq x$
- minimal in M, if for all $y \in M$, $y \not< x$
- maximal in M, if for all $y \in M$, $x \not< y$

Definition

Let $\leq$ be a partial order on M. Then $x \in M$ is

- least in M, if for all $y \in M$, $x \leq y$
- greatest in M, if for all $y \in M$, $y \leq x$
- minimal in M, if for all $y \in M$, $y \not< x$
- maximal in M, if for all $y \in M$, $x \not< y$

Definition

Let $\leq$ be a partial order on M. Then $x \in M$ is

- least in M, if for all $y \in M$, $x \leq y$
- greatest in M, if for all $y \in M$, $y \leq x$
- minimal in M, if for all $y \in M$, $y \not< x$
- maximal in M, if for all $y \in M$, $x \not< y$

Definition

Let $\leq$ be a partial order on M. Then $x \in M$ is

- least in M, if for all $y \in M$, $x \leq y$
- greatest in M, if for all $y \in M$, $y \leq x$
- minimal in M, if for all $y \in M$, $y \not< x$
- maximal in M, if for all $y \in M$, $x \not< y$

Definition

Let $\leq$ be a partial order . Then x is

- least , if for all y , $x \leq y$
- greatest , if for all y , $y \leq x$
- minimal , if for all y , $y \not< x$
- maximal , if for all y , $x \not< y$

Definition

Let $\leq$ be a partial order . Then x is

- least , if for all y , $x \leq y$
- greatest , if for all y , $y \leq x$
- minimal , if for all y , $y \not< x$
- maximal , if for all y , $x \not< y$

Example

$\leq$ generated by predecessor relation

$$< = \{(1,2),(1,4),(1,5),(2,4),(2,5),(3,4),(3,5),(4,5)\}$$

- minimal elements:
- maximal elements:
- least element:
- greatest element:

Definition

Let $\leq$ be a partial order . Then x is

- least , if for all y , $x \leq y$
- greatest , if for all y , $y \leq x$
- minimal , if for all y , $y \not< x$
- maximal , if for all y , $x \not< y$

Example

$\leq$ generated by predecessor relation

$$< = \{(1,2),(1,4),(1,5),(2,4),(2,5),(3,4),(3,5),(4,5)\}$$

- minimal elements: 1, 3
- maximal elements:
- least element:
- greatest element:

Definition

Let $\leq$ be a partial order . Then x is

- least , if for all y , $x \leq y$
- greatest , if for all y , $y \leq x$
- minimal , if for all y , $y \not< x$
- maximal , if for all y , $x \not< y$

Example

$\leq$ generated by predecessor relation

$$< = \{(1,2),(1,4),(1,5),(2,4),(2,5),(3,4),(3,5),(4,5)\}$$

- minimal elements: 1, 3
- maximal elements: 5
- least element:
- greatest element:

Definition

Let $\leq$ be a partial order . Then x is

- least , if for all y , $x \leq y$
- greatest , if for all y , $y \leq x$
- minimal , if for all y , $y \not< x$
- maximal , if for all y , $x \not< y$

Example

$\leq$ generated by predecessor relation

$$< = \{(1,2),(1,4),(1,5),(2,4),(2,5),(3,4),(3,5),(4,5)\}$$

- minimal elements: 1, 3
- maximal elements: 5
- least element:
- greatest element:

Definition

Let $\leq$ be a partial order . Then x is

- least , if for all y , $x \leq y$
- greatest , if for all y , $y \leq x$
- minimal , if for all y , $y \nless x$
- maximal , if for all y , $x \nless y$

Example

$\leq$ generated by predecessor relation

$$< = \{(1,2),(1,4),(1,5),(2,4),(2,5),(3,4),(3,5),(4,5)\}$$

- minimal elements: 1, 3
- maximal elements: 5
- least element:
- greatest element: 5